

### **User Guide**

# **AWS DataSync**



Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# **AWS DataSync: User Guide**

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

# **Table of Contents**

What is AWS DataSync?	. 1
Use cases	. 2
Benefits	. 2
Additional resources	3
How it works	4
DataSync transfer architecture	. 4
Transferring between on-premises storage and AWS	. 4
Transferring between AWS storage services	. 5
Transferring between AWS storage services and storage systems in other clouds	6
Concepts and terminology	. 7
Agent	7
Location	8
Task	. 8
Task execution	. 8
How DataSync transfers files, objects, and directories	8
How DataSync prepares your data transfer	. 8
How DataSync transfers your data	10
How DataSync verifies your data's integrity	10
How DataSync works with open and locked files	11
Recurring transfer options	11
Getting started	12
Sign up for an AWS account	12
Create a user with administrative access	12
Required IAM permissions for using DataSync	14
AWS managed policies	14
Customer managed policies	14
Where can I use DataSync?	14
How can I use DataSync?	
How much will DataSync cost?	15
Open-source components used by DataSync	15
Do I need an agent?	
Situations when you need a DataSync agent	15
Situations when you don't need a DataSync agent	16
Using multiple DataSync agents	16

Next steps	17
Agent requirements	17
Hypervisor requirements	17
Agent requirements for DataSync transfers	18
Agent requirements for AWS Region partitions	19
Agent management requirements	19
Deploying your agent	19
Deploying your agent on VMware	19
Deploying your agent on KVM	20
Deploying your agent on Microsoft Hyper-V	21
Deploying your Amazon EC2 agent	22
Deploying your agent on AWS Snowball Edge	27
Deploying your agent on AWS Outposts	27
Choosing a service endpoint for your agent	28
Choosing a public service endpoint	28
Choosing a FIPS service endpoint	28
Choosing a VPC service endpoint	29
Activating your agent	31
Prerequisites	31
Getting an activation key	32
Activating your agent	34
Next steps	36
Verifying your agent's network connections	37
Accessing your agent's local console	37
Verifying your agent's connection to your storage system	38
Verifying your agent's connection to the DataSync service	
Next steps	39
Connecting your network	40
1. Network connection between your storage system and agent	40
2. Network connection between your agent and DataSync service	40
Connecting your storage network to AWS	41
Choosing a service endpoint	41
3. Network connection between DataSync service and AWS storage service	41
Networking when you don't need a DataSync agent	41
How and where DataSync traffic flows through the network	42
Network security for DataSync	12

Network requirements	42
Network requirements for on-premises, self-managed, other cloud, and edge s	storage 42
Network requirements for AWS storage services	47
Network requirements for public or FIPS service endpoints	47
Network requirements for VPC service endpoints	54
Network interfaces for data transfers	57
Network interfaces for transfers with agents	57
Network interfaces for transfers without agents	58
Viewing your network interfaces	59
Architecture and routing examples with Direct Connect	59
Using Direct Connect with a DataSync VPC service endpoint	60
Using Direct Connect with a DataSync public or FIPS service endpoint	63
Next steps	64
Configuring your agent for multiple NICs	64
Transferring data	65
Where can I transfer my data?	66
Supported transfers in the same AWS account	66
Supported transfers across AWS accounts	68
Supported transfers in the same AWS Region	69
Supported transfers between AWS Regions	69
Determining if your transfer requires a DataSync agent	70
Transferring to or from on-premises storage	70
Configuring transfers with an NFS file server	71
Configuring transfers with an SMB file server	75
Configuring transfers with an HDFS cluster	85
Configuring transfers with an object storage system	90
Transferring to or from AWS storage	94
Configuring transfers with Amazon S3	95
Configuring transfers with Amazon EFS	125
Configuring transfers with FSx for Windows File Server	134
Configuring transfers with FSx for Lustre	139
Configuring transfers with FSx for OpenZFS	141
Configuring transfers with FSx for ONTAP	145
Transferring to or from other cloud storage	152
Planning transfers to or from third-party cloud storage systems	152
Configuring transfers with Google Cloud Storage	155

Configuring transfers with Microsoft Azure Blob Storage	164
Configuring transfers with Microsoft Azure Files	181
Configuring transfers with other cloud object storage	184
Transferring to or from S3 compatible storage on Snowball Edge	190
Prerequisites	191
Providing DataSync access to S3 compatible storage	191
Creating a DataSync agent in your on-premises storage environment	193
Configuring the source location for your transfer	194
Configuring the destination location for your transfer	195
Configuring your transfer settings	196
Starting your transfer	196
Limitations	197
Creating a task for transferring data	197
Creating your task	197
Task statuses	200
Partitioning large datasets with multiple tasks	200
Segmenting transferred data with multiple tasks	201
Choosing a task mode for your transfer	201
Choosing what data to transfer	206
Verifying data integrity	243
Setting bandwidth limits	246
Scheduling your task	248
Tagging your tasks	252
Starting a task to transfer data	255
Starting your task	256
Task execution statuses	257
Knowing when your task is queued	258
Cancelling your task execution	258
Monitoring data transfers	260
Understanding data transfer performance counters	260
Monitoring data transfers with CloudWatch metrics	277
CloudWatch metrics for DataSync	278
Monitoring data transfers with task reports	282
Use cases	282
Summary only task reports	282
Standard task reports	283

	Example task reports	286
	Limitations	289
	Creating your task reports	289
	Viewing your task reports	299
1	Monitoring data transfers with CloudWatch Logs	300
	Allowing DataSync to upload logs to a CloudWatch log group	300
	Configuring logging for your DataSync task	302
	Viewing DataSync task logs	305
l	ogging with CloudTrail	306
	Working with DataSync information in CloudTrail	306
	Understanding DataSync log file entries	307
1	Monitoring with EventBridge	309
	DataSync transfer events	309
1	Monitoring with manual tools	311
	Monitoring your transfer by using the DataSync console	311
	Monitoring your transfer by using the AWS CLI	312
	Monitoring your transfer by using the watch utility	314
Mai	naging resources	315
1	Managing your DataSync agent	315
-	Festing your DataSync agent's connectivity and system resources	315
F	Replacing your DataSync agent	315
(	Cleaning up DataSync resources	315
F	Reusing a DataSync agent's infrastructure	315
1	Managing your agent	315
	Agent software updates	316
	Agent statuses	316
	Troubleshooting your agent	317
F	Performing maintenance on your agent	317
	Accessing your agent's local console	317
	Configuring your agent's DHCP and DNS settings	318
	Checking your agent's system resources	321
	Synchronizing the time on your VMware agent	323
	Running maintenance-related commands for your agent	324
F	Replacing your agent	325
	Creating a new agent	325
	Updating your location with the new agent	325

	Next steps	331
	Filtering DataSync resources	331
	Parameters for filtering	331
	Filtering by location	332
	Filtering by task	333
	Cleaning up DataSync resources	334
	Deleting a DataSync agent	334
	Reusing a DataSync agent's infrastructure	335
	Deleting a DataSync location	336
	Deleting a DataSync task	336
Se	curity	338
	Data protection	339
	Encryption in transit	339
	Encryption at rest	342
	Internetwork traffic privacy	343
	Identity and access management	343
	Access management	344
	AWS managed policies	349
	Customer managed policies	359
	Using service-linked roles	362
	Tagging resources during creation	366
	Cross-service confused deputy prevention	368
	Compliance validation	370
	Resilience	371
	Infrastructure security	371
	Securing storage location credentials	372
	Using a service-managed secret encrypted with a default key	372
	Using a service-managed secret encrypted with a custom AWS KMS key	373
	Using a secret that you manage	374
Qı	ıotas	377
	Storage system, file, and object limits	377
	DataSync quotas	377
	Request a quota increase	381
Tr	oubleshooting	383
	Troubleshooting agent issues	383
	How do I connect to an Amazon EC2 agent's local console?	323

	What does the Failed to retrieve agent activation key error mean?	384
	I still can't activate an agent by using a VPC service endpoint	384
	What do I do if my agent is offline?	384
	I don't know what's going on with my agent. Can someone help me?	385
Tr	oubleshooting location issues	386
	My task failed with an NFS permissions denied error	386
	My task failed with an NFS mount error	387
	My task failed with an Amazon EFS mount error	387
	File ownership isn't maintained with NFS transfer	388
	My task can't access an SMB location that uses Kerberos	388
	My task failed with an input/output error	389
	Error: FsS3UnableToConnectToEndpoint	390
	Error: FsS3HeadBucketFailed	390
	Task fails with an Unable to list Azure Blobs on the volume root error	391
	Error: FsAzureBlobVolRootListBlobsFailed	391
	Error: SrcLocHitAccess	391
	Error: SyncTaskErrorLocationNotAdded	391
	Task with S3 source location fails with HeadObject or GetObjectTagging error	392
Tr	oubleshooting task issues	392
	Error: Invalid SyncOption value. Option: TransferMode,PreserveDeletedFiles, Value:	
	ALL,REMOVE.	392
	Task execution fails with an EniNotFound error	393
	Task execution fails with a Cannot allocate memory error	393
	Task execution has a launching status but nothing seems to be happening	394
	Task execution seems stuck in the preparing status	394
	Task execution stops before the transfer finishes	394
	Task execution fails when transferring from a Google Cloud Storage bucket	395
	There are mismatches between task execution's timestamps	395
	Task execution fails with NoMem error	395
	Object fails to transfer to Azure Blob Storage with user metadata key error	396
	There's an /.aws-datasync folder in the destination location	396
	Can't transfer symbolic links between locations using SMB	396
	Task report errors	397
Tr	oubleshooting data verification issues	397
	There are mismatches between a file's contents	397
	There's a mismatch hetween a file's SMR metadata	<b>30</b> 0

Files to transfer are no longer at source location	400
DataSync can't verify destination data	400
DataSync can't read object metadata	401
There's a mismatch in an object's system-defined metadata	. 402
Understanding data verification duration	403
Troubleshooting S3 storage costs with DataSync	. 404
Tutorials	405
Transferring from on-premises to S3 across accounts	405
Overview	405
Prerequisite: Required source account permissions	406
Prerequisite: Required destination account permissions	408
Step 1: In your source account, create a DataSync agent	409
Step 2: In your source account, create a DataSync IAM role for destination bucket access	. 409
Step 3: In your destination account, update your S3 bucket policy	411
Step 4: In your destination account, disable ACLs for your S3 bucket	413
Step 5: In your source account, create a DataSync source location for your on-premises	
storage	413
Step 6: In your source account, create a DataSync destination location for your S3	
bucket	. 413
Step 7: In your source account, create and start your DataSync task	415
Related resources	. 415
Transferring between S3 buckets across accounts	416
Overview	. 417
Prerequisite: Required source account permissions	. 418
Prerequisite: Required destination account permissions	421
Step 1: In your source account, create a DataSync IAM role for destination bucket access	. 421
Step 2: In your destination account, update your S3 bucket policy	423
Step 3: In your destination account, disable ACLs for your S3 bucket	425
Step 4: In your source account, create your DataSync locations	425
Step 5: In your source account, create and start your DataSync task	427
Troubleshooting	428
Related: Cross-account transfers with S3 buckets using server-side encryption	427
Performing a large migration	. 430
What is a large data migration?	430
Key stages of a large data migration	
Additional resources	431

Stage 1: Planning your migration	431
Gathering requirements	432
Running a proof of concept	437
Estimating migration timelines	439
Stage 2: Implementing your migration	441
Accelerating your migration with partitioning	
Running your DataSync tasks	443
Monitoring your transfers	444
DataSync API	446
Actions	446
CancelTaskExecution	449
CreateAgent	451
CreateLocationAzureBlob	456
CreateLocationEfs	462
CreateLocationFsxLustre	468
CreateLocationFsxOntap	472
CreateLocationFsxOpenZfs	477
CreateLocationFsxWindows	482
CreateLocationHdfs	487
CreateLocationNfs	494
CreateLocationObjectStorage	499
CreateLocationS3	506
CreateLocationSmb	513
CreateTask	521
DeleteAgent	531
DeleteLocation	533
DeleteTask	535
DescribeAgent	537
DescribeLocationAzureBlob	542
DescribeLocationEfs	547
DescribeLocationFsxLustre	553
DescribeLocationFsxOntap	556
DescribeLocationFsxOpenZfs	561
DescribeLocationFsxWindows	565
DescribeLocationHdfs	569
Describel esationNfs	E71

	DescribeLocationObjectStorage	578
	DescribeLocationS3	583
	DescribeLocationSmb	588
	DescribeTask	594
	DescribeTaskExecution	603
	ListAgents	619
	ListLocations	622
	ListTagsForResource	625
	ListTaskExecutions	628
	ListTasks	631
	StartTaskExecution	634
	TagResource	641
	UntagResource	644
	UpdateAgent	646
	UpdateLocationAzureBlob	648
	UpdateLocationEfs	653
	UpdateLocationFsxLustre	657
	UpdateLocationFsxOntap	660
	UpdateLocationFsxOpenZfs	663
	UpdateLocationFsxWindows	666
	UpdateLocationHdfs	670
	UpdateLocationNfs	676
	UpdateLocationObjectStorage	679
	UpdateLocationS3	685
	UpdateLocationSmb	689
	UpdateTask	696
	UpdateTaskExecution	702
Da	ita Types	704
	AgentListEntry	707
	AzureBlobSasConfiguration	709
	CmkSecretConfig	710
	CustomSecretConfig	712
	Ec2Config	714
	FilterRule	716
	FsxProtocol	718
	EcyDratacalNfc	710

722 723 725 727 729 731 732 734 736 737
725 727 731 732 734 736 737
727 729 731 732 734 736 737
729 731 732 734 736 737
731 732 734 736 737 746
732 734 736 737 746
734 736 737 746
736 737 746
737 746
746
747
749
750
751
753
754
756
757
758
760
762
763
764
766
768
770
773
775
777
779
781
782
784
787

AWS DataSync	User Guide
AWS Datasylic	USEI GUIGE

AWS Glossary ...... 800

# What is AWS DataSync?

AWS DataSync is an online data transfer service that simplifies data migration and helps you quickly, easily, and securely transfer your file or object data to, from, and between AWS storage services.

#### **On-premises storage transfers**

DataSync works with the following on-premises storage systems:

- Network File System (NFS)
- Server Message Block (SMB)
- Hadoop Distributed File Systems (HDFS)
- Object storage

#### **AWS storage transfers**

DataSync works with the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

#### Other cloud storage transfers

DataSync works with the following storage services in other clouds:

- Google Cloud Storage
- Microsoft Azure Blob Storage
- Microsoft Azure Files
- Wasabi Cloud Storage
- DigitalOcean Spaces

- Oracle Cloud Infrastructure Object Storage
- Cloudflare R2 Storage
- Backblaze B2 Cloud Storage
- NAVER Cloud Object Storage
- Alibaba Cloud Object Storage Service
- IBM Cloud Object Storage
- Seagate Lyve Cloud

#### **Edge storage transfers**

DataSync works with the following edge storage services and devices:

Amazon S3 compatible storage on AWS Snowball Edge

### **Use cases**

These are some of the main use cases for DataSync:

- Migrate data Transfer active datasets rapidly over the network into AWS storage services.
   DataSync includes automatic encryption and data integrity validation to help make sure that your data arrives securely, intact, and ready to use.
- Archive cold data Move cold data stored in on-premises storage directly to durable and secure long-term storage classes such as S3 Glacier Flexible Retrieval or S3 Glacier Deep Archive. Doing so can free up on-premises storage capacity and shut down legacy systems.
- Replicate data Copy data into any Amazon S3 storage class, choosing the most cost-effective storage class for your needs. You can also send data to Amazon EFS, FSx for Windows File Server, FSx for Lustre, or FSx for OpenZFS for a standby file system.
- Transfer data for timely in-cloud processing Transfer data in or out of AWS for processing.
   This approach can speed up critical hybrid cloud workflows across many industries. These include machine learning in the life-sciences industry, video production in media and entertainment, bigdata analytics in financial services, and seismic research in the oil and gas industry.

# **Benefits**

By using DataSync, you can get the following benefits:

Use cases 2

 Automate data movement – DataSync makes it easier to transfer data over the network between storage systems and services. DataSync automates both the management of datatransfer processes and the infrastructure required for high performance and secure data transfers.

- Transfer data securely DataSync provides end-to-end security, including encryption and
  data integrity validation, to help ensure that your data arrives securely, intact, and ready to
  use. DataSync accesses your AWS storage through built-in AWS security mechanisms, such as
  AWS Identity and Access Management (IAM) roles. It also supports virtual private cloud (VPC)
  endpoints, giving you the option to transfer data without traversing the public internet and
  further increasing the security of data copied online.
- Move data faster DataSync uses a purpose-built network protocol and a parallel, multithreaded architecture to accelerate your transfers. This approach speeds up migrations, recurring data-processing workflows for analytics and machine learning, and data-protection processes.

### **Additional resources**

We recommend that you read the following:

- DataSync resources Includes blogs, videos, and other training materials
- AWS re:Post See the latest discussion around DataSync
- AWS DataSync pricing

Additional resources 3

# **How AWS DataSync works**

Learn the key concepts and terminology related to AWS DataSync transfers, including how data gets transferred from on-premises and cloud locations.

# **DataSync transfer architecture**

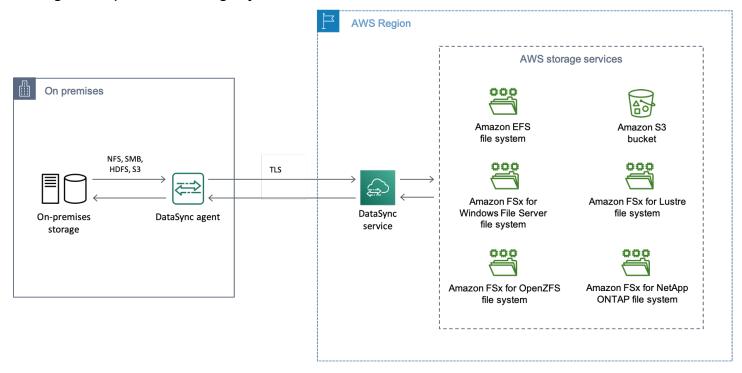
The following diagrams show how and where DataSync commonly transfers storage data. For a full list of DataSync supported storage systems and services, see <a href="Where can I transfer my data with AWS DataSync?">Where can I transfer my data with AWS DataSync?</a>

#### **Topics**

- Transferring between on-premises storage and AWS
- Transferring between AWS storage services
- Transferring between AWS storage services and storage systems in other clouds

# Transferring between on-premises storage and AWS

The following diagram shows a high-level overview of DataSync transferring files between self-managed, on-premises storage systems and AWS services.

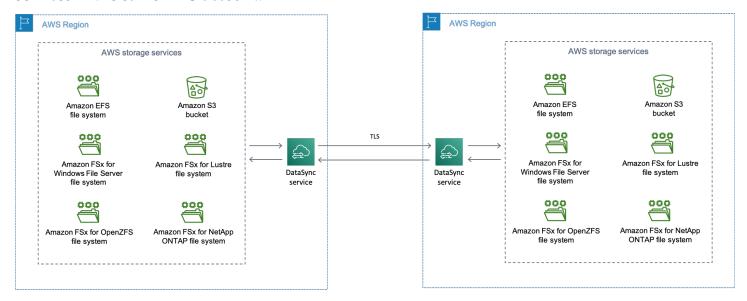


The diagram illustrates a common DataSync use case:

- A DataSync agent copying data from an on-premises storage system.
- Data moving into AWS, encrypted using Transport Layer Security (TLS).
- DataSync copying data to a supported AWS storage service.

## Transferring between AWS storage services

The following diagram shows a high-level overview of DataSync transferring files between AWS services in the same AWS account.



The diagram illustrates a common DataSync use case:

- DataSync copying data from a supported AWS storage service.
- Data moving across AWS Regions, encrypted using TLS.
- DataSync copying data to a supported AWS storage service.

When transferring between AWS storage services (whether in the same AWS Region or across AWS Regions), your data remains in the AWS network and doesn't traverse the public internet.

#### Important

You pay for data transferred between AWS Regions. This is billed as data transfer OUT from your source Region to your destination Region. For more information, see Data transfer pricing.

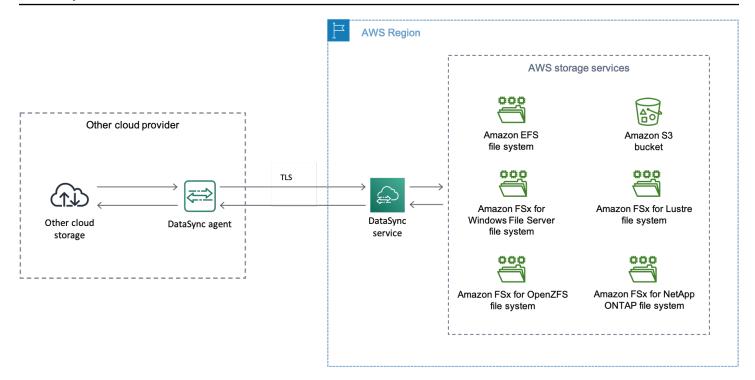
# Transferring between AWS storage services and storage systems in other clouds

With DataSync, you can transfer data between other cloud storage systems and AWS services. In this context, cloud storage systems can include:

- Self-managed storage systems, such as an NFS file server in your virtual private cloud (VPC) within AWS.
- Storage systems or services hosted by another cloud provider. For more information, see Transferring to or from other cloud storage with AWS DataSync.

DataSync can copy data to and from other clouds with or without using an agent. For more information about when to use an agent, see Do I need an AWS DataSync agent?.

The following diagram shows a high-level overview of DataSync transferring data between AWS storage services and another cloud provider.



# **Concepts and terminology**

Familiarize yourself with DataSync transfer features.

#### **Topics**

- Agent
- Location
- Task
- Task execution

### **Agent**

An *agent* is a virtual machine (VM) appliance that DataSync uses to read from and write to storage during a transfer.

You can deploy an agent in your storage environment on VMware ESXi, Linux Kernel-based Virtual Machine (KVM), or Microsoft Hyper-V hypervisors. For storage in a virtual private cloud (VPC) in AWS, you can deploy an agent as an Amazon EC2 instance.

To get started, see Do I need an AWS DataSync agent?

Concepts and terminology 7

#### Location

A *location* describes where you're copying data from or to. Each DataSync transfer (also known as a *task*) has a source and destination location. For more information, see Where can I transfer my data with AWS DataSync?

#### **Task**

A *task* describes a DataSync transfer. It identifies a source and destination location along with details about how to copy data between those locations. You also can specify how a task handles metadata, deleted files, and permissions.

#### Task execution

A *task execution* is an individual run of a DataSync transfer task. There are several phases involved in a task execution. For more information, see Task execution statuses.

# How DataSync transfers files, objects, and directories

During a <u>task execution</u>, DataSync prepares, transfers, and verifies your data. How DataSync performs these actions depends on how you configure your DataSync task options, such as the <u>task mode</u>. Basic mode tasks prepare, transfer, and verify your data sequentially, while Enhanced mode tasks do these in parallel.

#### **Topics**

- How DataSync prepares your data transfer
- How DataSync transfers your data
- How DataSync verifies your data's integrity
- How DataSync works with open and locked files
- Recurring transfer options

# How DataSync prepares your data transfer

DataSync by default prepares your transfer by examining your source and destination locations to determine what to transfer. This is done by scanning the contents and metadata of both locations to identify differences between the two.

Location 8



#### Note

If you configure your task to transfer all data, there's no preparation. When you start your task, DataSync immediately transfers everything from your source to your destination without comparing locations.

How DataSync prepares your transfer also depends on your task mode:

Enhanced mode preparation	Basic mode preparation
DataSync prepares objects as they're found at the source location. Preparation continues throughout the task execution until there are no more objects listed at the source.	Preparation can take just minutes, a few hours, or even longer depending on the number of files, objects, or directories in both locations and the performance of your storage.
Unlike Basic mode, DataSync can prepare virtually unlimited numbers of objects with each task execution.	The items that DataSync inventories in your source and destination count towards your task quotas. These quotas aren't based on the number of items that DataSync transfers during each task execution.

DataSync might skip some files, objects, and directories during preparation. The reasons for this can depend on several factors, such as how you configure your task and storage system permissions. Here are some examples:

- There's a file that exists in your source and destination locations. The file in the source hasn't been modified since the previous task execution. Since you're only transferring data that has changed, DataSync doesn't transfer that file next time you run your task.
- An object that exists in both of your locations changes in your source. When you run your task, DataSync skips this object in your destination because your task doesn't overwrite data in the destination.
- DataSync skips an object in your source location that's using an archival storage class and isn't restored. You must restore an archived object for DataSync to read it.

• DataSync skips a file, object, or directory in your source location because it can't read it. If this happens and isn't expected, check your storage's access permissions and make sure that DataSync can read what was skipped.

### How DataSync transfers your data

DataSync copies your data (including metadata) from the source to the destination based on your task options. For example, you can specify what <u>metadata</u> gets copied, <u>exclude</u> certain files, and limit how much <u>bandwidth</u> DataSync uses, among other options.

How DataSync transfers your data also depends on your task mode:

Enhanced mode transferring	Basic mode transferring
DataSync transfers each object as soon as it's prepared.	Once DataSync prepares all of your data, the transfer begins.

DataSync might skip some items during the transfer. If you configure your task to <u>transfer all data</u>, this can happen with an object in your source location that's using an <u>archival storage class</u> and isn't restored.

# How DataSync verifies your data's integrity

DataSync always performs integrity checks on your data during a transfer. At the end of a transfer, DataSync can also perform additional checks on just the transferred data or the entire dataset in both locations. For more information, see Configuring how AWS DataSync verifies data integrity.

When checking data integrity, DataSync calculates and compares the checksum and metadata of the files, objects, or directories in your locations. If DataSync notices differences between locations, verification fails with an error. For example, you might see errors such as Checksum failure, Metadata failure, Files were added, or Files were removed.

How verification works depends on your task mode and whether you configure DataSync to verify data integrity at the end of your transfer.

Enhanced mode verification	Basic mode verification
DataSync verifies each object as it's transferr ed to your destination.	At the end of your transfer, DataSync verifies the integrity of your data.
With Enhanced mode, DataSync <u>verifies only</u> <u>transferred data</u> .	Depending on how you <u>configure data</u> <u>verification</u> , this can take a significant amount of time for large datasets.

### How DataSync works with open and locked files

Keep in mind the following when trying to transfer files that are open (in use) or locked:

- In general, DataSync can transfer open files without any limitations.
- If a file is open and being written to during a transfer, DataSync can detect this kind of inconsistency during the transfer task's verification phase. To get the latest version of the file, you must run the task again.
- If a file is locked and the server prevents DataSync from opening it, DataSync skips the file during the transfer and logs an error.
- DataSync can't lock or unlock files.

### **Recurring transfer options**

In addition to one-time transfers, DataSync can transfer data on a recurring basis. Some of the options for these situations include:

- Scheduling when your task executes.
- Transferring only the data that's changed since the previous task execution.
- Deleting data in the destination location that's no longer present in the source.

# **Getting started with AWS DataSync**

Before you get started with AWS DataSync, you need to sign up for an AWS account if you don't have one. We also recommend learning where DataSync can be used and how much it might cost to transfer your data.

# Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

#### To sign up for an AWS account

- 1. Open https://portal.aws.amazon.com/billing/signup.
- 2. Follow the online instructions.

Part of the sign-up procedure involves receiving a phone call or text message and entering a verification code on the phone keypad.

When you sign up for an AWS account, an AWS account root user is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to <a href="https://aws.amazon.com/">https://aws.amazon.com/</a> and choosing **My Account**.

### Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

#### Secure your AWS account root user

 Sign in to the <u>AWS Management Console</u> as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

Sign up for an AWS account 12

For help signing in by using root user, see <u>Signing in as the root user</u> in the *AWS Sign-In User Guide*.

2. Turn on multi-factor authentication (MFA) for your root user.

For instructions, see <u>Enable a virtual MFA device for your AWS account root user (console)</u> in the *IAM User Guide*.

#### Create a user with administrative access

1. Enable IAM Identity Center.

For instructions, see <u>Enabling AWS IAM Identity Center</u> in the *AWS IAM Identity Center User Guide*.

2. In IAM Identity Center, grant administrative access to a user.

For a tutorial about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the AWS IAM Identity Center User Guide.

#### Sign in as the user with administrative access

• To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

For help signing in using an IAM Identity Center user, see <u>Signing in to the AWS access portal</u> in the *AWS Sign-In User Guide*.

#### Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

For instructions, see Create a permission set in the AWS IAM Identity Center User Guide.

2. Assign users to a group, and then assign single sign-on access to the group.

For instructions, see Add groups in the AWS IAM Identity Center User Guide.

# Required IAM permissions for using DataSync

DataSync can transfer your data to an Amazon S3 bucket, Amazon EFS file system, or a <u>number</u> <u>of other AWS storage services</u>. To get your data where you want it to go, you need the right IAM permissions granted to your identity. For example, the IAM role that you use with DataSync needs permission to use the Amazon S3 operations required to transfer data to an S3 bucket.

You can grant these permissions with IAM policies provided by AWS or by creating your own policies.

#### **Contents**

- · AWS managed policies
- Customer managed policies

### **AWS managed policies**

AWS provides the following managed policies for common DataSync use cases:

- AWSDataSyncReadOnlyAccess Provides read-only access to DataSync.
- AWSDataSyncFullAccess Provides full access to DataSync and minimal access to its dependencies.

For more information, see AWS managed policies for AWS DataSync.

### **Customer managed policies**

You can create custom IAM policies to use with DataSync. For more information, see <u>IAM customer</u> managed policies for AWS DataSync.

# Where can I use DataSync?

For a list of AWS Regions and endpoints that DataSync supports, see <u>AWS DataSync endpoints and quotas</u> in the *AWS General Reference*.

# How can I use DataSync?

There are several ways to use DataSync:

- DataSync console, which is part of the AWS Management Console.
- DataSync API or the AWS CLI to programmatically configure and manage DataSync.
- AWS CloudFormation or Terraform to provision your DataSync resources.
- AWS SDKs to build applications that use DataSync.

# How much will DataSync cost?

On the <u>DataSync pricing</u> page, create a custom estimate using the amount of data that you plan to transfer.

# Open-source components used by DataSync

To view the open-source components used by DataSync, download the following link:

datasync-open-source-components.zip

# Do I need an AWS DataSync agent?

To use AWS DataSync, you might need an agent. An *agent* is a virtual machine (VM) appliance that you deploy in your storage environment for data transfers.

Whether you need an agent depends on several factors, including the type of storage you're transferring to or from, if you're transferring across AWS accounts, and which AWS Regions you're transferring between. Before reading further, <a href="mailto:check that DataSync supports the transfer you're interested in.">check that DataSync supports the transfer you're interested in.</a>

After you determine that DataSync supports your transfer scenario, review the following information to help you understand whether you need an agent.

# Situations when you need a DataSync agent

Most situations that require a DataSync agent involve storage that's managed by you or another cloud provider.

- Transferring between AWS storage services and on-premises storage
- Transferring between Amazon EFS or Amazon FSx and storage in other clouds

- Transferring to or from edge storage
- Transferring between some AWS storage services across AWS accounts (when neither storage service is Amazon S3)

For more information, see Supported transfers across AWS accounts.

• Transferring between a commercial AWS Region and an AWS GovCloud (US) where the source and destination are either Amazon EFS or Amazon FSx.

### Situations when you don't need a DataSync agent

The situations that don't require an agent apply whether you're transferring in the <u>same AWS</u> Region or across Regions.

- Transferring between AWS storage services in the same AWS account
- Transferring between Amazon S3 and a different AWS storage service across AWS accounts
- Transferring between Amazon S3 and object storage in other clouds
- Transferring between a commercial AWS Region and an AWS GovCloud (US) where either the source or destination is Amazon S3

### **Using multiple DataSync agents**

While most transfers only need one agent, using multiple agents can speed up transfers for large datasets with millions of files or objects. In these situations, we recommend running transfer tasks in parallel. This approach spreads the transfer workload across multiple tasks, with each task using its own agent. It also helps reduce the time it takes DataSync to prepare and transfer your data. For more information, see Partitioning large datasets with multiple tasks.

Another option—especially if you have millions of small files—is to use multiple agents with a transfer location. For example, you can connect up to four agents to your on-premises Network File System (NFS) file service. This option might speed up your transfer, although the time it takes DataSync to prepare the transfer doesn't change.

With either approach, be mindful that these can increase the I/O operations on your storage and affect your network bandwidth. For more information on using multiple agents for your DataSync transfers, see the AWS Storage Blog.

If you're thinking of using multiple agents, remember the following:

• Using multiple agents with a location doesn't provide high availability. All the agents associated with a location must be online before you can start your transfer task. If one of the agents is offline, you can't run your task.

• If you're <u>using a virtual private cloud (VPC) service endpoint</u> to communicate with the DataSync service, all the agents must use the same endpoint and subnet.

### **Next steps**

- If you need an agent, review the <u>agent requirements</u> to understand what makes sense for your storage environment.
- If you don't need an agent for your transfer, you can start configuring your transfer.

# Requirements for AWS DataSync agents

Before you <u>deploy</u> an AWS DataSync agent in your storage environment, make sure that you understand the agent hypervisor and resource requirements.

# **Hypervisor requirements**

You can run a DataSync agent on the following hypervisors:

- VMware ESXi (version 6.5, 6.7, 7.0, or 8.0): VMware ESXi is available on the <u>Broadcom website</u>.
   You also need a VMware vSphere client to connect to the host.
- Linux Kernel-based Virtual Machine (KVM): A free, open-source virtualization technology.
   KVM is included in Linux versions 2.6.20 and newer. DataSync is tested and supported for the CentOS/RHEL 7 and 8, Ubuntu 16.04 LTS, and Ubuntu 18.04 LTS distributions. Other modern Linux distribution might work, but function or performance is not guaranteed. You must enable hardware accelerated virtualization on your KVM host to deploy your DataSync agent.

We recommend this option if you already have a KVM environment up and running and you're already familiar with how KVM works.

Running KVM on Amazon EC2 isn't supported and can't be used for DataSync agents.

• Microsoft Hyper-V (version 2012 R2, 2016, or 2019): For this setup, you need a Microsoft Hyper-V Manager on a Microsoft Windows client computer to connect to the host.

Next steps 17

The DataSync agent is a generation 1 virtual machine (VM). For more information about the differences between generation 1 and generation 2 VMs, see Should I create a generation 1 or 2 virtual machine in Hyper-V?

• Amazon EC2: DataSync provides an Amazon Machine Image (AMI) that contains the DataSync image. For the recommended instance types, see Amazon EC2 instance requirements.

### Agent requirements for DataSync transfers

For DataSync transfers, your agent must meet the following resource requirements.



#### Important

Keep in mind that the agent requirements for working with up to 20 million files, objects, or directories are general guidelines. Your agent may need more resources because of other factors, such as how many directories you have and object metadata size. For example, the m5.2xlarge instance for an Amazon EC2 agent still might not be enough for a transfer of less than 20 million files.

#### **Contents**

- Virtual machine requirements
- Amazon EC2 instance requirements

### Virtual machine requirements

When deploying a DataSync agent that isn't on an Amazon EC2 instance, the agent VM requires the following resources:

- Virtual processors: Four virtual processors assigned to the VM.
- **Disk space**: 80 GB of disk space for installing the VM image and system data.
- RAM: Depending on your transfer scenario, you need the following amount of memory:
  - 32 GB of RAM assigned to the VM for task executions working with up to 20 million files, objects, or directories.
  - 64 GB of RAM assigned to the VM for task executions working with more than 20 million files, objects, or directories.

### **Amazon EC2 instance requirements**

When deploying a DataSync agent on an Amazon EC2 instance, the instance size must be at least 2xlarge.

We recommend using one of the following instance sizes:

- m5.2xlarge: For task executions working with up to 20 million files, objects, or directories.
- m5.4xlarge: For task executions working with more than 20 million files, objects, or directories.

### Agent requirements for AWS Region partitions

DataSync agent images are associated with specific AWS Region partitions. For example, by default you can't download an agent in a commercial AWS Region and then activate it in an AWS GovCloud (US) Region.

### Agent management requirements

Once you activate your DataSync agent, AWS manages the agent for you. For more information, see Managing your AWS DataSync agent.

# Deploying your AWS DataSync agent

When creating an AWS DataSync agent, the first step is to deploy the agent in your storage environment. You can deploy an agent as a virtual machine (VM) on VMware ESXi, Linux Kernelbased Virtual Machine (KVM), and Microsoft Hyper-V hypervisors. You also can deploy an agent as an Amazon EC2 instance in a virtual private cloud (VPC) within AWS.



Before you begin, confirm whether you need a DataSync agent.

# **Deploying your agent on VMware**

You can download an agent from the DataSync console and deploy it in your VMware environment.

Before you begin: Make sure that your storage environment can support a DataSync agent. For more information, see Virtual machine requirements.

#### To deploy an agent on VMware

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, choose **Agents**, and then choose **Create agent**.
- 3. For **Hypervisor**, choose **VMWare ESXi**, and then choose **Download the image**.
  - The agent downloads in a .zip file that contains an .ova image file.
- 4. To minimize network latency, deploy the agent as close as possible to the storage system that DataSync needs to access (the same local network if possible). For more information, see Network requirements for on-premises, self-managed, other cloud, and edge storage.
  - If needed, see your hypervisor's documentation on how to deploy an .ova file in a VMware host.
- 5. Power on your hypervisor, log in to the agent VM, and get the agent's IP address. You need this IP address to activate the agent.
  - The agent VM's default credentials are login **admin** and password **password**. If needed, change the password through the VM's local console.

Next step: Choosing a service endpoint for your AWS DataSync agent

### Deploying your agent on KVM

You can download an agent from the DataSync console and deploy it in your KVM environment.

**Before you begin**: Make sure that your storage environment can support a DataSync agent. For more information, see Virtual machine requirements.

#### To deploy an agent on KVM

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, choose **Agents**, and then choose **Create agent**.
- For Hypervisor, choose Kernel-based Virtual Machine (KVM), and then choose Download the image.

The agent downloads in a .zip file that contains a .gcow2 image file.

Deploying your agent on KVM 20

4. To minimize network latency, deploy the agent as close as possible to the storage system that DataSync needs to access (the same local network if possible). For more information, see Network requirements for on-premises, self-managed, other cloud, and edge storage.

5. Run the following command to install your .qcow2 image.

```
virt-install \
    --name "datasync" \
    --description "DataSync agent" \
    --os-type=generic \
    --ram=32768 \
    --vcpus=4 \
    --disk path=datasync-yyyymmdd-x86_64.qcow2,bus=virtio,size=80 \
    --network default,model=virtio \
    --graphics none \
    --virt-type kvm \
    --import
```

For information about how to manage this VM and your KVM host, see your hypervisor's documentation.

6. Power on your hypervisor, log in to your VM, and get the IP address of the agent. You need this IP address to activate the agent.

The agent VM's default credentials are login **admin** and password **password**. If needed, change the password through the VM's local console.

Next step: Choosing a service endpoint for your AWS DataSync agent

# Deploying your agent on Microsoft Hyper-V

You can download an agent from the DataSync console and deploy it in your Microsoft Hyper-V environment.

**Before you begin**: Make sure that your storage environment can support a DataSync agent. For more information, see <u>Virtual machine requirements</u>.

#### To deploy an agent on Hyper-V

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Agents**, and then choose **Create agent**.

For Hypervisor, choose Microsoft Hyper-V, and then choose Download the image. 3.

The agent downloads in a .zip file that contains a .vhdx image file.

To minimize network latency, deploy the agent as close as possible to the storage system that 4. DataSync needs to access (the same local network if possible). For more information, see Network requirements for on-premises, self-managed, other cloud, and edge storage.

If needed, see your hypervisor's documentation on how to deploy a .vhdx file in a Hyper-V host.



#### Marning

You may notice poor network performance if you enable virtual machine queue (VMQ) on a Hyper-V host that's using a Broadcom network adapter. For information about a workaround, see the Microsoft documentation.

Power on your hypervisor, log in to your VM, and get the IP address of the agent. You need this 5. IP address to activate the agent.

The agent VM's default credentials are login **admin** and password **password**. If needed, change the password through the VM's local console.

Next step: Choosing a service endpoint for your AWS DataSync agent

## **Deploying your Amazon EC2 agent**

You might deploy a DataSync agent as an Amazon EC2 instance when transferring data between:

- A self-managed cloud storage system (for example, an NFS file server in AWS) and an AWS storage service.
- A cloud storage provider (such as Microsoft Azure Blob Storage or Google Cloud Storage) and an AWS storage service.
- An S3 bucket in a commercial AWS Region and an S3 bucket in an AWS GovCloud (US) Region.
- Amazon S3 on AWS Outposts and an AWS storage service.



#### **∧** Warning

We don't recommend using an Amazon EC2 agent with on-premises storage because of increased network latency. Instead, deploy the agent as a VMware, KVM, or Hyper-V virtual machine in your data center as close to your on-premises storage as possible.

#### **Deploying your EC2 agent**

#### To choose the agent AMI for your AWS Region

Open a terminal and copy the following AWS CLI command to get the latest DataSync Amazon Machine Image (AMI) ID for the Region where you want to deploy your Amazon EC2 agent.

```
aws ssm get-parameter --name /aws/service/datasync/ami --region your-region
```

Run the command. In the output, take note of the "Value" property with the DataSync AMI 2. ID.

#### **Example Example command and output**

```
aws ssm get-parameter --name /aws/service/datasync/ami --region us-east-1
{
    "Parameter": {
        "Name": "/aws/service/datasync/ami",
        "Type": "String",
        "Value": "ami-1234567890abcdef0",
        "Version": 6,
        "LastModifiedDate": 1569946277.996,
        "ARN": "arn:aws:ssm:us-east-1::parameter/aws/service/datasync/ami"
    }
}
```

#### To deploy your Amazon EC2 agent



#### (i) Tip

To avoid charges for transferring across Availability Zones, deploy your agent in a way that it doesn't require network traffic between Availability Zones. (To learn more about data transfer prices for all AWS Regions, see Amazon EC2 Data Transfer pricing.) For example, deploy your agent in the Availability Zone where your self-managed cloud storage system is located.

#### Copy the following URL:

https://console.aws.amazon.com/ec2/v2/home?region=agentregion#LaunchInstanceWizard:ami=ami-id

- Replace agent-region with the Region where you want to deploy your agent.
- Replace ami-id with the DataSync AMI ID that you obtained.
- Paste the URL into a browser. 2.

The Amazon EC2 instance launch page in the AWS Management Console displays.

- For **Instance type**, choose one of the recommended Amazon EC2 instances for DataSync. 3.
- For **Key pair**, choose an existing key pair, or create a new one. 4.
- 5. For **Network settings**, choose **Edit** and then do the following:
  - For **VPC**, choose a VPC where you want to deploy your agent. a.
  - For **Auto-assign public IP**, choose whether you want your agent to be accessible from the b. public internet.

You use the instance's public or private IP address later to activate your agent.

- For **Firewall (security groups)**, create or a select a security group that does the following:
  - If needed, allows inbound traffic to the Amazon EC2 instance on port 80 (HTTP). Some options for getting an agent activation key require this connection.
  - Allows inbound and outbound traffic between the Amazon EC2 instance the storage system that you're transferring data to or from. For more information, see Network requirements for on-premises, self-managed, other cloud, and edge storage.



#### Note

There are additional ports to configure depending on the type of service endpoint that your agent uses.

- (Recommended) To increase performance when transferring from a cloud-based file system, expand **Advanced details** and choose a **Placement group** value where your storage is located.
- Choose Launch instance to launch your Amazon EC2 instance. 7.
- 8. Once your instance status is **Running**, choose the instance.
- If you configured your instance to be accessible from the public internet, make note of the 9. instance's public IP address. If you didn't, make note of the private IP address.

You need this IP address when activating your agent.

#### Examples: Deploying your EC2 agent in an AWS Region

The following guidance can help with common scenarios if you deploy an DataSync agent in an AWS Region.

#### **Topics**

- Deploying your agent for transfers between cloud storage and AWS storage services
- Deploying your agent for transfers between Amazon S3 and AWS file systems

## Deploying your agent for transfers between cloud storage and AWS storage services

To transfer data between AWS accounts, or between cloud storage systems, the DataSync agent must be located in the same AWS Region and AWS account where the source file system resides. This type of transfer includes the following:

- Transfers between Amazon EFS or Amazon FSx to AWS storage in a different AWS account.
- Transfers from self-managed file systems to AWS storage services.

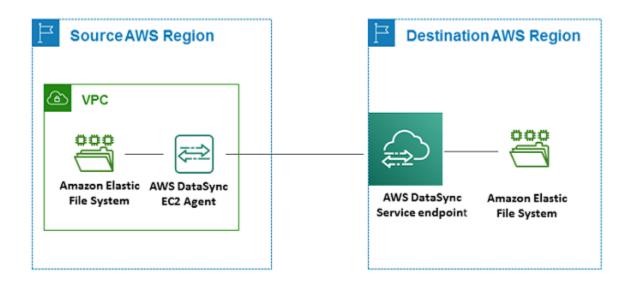
#### Important

Deploy your agent such that it doesn't require network traffic between Availability Zones (to avoid charges for such traffic).

- To access your Amazon EFS or FSx for Windows File Server file system, deploy the agent in an Availability Zone that has a mount target to your file system.
- For self-managed file systems, deploy the agent in the Availability Zone where your file system resides.

To learn more about data transfer prices for all AWS Regions, see Amazon EC2 On-Demand pricing.

For example, the following diagram shows a high-level view of the DataSync architecture for transferring data from in-cloud Network File System (NFS) to in-cloud NFS or Amazon S3.

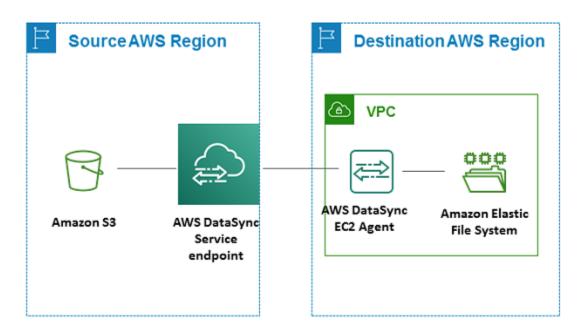


Remember the following when transferring between AWS storage services across AWS accounts:

- When transferring between Amazon EFS file systems or Amazon FSx file systems using the NFS protocol, configure your source file system as an NFS location.
- When transferring between Amazon FSx file systems using the SMB protocol, configure your source file system as an SMB location.

#### Deploying your agent for transfers between Amazon S3 and AWS file systems

The following diagram provides a high-level view of the DataSync architecture for transferring data from Amazon S3 to an AWS file system, such as Amazon EFS or Amazon FSx. You can use this architecture to transfer data from one AWS account to another, or to transfer data from Amazon S3 to a self-managed in-cloud file system.



# Deploying your agent on AWS Snowball Edge

For more information and instructions, see <u>Creating a DataSync agent in your on-premises storage</u> environment for Amazon S3 compatible storage.

# **Deploying your agent on AWS Outposts**

You can launch a DataSync Amazon EC2 instance on your Outpost. To learn more about launching an AMI on AWS Outposts, see <u>Launch an instance on your Outpost</u> in the *AWS Outposts User Guide*.

When using DataSync to access Amazon S3 on Outposts, you must launch the agent in a VPC that's allowed to access your Amazon S3 access point, and activate the agent in the parent Region of the Outpost. The agent must also be able to route to the Amazon S3 on Outposts endpoint for the bucket. To learn more about working with Amazon S3 on Outposts endpoints, see <a href="Working with Amazon S3">Working with Amazon S3</a> on Outposts in the Amazon S3 User Guide.

# Choosing a service endpoint for your AWS DataSync agent

A <u>service endpoint</u> is how your AWS DataSync <u>agent communicates with the DataSync service</u>. DataSync supports the following types of service endpoints:

- Public service endpoint Data is sent over the public internet.
- Federal Information Processing Standard (FIPS) service endpoint Data is sent over the public internet by using processes that comply with FIPS.
- Virtual private cloud (VPC) service endpoint Data is sent through your VPC instead of over the public internet, increasing the security of your transferred data.

You need a service endpoint to <u>activate your agent</u>. When choosing a service endpoint, remember the following:

- An agent can only use one type of endpoint. If you need to transfer data using different endpoint types, create an agent for each type.
- How you connect your storage network to AWS determines what service endpoints you can use.

# Choosing a public service endpoint

If you use a public service endpoint, all communication between your DataSync agent and the DataSync service occurs over the public internet.

- 1. Determine the DataSync public service endpoint that you want to use.
- 2. <u>Configure your network</u> to allow the traffic required for using DataSync public service endpoints.

Next step: Activating your AWS DataSync agent

# Choosing a FIPS service endpoint

DataSync provides some service endpoints that comply with FIPS. For more information, see <u>FIPS</u> endpoints in the *AWS General Reference*.

- 1. Determine the DataSync FIPS service endpoint that you want to use.
- 2. Configure your network to allow the traffic required for using DataSync FIPS service endpoints.

Next step: Activating your AWS DataSync agent

# **Choosing a VPC service endpoint**

If you use a VPC service endpoint, your data isn't transferred across the public internet. DataSync instead transfers data through a VPC that's based on the Amazon VPC service.

#### **Contents**

- How DataSync agents work with VPC service endpoints
- DataSync limitations with VPCs
- Creating a VPC service endpoint for DataSync

### How DataSync agents work with VPC service endpoints

VPC service endpoints are provided by AWS PrivateLink. These types of endpoints let you privately connect supported AWS services to your VPC. When you use a VPC service endpoint with DataSync, all communication between your DataSync agent and the DataSync service remains in your VPC.

The VPC service endpoint (along with the <u>network interfaces</u> DataSync creates for data transfer traffic) are private IP addresses that are only accessible from inside your VPC. For more information, see <u>Connecting your network for AWS DataSync transfers</u>.

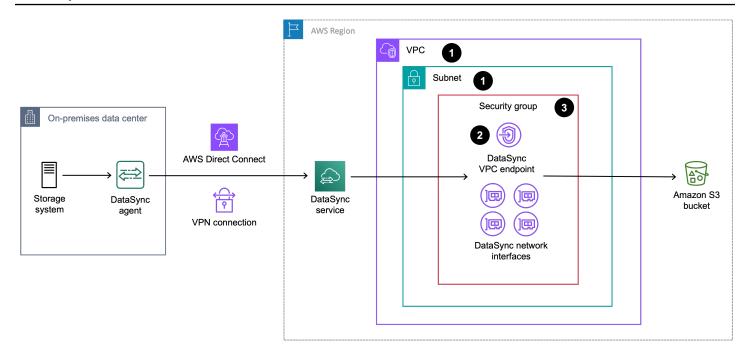
# **DataSync limitations with VPCs**

- VPCs that you use with DataSync must have default tenancy. VPCs with dedicated tenancy aren't supported.
- DataSync doesn't support shared VPCs.
- DataSync VPC service endpoints only support IPv4. IPv6 and dualstack options aren't supported.

## Creating a VPC service endpoint for DataSync

You create a VPC service endpoint for DataSync in a VPC that you manage. Your service endpoint, VPC, and DataSync agent must belong to the same AWS account.

The following diagram shows an example of DataSync using a VPC service endpoint for transferring from an on-premises storage system to an Amazon S3 bucket. The numbered callouts correspond to the steps to create a VPC service endpoint.



#### To create a VPC service endpoint for DataSync

1. Create or determine a VPC and subnet where you want to create your VPC service endpoint.

If you're transferring to or from storage that's outside AWS, the VPC should extend to that storage environment (for example, your storage environment might be a data center where your on-premises NFS file server is located). You can do this by using routing rules over <u>AWS</u> <u>Direct Connect or VPN</u>.

- 2. Create a DataSync VPC service endpoint by doing the following:
  - a. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
  - b. In the left navigation pane, choose **Endpoints**, then choose **Create endpoint**.
  - c. For **Service category**, choose **AWS services**.
  - d. For **Services**, search for **datasync** and choose the endpoint for the Region you're in (for example, com.amazonaws.us-east-1.datasync).
  - e. For **VPC**, choose the VPC where you want to create the VPC service endpoint.
  - f. Expand **Additional settings** and clear the **Enable Private DNS Name** check box to disable this setting.

We recommend disabling this setting in case you have agents in the same VPC that need to use a public service endpoint. An agent can't reach a <u>public service endpoint</u> over the network when this setting is enabled.

For **Subnet**, choose the subnet where you want to create the VPC service endpoint. Take note of the subnet ARN (you need this when activating your agent).

- Choose Create endpoint. Take note of the endpoint ID (you need this when activating your agent).
- In your VPC, configure a security group that allows the traffic required for using DataSync VPC 3. service endpoints. Take note of the security group ARN (you need this when activating your agent).

The security group must allow your agent to connect with the private IP addresses of the VPC service endpoint and your network interfaces (which get created when you create your task).

Next step: Activating your AWS DataSync agent

# **Activating your AWS DataSync agent**

To finish creating your AWS DataSync agent, you must activate it. This step associates the agent with your AWS account.



#### Note

You can't activate an agent in more than one AWS account and AWS Region at a time.

# **Prerequisites**

To activate your DataSync agent, make sure that you have the following information:

The DataSync service endpoint that you're activating your agent with.

If you're using a VPC service endpoint, you need these details:

- The VPC service endpoint ID.
- The subnet where your VPC service endpoint is located.
- The security group that allows the traffic required for using DataSync VPC service endpoints.

• Your agent's IP address or domain name.

31 Activating your agent

How you find this depends on the type of agent that you deploy. For example, if your agent is an Amazon EC2 instance, you can find its IP address by going to the instance's page on the Amazon EC2 console.

# **Getting an activation key**

You can obtain an activation key for your deployed DataSync agent a few different ways. Some options require access to your agent on port 80 (HTTP). If you use one of these options, DataSync closes the port once you activate the agent.



#### Note

Agent activation keys expire in 30 minutes if unused.

#### DataSync console

When activating your agent in the DataSync console, DataSync can get the activation key for you by using the Automatically get the activation key from your agent option.

To use this option, your browser must be able to reach your agent on port 80.

#### Agent local console

Unlike the other options for getting an activation key, this option doesn't require your agent to be accessible on port 80.

- Log in to the local console of your agent virtual machine (VM) or Amazon EC2 instance.
- 2. On the AWS DataSync Activation - Configuration main menu, enter 0 to get an activation key.
- 3. Enter the AWS Region that you're activating your agent in.
- 4. Enter the type of service endpoint type that your agent is using.
- Copy the activation key that displays. 5.

For example: F0EFT-7FPPR-GG7MC-3I9R3-27D0H

You specify this key when activating your agent.

Getting an activation key 32

#### CLI

With standard Unix tools, you can run a curl request to your agent's IP address to get its activation key.

To use this option, your client must be able to reach your agent on port 80. You can run the following command to check:

```
nc -vz agent-ip-address 80
```

Once you confirm you can reach the agent, run one of the following commands depending on the type of service endpoint that you're using:

• Public service endpoints:

```
curl "http://agent-ip-address/?gatewayType=SYNC&activationRegion=your-
region&no_redirect"
```

• FIPS service endpoints:

```
curl "http://agent-ip-address/?gatewayType=SYNC&activationRegion=your-
region&endpointType=FIPS&no_redirect"
```

• VPC service endpoints:

```
curl "http://agent-ip-address/?gatewayType=SYNC&activationRegion=your-
region&privateLinkEndpoint=vpc-endpoint-ip-
address&endpointType=PRIVATE_LINK&no_redirect"
```

To find the *vpc-endpoint-ip-address*, open the <u>Amazon VPC console</u>, choose **Endpoints**, and select your DataSync VPC service endpoint. On the **Subnets** tab, locate the IP address for your VPC service endpoint's subnet. This is the endpoint's IP address.

This command returns an activation key. For example:

```
F0EFT-7FPPR-GG7MC-3I9R3-27D0H
```

You specify this key when <u>activating your agent</u>.

Getting an activation key 33

# **Activating your agent**

You have several options for activating your DataSync agent. Once activated, AWS <u>manages the</u> agent for you.

#### DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Agents**, and then choose **Create agent**.
- 3. In the **Service endpoint** section, do the following to specify the service endpoint for your agent:
  - For a public service endpoint, choose Public service endpoints in your current AWS
     Region.
  - For a FIPS service endpoint, choose FIPS service endpoints in your current AWS
     Region.
  - For a VPC service endpoint, do the following:
    - Choose VPC endpoints using AWS PrivateLink.
    - For **VPC endpoint**, choose the VPC service endpoint that you want your agent to use.
    - For **Subnet**, choose the subnet where your VPC service endpoint is located.
    - For Security group, choose the security group that allows the traffic required for using DataSync VPC service endpoints.
- 4. In the **Activation key** section, do one of the following to specify your agent's activation key:
  - Choose Automatically get the activation key from your agent for DataSync to get the key for you.
    - For **Agent address**, enter your agent's IP address or domain name.
    - Choose Get key.
      - If activation fails, <u>check your network configuration</u> based on the type of service endpoint you're using.
  - Choose **Manually enter your agent's activation key** if you don't want a connection between your browser and agent.
    - Get the key from the agent local console or by using a curl command.
    - Back in the DataSync console, enter the key in the Activation key field.

Activating your agent 34

- (Recommended) For **Agent name**, give your agent a name that you can remember. 5.
- 6. (Optional) For **Tags**, enter values for the **Key** and **Value** fields to tag your agent.

Tags help you manage, filter, and search for your AWS resources.

- 7. Choose Create agent.
- On the **Agents** page, verify that your agent is using the correct service endpoint type.



#### Note

At this point, you might notice that your agent is offline. This happens briefly after activating an agent.

#### **AWS CLI**

- Once you get your activation key, copy one of the following create-agent commands depending on the type of service endpoint that you're using:
  - Public or FIPS service endpoint:

```
aws datasync create-agent \
  --activation-key \ activation-key \
  --agent-name name-for-agent
```

• VPC service endpoint:

```
aws datasync create-agent \
 --activation-key activation-key \
  --agent-name name-for-agent \
 --vpc-endpoint-id \
vpc-endpoint-id \
  --subnet-arns subnet-arn \
  --security-group-arns security-group-arn
```

- 2. For --activation-key, specify your agent activation key.
- 3. (Recommended) For --agent-name, specify a name for your agent that you can remember.
- If you're using a VPC service endpoint, specify the following options:
  - For --vpc-endpoint-id, specify the ID of the VPC service endpoint that you're using.

35 Activating your agent

• For --subnet-arns, specify the ARN of the subnet where your VPC service endpoint is located.

- For --security-group-arns, specify the ARN of the security group that allows the traffic required for using DataSync VPC service endpoints.
- 5. Run the create-agent command.

You get a response with the ARN of the agent that you just activated. For example:

```
{
    "AgentArn": "arn:aws:datasync:us-east-1:111222333444:agent/
agent-0b0addbeef44baca3"
}
```

Verify that your agent is activated by running the list-agents command:

```
aws datasync list-agents
```



#### Note

At this point, you might notice that your agent Status is OFFLINE. This happens briefly after activating an agent.

#### DataSync API

Once you get your activation key, activate your agent by using the CreateAgent operation.



#### Note

When you're done, you might notice that your agent is offline. This happens briefly after activating an agent.

# **Next steps**

- Verify your agent's connection to your storage system and the DataSync service.
- If you run into issues trying to activate your agent, get help with troubleshooting.

Next steps

• Create the DataSync location that you want to use with your agent. This might be an onpremises or other cloud location.

# Verifying your agent's network connections

Once you activate your AWS DataSync agent, make sure that the agent has network connectivity to your storage system and the DataSync service.

# Accessing your agent's local console

How you access your agent's local console depends on the type of agent you're using.

#### Accessing the local console (VMware ESXi, Linux KVM, or Microsoft Hyper-V)

For security reasons, you can't remotely connect to the local console of the DataSync agent virtual machine (VM).

If this is your first time using the local console, log in with the default credentials. The default user name is **admin** and the password is **password**.



#### Note

We recommend changing the default password. To do this, on the console main menu enter 5 (or 6 for VMware VMs), then run the passwd command to change the password.

#### Accessing the local console (Amazon EC2)

To connect to an Amazon EC2 agent's local console, you must use SSH.

Before you begin: Make sure that your EC2 instance's security group allows access with SSH (TCP port 22).

Open a terminal and copy the following ssh command:

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-ip-address
```

• For /path/key-pair-name, specify the path and file name (.pem) of the private key required to connect to your instance.

- For instance-user-name, specify admin.
- For instance-public-ip-address, specify the public IP address of your instance.
- 2. Run the ssh command to connect to the instance.

Once connected, the main menu of the agent's local console displays.

# Verifying your agent's connection to your storage system

Test whether your DataSync agent can connect to your storage system. For more information, see 1. Network connection between your storage system and agent.

- 1. Access your agent's local console.
- 2. On the AWS DataSync Activation Configuration main menu, enter 3.
- 3. Enter one of the following options:
  - a. Enter 1 to test an NFS server connection.
  - b. Enter 2 to test an SMB server connection.
  - c. Enter **3** to test an object storage server connection.
  - d. Enter 4 to test an HDFS connection.
  - e. Enter **5** to test a Microsoft Azure Blob Storage connection.
- 4. Enter the storage server's IP address or domain name.

Remember the following when entering the IP address or domain name:

- Don't include a protocol. For example, enter mystorage.com instead of https://mystorage.com.
- For HDFS, enter the IP address or domain name of the NameNode or DataNode in the Hadoop cluster.
- 5. If requested, enter the TCP port for connecting to the storage server (for example, 443).

See if the connectivity test **PASSED** or **FAILED**.

# Verifying your agent's connection to the DataSync service

Test whether your DataSync agent can connect to the DataSync service. For more information, see 2. Network connection between your agent and DataSync service.

- 1. Access your agent's local console.
- 2. On the **AWS DataSync Activation Configuration** main menu, enter **2** to begin testing network connectivity.

If your agent is activated, the **Test Network Connectivity** option can be initiated without any additional user input, because the Region and endpoint type are taken from the activated agent information.

- 3. Enter the type of DataSync service endpoint that your agent uses:
  - a. For public service endpoints, enter **1** and the AWS Region where your agent is activated.
  - b. For FIPS service endpoints, enter **2** and the Region where your agent is activated.
  - c. For VPC service endpoints, enter **3**.

You see a **PASSED** or **FAILED** message.

4. If you see a **FAILED** message, check your network configuration. For more information, see AWS DataSync network requirements.

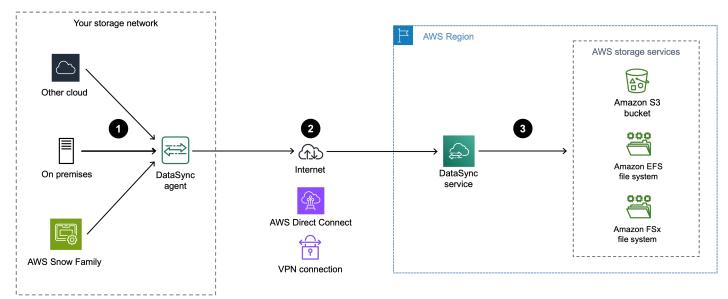
## **Next steps**

Create the DataSync location that you want to use with your agent. This might be an <u>on-premises</u> or <u>other cloud</u> location.

Next steps 39

# Connecting your network for AWS DataSync transfers

If you need an AWS DataSync agent, you must establish several network connections for a data transfer. The following diagram shows the three network connections in a DataSync transfer from a storage system (which could be on premises, in another cloud, or at the edge) to an AWS storage service.



# 1. Network connection between your storage system and agent

Your DataSync agent connects to your on-premises, other cloud, or edge storage system. For more information, see <a href="Network requirements for on-premises">Network requirements for on-premises</a>, self-managed, other cloud, and edge storage.

# 2. Network connection between your agent and DataSync service

There are a few aspects to connecting your agent to the DataSync service. First, you must connect your storage network to AWS. Second, your agent needs a service endpoint to communicate with DataSync.

#### **Contents**

- Connecting your storage network to AWS
- Choosing a service endpoint

# Connecting your storage network to AWS

When using DataSync, consider the following options for connecting your storage network to AWS:

 AWS Direct Connect - With <u>Direct Connect</u>, you can create a dedicated connection between your storage network and AWS. From a DataSync perspective, this lets you:

- Transfer data over a private path to your virtual private cloud (VPC), which avoids routing over the public internet.
- Get a more predictable connection than using a virtual private network (VPN) to connect your storage network to AWS (particularly if your agent is an Amazon EC2 instance).
- Use any type of DataSync service endpoint, including <u>public</u>, <u>Federal Information Processing</u> Standard (FIPS), or VPC endpoints.

For more information, see DataSync architecture and routing examples with AWS Direct Connect.

- **VPN** You can connect your storage network to AWS by using a VPN (such as <u>AWS Site-to-Site</u> VPN).
- Public internet You can connect your storage network directly to DataSync over the internet by using a public or FIPS service endpoint.

# Choosing a service endpoint

Your agent uses a service endpoint to communicate with DataSync. For more information, see Choosing a service endpoint for your AWS DataSync agent.

# 3. Network connection between DataSync service and AWS storage service

To connect DataSync to an AWS storage service, you just have to make sure that the DataSync service can access your S3 bucket or file system. For more information, see <a href="Network requirements">Network requirements</a> for AWS storage services.

# Networking when you don't need a DataSync agent

For transfers that <u>don't require a DataSync agent</u>, you just have to make sure that the DataSync service can access the AWS storage systems you're transferring between. For more information, see Network requirements for AWS storage services.

# How and where DataSync traffic flows through the network

DataSync has data plane and control plane traffic. Knowing how each of these flows through the network is important if you want to separate your DataSync traffic.

- Data plane traffic Includes the file or object data moving between your storage locations. In most cases, data plane traffic routes through network interfaces that DataSync automatically generates and manages when you create a task. Where these network interfaces get created depends on the type of AWS storage service you're transferring to or from and the service endpoint that your DataSync agent uses.
- Control plane traffic Includes management activities for your DataSync resources. This traffic routes through the service endpoint that your agent uses.

# **Network security for DataSync**

For information about how your storage data (including metadata) is secured during a transfer, see AWS DataSync encryption in transit.

# AWS DataSync network requirements

Configuring your network is an important step in setting up AWS DataSync. Your network configuration depends on several factors, such as what kind of storage systems you're working with. It's also based on what kind of DataSync service endpoint that you plan to use.

# Network requirements for on-premises, self-managed, other cloud, and edge storage

The following network requirements can apply to on-premises, self-managed, other cloud, and edge storage systems. These are typically storage systems that you manage or might be managed by another cloud provider.



#### Note

Depending on your network, you might need to allow traffic on ports other than what's listed here for your DataSync agent to connect with your storage.

From	То	Protocol	Port	How it's used by DataSync
DataSync agent	NFS file server	TCP	2049 (for NFS versions 4.1 and 4.0) 111 and 2049 (for NFS version 3.x)	Mounts the NFS file server.  DataSync supports NFS versions 3.x, 4.0, and 4.1.
DataSync agent	SMB file server	TCP	139 or 445	Mounts the SMB file server.  DataSync supports SMB versions 1.0 and later. For security reasons, we recommend using SMB version 3.0.2 or later. Earlier versions, such as SMB 1.0, contain known security vulnerabilities that attackers can exploit to compromise your data.
DataSync agent	Object storage	TCP	443 (HTTPS) or 80 (HTTP)  (i) Note Dependent	Accesses your Amazon S3-compatible object storage on-premises or in other clouds.

From	То	Protocol	Port	How it's used by DataSync
			your	
			object	
			storage	
			you	
			might	
			need	
			to	
			allow	
			traffic	
			on	
			nonsta	
			rd	
			HTTPS	
			and	
			HTTP	
			ports	
			(such	
			as	
			8443	
			or	
			8080).	

From	То	Protocol	Port	How it's used by DataSync
DataSync agent	Hadoop cluster	TCP	NameNode port (default is 8020) In most clusters, you can find this port number in the coresite .xml file under the fs.default or fs.default.name property (depending on the Hadoop distribut ion).	Accesses the NameNodes in your Hadoop cluster. Specify the port used when creating an HDFS location.

From	То	Protocol	Port	How it's used by DataSync
DataSync	Hadoop cluster	TCP	DataNode port (default is 50010)  In most clusters, you can find this port number in the hdfs- site .xml file under the dfs.datan ode.addre ss property.	Accesses the DataNodes in your Hadoop cluster. The DataSync agent automatically determines the port to use.
DataSync agent	Hadoop Key Managemen t Server (KMS)	ТСР	KMS port (default is 9600)	Accesses the KMS for your Hadoop cluster.
DataSync agent	Kerberos Key Distribution Center (KDC) server	TCP	KDC port (default is 88)	Authenticates with the Kerberos realm. This port is used only with HDFS and SMB locations that use Kerberos authentic ation.
DataSync agent	Storage system's management interface	ТСР	Depends on your network	Connects to your storage system.

# **Network requirements for AWS storage services**

The network ports required for DataSync to connect to an AWS storage service during a transfer vary.

From	То	Protocol	Port		
DataSync service	Amazon EFS	TCP	2049		
DataSync service	FSx for Windows File Server	See <u>file system access control for</u> <u>FSx for Windows File Server</u> .			
DataSync service	FSx for Lustre	See <u>file system acc</u> FSx for Lustre.	cess control for		
DataSync service	FSx for OpenZFS	See <u>file system acc</u> FSx for OpenZFS.	cess control for		
DataSync service	FSx for ONTAP	ТСР	111, 635, and 2049 (NFS)		
			445 (SMB)		
DataSync service	Amazon S3	N/A (DataSync connects to S3 buckets on your behalf)			

# Network requirements for public or FIPS service endpoints

Your DataSync agent requires the following network access when using public or FIPS service endpoints. If you use a firewall or router to filter or limit network traffic, configure your firewall or router to allow these endpoints.

From	То	Protocol	Port	How it's used	Endpoints accessed
Your web browser	DataSync agent	TCP	80 (HTTP)	Allows your browser to obtain the DataSync	N/A

From	То	Protocol	Port	How it's used	Endpoints accessed
				agent's activation key. Once activated , DataSync closes the agent's port 80.  Your agent doesn't require port 80 to be publicly accessible. The required level of access to port 80 depends on your network configuration.	
				(3) Note  You can get the activatio n key without a connectio n between your browser and agent.  For	

From	То	Protocol	Port	How it's used	Endpoints accessed
				more informati on, see Getting an activatio n key.	
DataSync agent	Amazon CloudFront	TCP	443 (HTTPS)	Helps bootstrap your DataSync agent prior to activation.	<ul> <li>AWS Regions:         <ul> <li>d3dvvaliwoko8h.cloudfront.net</li> </ul> </li> <li>AWS GovCloud (US) Regions:         <ul> <li>s3.us-gov-west-1.amazonaws.com/fmrsendpoints-endpoints</li> <li>bucket-go4p5gpna6sk</li> </ul> </li> </ul>
DataSync agent	AWS	TCP	443 (HTTPS)	Activates your DataSync agent and associate s it with your AWS account. You can block the public endpoint after activation.	The activation-region is the AWS Region where you activate your DataSync agent.  Public endpoint activation:  • activation.datasyn c. activation-region.amazonaws.com  FIPS endpoint activation:  • activation.datasyn c-fips. activation-region.amazonaws.com

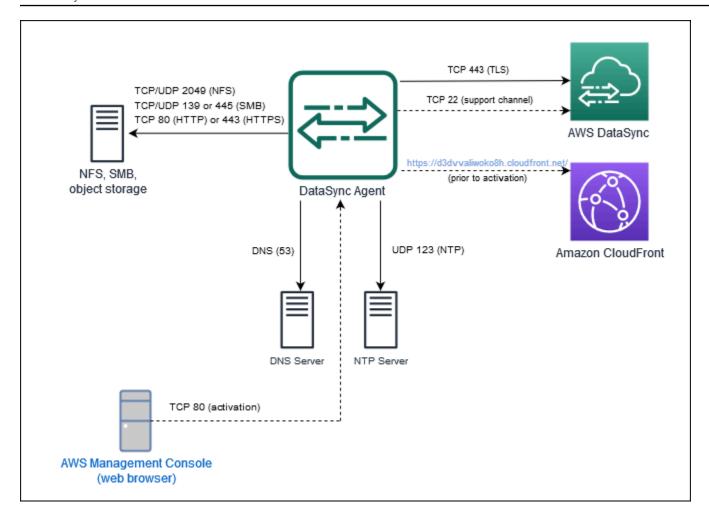
From	То	Protocol	Port	How it's used	Endpoints accessed
DataSync	AWS	TCP	443 (HTTPS)	Allows communication between the DataSync agent and DataSync service endpoint.  For informati on, see Choosing a service endpoint for your AWS DataSync agent.	The activation-region is the AWS Region where you activate your DataSync agent. Depending on what you're using DataSync for, you might not need to allow access to every endpoint listed here.  DataSync control plane endpoints:  Public endpoint: cp.datasy nc. activation-region.amazonaws.com  FIPS endpoint: cp.datasy nc-fips. activation-region.amazonaws.com  DataSync data plane endpoint (for transfer tasks only):  your-task-id .datasync-dp. activation-region.amazonaws.com

From	То	Protocol	Port	How it's used	Endpoints accessed
Your client	AWS	TCP	443 (HTTPS)	Allows you to make DataSync API requests.	The activation-region is the AWS Region where you activate your DataSync agent.
					Public endpoint:
					<ul> <li>datasync. activation- region.amazonaws.com</li> </ul>
					FIPS endpoint:
					<ul> <li>datasync-fips. activatio</li> <li>n-region .amazonaws.com</li> </ul>
DataSync	AWS	TCP	443 (HTTPS)	Allows the DataSync agent to get updates from AWS. For more information, see Managing your AWS DataSync agent.	The activation-region is the AWS Region where you activate your DataSync agent.  • amazonlinux.defaul t.amazonaws.com  • cdn.amazonlinux.com  • amazonlinux-2-repos-activatio n-region.s3.duals tack. activation-region.amazonaws.com  • amazonlinux-2-repos-activation-region.s3.activation-region.amazonaws.com  • *.s3.activation-region.amazonaws.com

From	То	Protocol	Port	How it's used	Endpoints accessed
DataSync agent	Domain Name Service (DNS) server	TCP/ UDP	53 (DNS)	Allows communication between the DataSync agent and DNS server.	N/A
DataSync agent	AWS	TCP	22 (Support channel)	Allows AWS Support to access your DataSync agent to help you troubleshoot issues. You don't need this port open for normal operation.	AWS Support channel:  • 54.201.223.107

From	То	Protocol	Port	How it's used	Endpoints accessed
DataSync	Network Time Protocol (NTP) server	UDP	123 (NTP)	Allows local systems to synchronize the VM time to the host time.	• 0.amazon.pool.ntp.org • 1.amazon.pool.ntp.org • 2.amazon.pool.ntp.org • 3.amazon.pool.ntp.org  • To change the default NTP configuration of your VM agent to use a different NTP server using the local console, see Synchronizing the time on your VMware agent.

The following diagram shows the ports required by DataSync when using public or FIPS service endpoints.



# **Network requirements for VPC service endpoints**

A virtual private cloud (VPC) endpoint provides a private connection between your agent and AWS that doesn't cross the internet or use public IP addresses. This also helps prevent packets from entering or exiting the network. For more information, see <a href="Choosing a VPC service endpoint">Choosing a VPC service endpoint</a>.

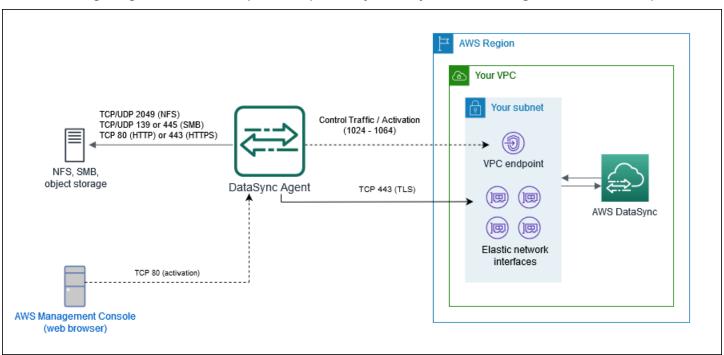
DataSync requires the following ports for your agent to use a VPC service endpoint.

From	То	Protocol	Port	How it's used
Your web browser	Your DataSync agent	TCP	80 (HTTP)	Allows your browser to obtain the agent activation key. Once activated, DataSync closes the agent's port 80.

From	То	Protocol	Port	How it's used
				Your agent doesn't require port 80 to be publicly accessible. The required level of access to port 80 depends on your network configuration.   Note  You can get the activatio n key without a connection between your browser and agent. For more information, see Getting an activation key.
DataSync agent	Your DataSync VPC service endpoint  To find the endpoint' s IP address, open the Amazon VPC console, choose Endpoints, and select your DataSync VPC service endpoint. On the Subnets tab, locate the IP address for your VPC service endpoint' s subnet. This is the endpoint's IP address.	TCP	1024-1064	For control plane traffic.

From	То	Protocol	Port	How it's used
DataSync agent	Your DataSync task's network interfaces  To find the IP addresses of these interfaces, see Viewing your network interfaces.	ТСР	443 (HTTPS)	For <u>data plane traffic</u> .
DataSync agent	Your DataSync VPC service endpoint	TCP	22 (Support channel)	To allow AWS Support to access your DataSync agent for troublesh ooting.  You don't need this port open for normal operation.

The following diagram shows the ports required by DataSync when using VPC service endpoints.



# **Network interfaces for AWS DataSync transfers**

For every task you create, AWS DataSync automatically generates and manages <u>network interfaces</u> for data transfer traffic. How many network interfaces DataSync creates and where they're created depends on the following details about your transfer task:

- Whether your task requires a DataSync agent.
- Your source and destination locations (where you're copying data from and to).
- The type of service endpoint that your agent uses.

Each network interface uses a single IP address in your subnet (the more network interfaces there are, the more IP addresses you need). Use the following tables to make sure your subnet has enough IP addresses for your task.

# **Network interfaces for transfers with agents**

In general, you need a DataSync agent when copying data between an AWS storage service and storage system that isn't AWS.

Location	Network interfaces created by default	Where network interfaces are created when using a public or FIPS endpoint	Where network interfaces are created when using a private (VPC) endpoint
Amazon S3	4	N/A <sup>1</sup>	The subnet you specify when activating your DataSync agent.
Amazon EFS	4	The subnet you specify when creating the Amazon EFS location.	
Amazon FSx for Windows File Server	4	The same subnet as the file server.	e file system's preferred

Location	Network interfaces created by default	Where network interfaces are created when using a public or FIPS endpoint	Where network interfaces are created when using a private (VPC) endpoint
Amazon FSx for Lustre	4	The same subnet as the	e file system.
Amazon FSx for OpenZFS	4	The same subnet as the	e file system.
Amazon FSx for NetApp ONTAP	4	The same subnet as the	e file system.

<sup>&</sup>lt;sup>1</sup> Network interfaces aren't needed because the DataSync service communicates directly with the S3 bucket.

# **Network interfaces for transfers without agents**

You don't need a DataSync agent when copying data between AWS services.

The total number of network interfaces depends on the DataSync locations in your transfer. For example, transferring between Amazon EFS and FSx for Lustre file systems requires four network interfaces. Meanwhile, transferring between FSx for Windows File Server and an S3 bucket requires two network interfaces.

Location	Network interfaces created by default	Where network interfaces are created
Amazon S3	N/A <sup>1</sup>	N/A <sup>1</sup>
Amazon EFS	2	The subnet you specify when creating the Amazon EFS location.

Location	Network interfaces created by default	Where network interfaces are created
FSx for Windows File Server	2	The same subnet as the preferred file server for the file system.
FSx for Lustre	2	The same subnet as the file system.
FSx for OpenZFS	2	The same subnet as the file system.
FSx for ONTAP	2	The same subnet as the file system.

<sup>&</sup>lt;sup>1</sup> Network interfaces aren't needed because the DataSync service communicates directly with the S3 bucket.

## Viewing your network interfaces

To see the network interfaces allocated to your DataSync transfer task, do one of the following:

• Use the <u>DescribeTask</u> operation. The operation returns SourceNetworkInterfaceArns and DestinationNetworkInterfaceArns with responses that look like this:

```
arn:aws:ec2:your-region:your-account-id:network-interface/eni-f012345678abcdef0
```

In this example, the network interface ID is eni-f012345678abcdef0.

• In the Amazon EC2 console, search for your task ID (such as task-f012345678abcdef0) to find its network interfaces.

# DataSync architecture and routing examples with AWS Direct Connect

Consider the following network architectures when using <u>AWS Direct Connect</u> with your AWS DataSync transfers.



#### (i) Tip

If your network uses a transit gateway, we recommend separating your DataSync transfer's logical path to optimize costs (particularly if you're migrating a large amount of data). For example, if you use AWS Transit Gateway for normal traffic between your on-premises networks and virtual private clouds (VPCs), you can configure your network so that DataSync traffic bypasses the transit gateway and its data processing charges.

## Using Direct Connect with a DataSync VPC service endpoint

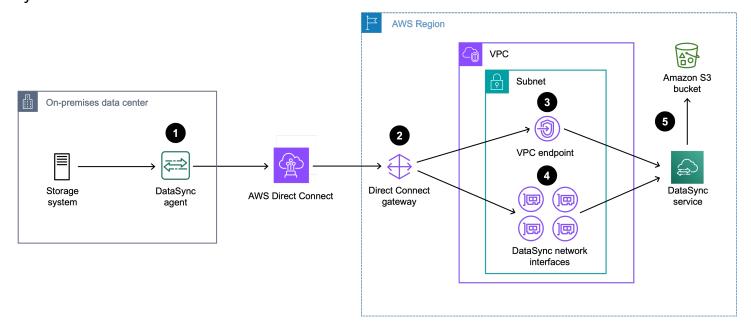
If your DataSync agent uses a VPC service endpoint, you need a Direct Connect gateway to connect to your VPC.

#### **Contents**

- Direct Connect architecture with VPC endpoint and S3 destination
- Direct Connect architecture with VPC endpoint and file system destination in same subnet
- Direct Connect architecture with VPC endpoint and file system destination in different subnets

### Direct Connect architecture with VPC endpoint and S3 destination

The following Direct Connect architecture shows a DataSync transfer from an on-premises storage system to an S3 bucket.



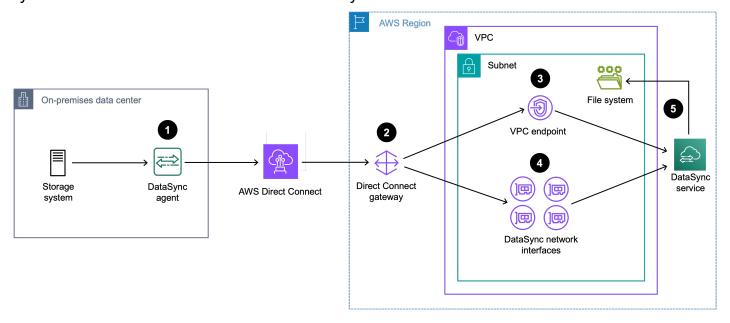
1. The DataSync agent routes DataSync traffic from the on-premises storage system (source location) to the Direct Connect connection.

- 2. DataSync traffic routes to a Direct Connect gateway that's used for your transfer. To set this up, you must:
  - a. Associate the Direct Connect gateway with a <u>virtual private gateway</u> for the VPC. This is the VPC where the DataSync VPC endpoint is located and where the DataSync task creates network interfaces.
  - b. Create a private virtual interface that connects this VPC to the Direct Connect gateway.
- 3. DataSync traffic (control plane) routes through the DataSync VPC endpoint.
- 4. DataSync traffic (data plane) routes through the DataSync network interfaces in the subnet that you specify when <u>creating the DataSync agent</u>.
- 5. DataSync traffic routes through the DataSync service to the S3 bucket (destination location).

## Direct Connect architecture with VPC endpoint and file system destination in same subnet

When transferring to or from an Amazon EFS or Amazon FSx file system, your file system and DataSync VPC endpoint can be in the same subnet.

The following Direct Connect architecture shows a DataSync transfer from an on-premises storage system to an Amazon EFS or Amazon FSx file system.



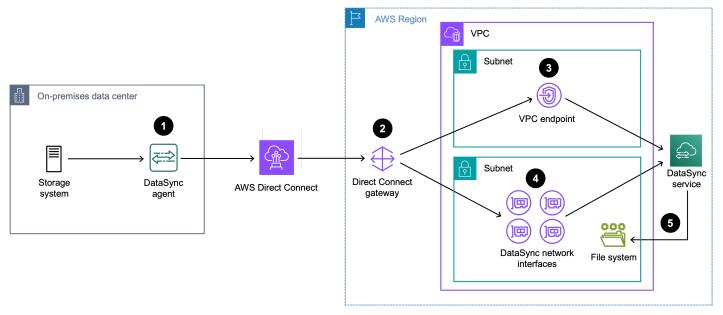
1. The DataSync agent routes DataSync traffic from the on-premises storage system (source location) to the Direct Connect connection.

- 2. DataSync traffic routes to a Direct Connect gateway that's used for your transfer. To set this up, you must:
  - a. Associate the Direct Connect gateway with a <u>virtual private gateway</u> for the VPC. This is the VPC where the DataSync VPC endpoint is located and where the DataSync task creates <u>network interfaces</u> for the file system (destination location).
  - b. Create a private virtual interface that connects this VPC to the Direct Connect gateway.
- 3. DataSync traffic (control plane) routes through the DataSync VPC endpoint.
- 4. DataSync traffic (data plane) routes through the DataSync network interfaces in the file system's subnet. This is the same subnet where the DataSync VPC endpoint is located.
- 5. DataSync traffic routes through the DataSync service to the file system (destination location).

## Direct Connect architecture with VPC endpoint and file system destination in different subnets

When transferring to or from an Amazon EFS or Amazon FSx file system, your file system and DataSync VPC endpoint can be in different subnets.

The following Direct Connect architecture shows a DataSync transfer from an on-premises storage system to an Amazon EFS or Amazon FSx file system.



1. The DataSync agent routes DataSync traffic from the on-premises storage system (source location) to the Direct Connect connection.

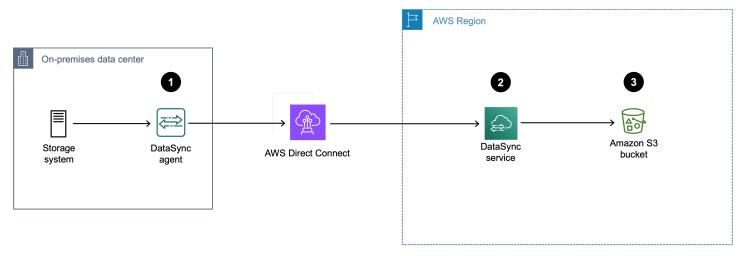
- 2. DataSync traffic routes to a Direct Connect gateway that's used for your transfer. To set this up, you must:
  - a. Associate the Direct Connect gateway with a <u>virtual private gateway</u> for the VPC. This is the VPC where the DataSync VPC endpoint is located and where the DataSync task creates network interfaces for the file system (destination location).
  - b. Create a private virtual interface that connects these VPCs to the Direct Connect gateway.
- 3. DataSync traffic (control plane) routes through the DataSync VPC endpoint.
- 4. DataSync traffic (data plane) routes through the DataSync network interfaces in the file system's subnet. This is a different subnet than where the DataSync VPC endpoint is located.
- 5. DataSync traffic routes through the DataSync service to the file system (destination location).

## Using Direct Connect with a DataSync public or FIPS service endpoint

If your DataSync agent uses a <u>public</u> or <u>Federal Information Processing Standard (FIPS)</u> service endpoint, you can route your data transfer traffic through a Direct Connect connection by using a <u>public virtual interface</u>.

While Direct Connect advertises all local and remote AWS Region prefixes by default, you can use <u>BGP community tags</u> to control the scope (Regional or global) and route preference of traffic on the public virtual interface. You must advertise at least one public prefix to create your DataSync agent.

The following Direct Connect architecture shows a DataSync transfer from an on-premises storage system through a public or FIPS endpoint to an S3 bucket.



1. The DataSync agent routes DataSync traffic from the on-premises storage system (source location) to the Direct Connect connection.

- 2. DataSync traffic routes to the DataSync service through a public virtual interface.
- 3. DataSync traffic to the S3 bucket (destination location).

### **Next steps**

If <u>you need a DataSync agent</u> and haven't created one yet, <u>deploy</u> the agent, <u>choose a service</u> endpoint for the agent, and then activate the agent.

Once you create the agent, you can configure your network for DataSync.

## Configuring your AWS DataSync agent for multiple NICs

If you configure your AWS DataSync agent to use multiple network adapters (NICs), the agent can be accessed by more than one IP address. You might want to do this in the following situations:

- Maximizing throughput You might want to maximize throughput to an agent when network adapters are a bottleneck.
- Network isolation Your Network File System (NFS), Server Message Block (SMB), Hadoop
  Distributed File System (HDFS), or object storage server might reside on a virtual LAN (VLAN)
  that lacks internet connectivity for security reasons.

In a typical multiple-adapter use case, one adapter is configured as the route by which the agent communicates with AWS (as the default agent). Except for this one adapter, NFS, SMB, HDFS, or self-managed object storage locations must be in the same subnet as the adapter that connects to them.

Otherwise, communication with the intended NFS, SMB, HDFS, or object storage locations might not be possible. In some cases, you might configure an NFS, SMB, HDFS, or object storage location on the same adapter that's used for communication with AWS. In these cases, NFS, SMB, HDFS, or object storage traffic for that server and AWS traffic flows through the same adapter.

In some cases, you might configure one adapter to connect to the AWS DataSync console and then add a second adapter. In such a case, DataSync automatically configures the route table to use the second adapter as the preferred route.

Next steps 64

## Transferring your data with AWS DataSync

With AWS DataSync, you can transfer data to or from storage that's on-premises, in AWS, in other clouds, and on the edge.

Setting up a DataSync transfer generally involves the following steps:

- 1. Determine if DataSync supports your transfer.
- 2. If <u>you need a DataSync agent</u> for your transfer, deploy and activate an agent as close as possible to one of your storage systems.

For example, if you're transferring from an on-premises Network File System (NFS) file server, deploy the agent as close as you can to that file server.

Provide DataSync access to your storage system.

DataSync needs permission to read from or write to your storage (depending on whether your storage is a source or destination location). For example, learn how to <u>provide DataSync access</u> to NFS file servers.

- 4. Connect your network for traffic between your storage system and DataSync.
- 5. Create a location for your storage system by using the DataSync console, AWS CLI, or DataSync API.

For example, learn how to <u>create an NFS location</u> or <u>Amazon S3 location</u>.

- 6. Repeat steps 3-5 to create your transfer's other location.
- 7. <u>Create and start a DataSync transfer task</u> that includes your source and destination locations.

#### **Topics**

- Where can I transfer my data with AWS DataSync?
- Transferring to or from on-premises storage with AWS DataSync
- Transferring to or from AWS storage with AWS DataSync
- Transferring to or from other cloud storage with AWS DataSync
- Transferring to or from S3 compatible storage on Snowball Edge
- Creating a task for transferring your data
- Starting a task to transfer your data

## Where can I transfer my data with AWS DataSync?

Where you can transfer your data with AWS DataSync depends on the following factors:

- Your transfer's source and destination locations
- If your locations are in different AWS accounts
- If your locations are in different AWS Regions

## Supported transfers in the same AWS account

DataSync supports transfers between the following storage resources that are associated with the same AWS account.

Source (from)	Destination (to)
<ul><li>NFS</li><li>SMB</li><li>HDFS</li><li>Object storage</li></ul>	<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>Amazon FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>
<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>	<ul><li>NFS</li><li>SMB</li><li>HDFS</li><li>Object storage</li></ul>
<ul> <li>Google Cloud Storage</li> <li>Microsoft Azure Blob Storage</li> <li>Microsoft Azure Files</li> <li>Wasabi Cloud Storage</li> <li>DigitalOcean Spaces</li> </ul>	<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>Amazon FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> </ul>

Source (from)	Destination (to)
<ul> <li>Oracle Cloud Infrastructure Object Storage</li> <li>Cloudflare R2 Storage</li> <li>Backblaze B2 Cloud Storage</li> <li>NAVER Cloud Object Storage</li> <li>Alibaba Cloud Object Storage Service</li> <li>IBM Cloud Object Storage</li> <li>Seagate Lyve Cloud</li> </ul>	FSx for ONTAP
<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>Amazon FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>	<ul> <li>Google Cloud Storage</li> <li>Microsoft Azure Blob Storage</li> <li>Microsoft Azure Files</li> <li>Wasabi Cloud Storage</li> <li>DigitalOcean Spaces</li> <li>Oracle Cloud Infrastructure Object Storage</li> <li>Cloudflare R2 Storage</li> <li>Backblaze B2 Cloud Storage</li> <li>NAVER Cloud Object Storage</li> <li>Alibaba Cloud Object Storage Service</li> <li>IBM Cloud Object Storage</li> <li>Seagate Lyve Cloud</li> </ul>
Amazon S3 compatible storage on AWS Snowball Edge	<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>Amazon FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>

Source (from)	Destination (to)
<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>	Amazon S3 compatible storage on Snowball Edge
<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>	<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>
Amazon S3 (in AWS Regions)	Amazon S3 on AWS Outposts
<ul> <li>Amazon S3 on AWS Outposts</li> </ul>	Amazon S3 (in AWS Regions)

## **Supported transfers across AWS accounts**

DataSync supports some transfers between storage resources that are associated with different AWS accounts.

Source (from)	Destination (to)
<ul> <li>Amazon EFS<sup>1</sup></li> <li>FSx for Windows File Server<sup>2</sup></li> </ul>	<ul> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>

Source (from)	Destination (to)
Amazon S3 (in AWS Regions)	<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>
<ul> <li>Amazon S3 (in AWS Regions)</li> <li>Amazon EFS</li> <li>FSx for Windows File Server</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP</li> </ul>	Amazon S3 (in AWS Regions)
<ul><li>NFS</li><li>SMB</li><li>HDFS</li><li>Object storage</li></ul>	Amazon S3 (in AWS Regions)

<sup>&</sup>lt;sup>1</sup> Configured as an NFS location.

## **Supported transfers in the same AWS Region**

There are no restrictions when transferring data within the same AWS Region (including opt-in Regions). For more information, see AWS Regions supported by DataSync.

## **Supported transfers between AWS Regions**

Note the following when transferring data between AWS Regions supported by DataSync:

<sup>&</sup>lt;sup>2</sup> Configured as an SMB location.

• When transferring between AWS storage services in different AWS Regions, one of the two locations must be in the Region where you're using DataSync.

- You can't transfer across Regions with an NFS, SMB, HDFS, or object storage location. In these
  situations, both of your transfer locations must be in the same Region where you <u>activate your</u>
  <u>DataSync agent</u>.
- With AWS GovCloud (US) Regions, you can:
  - Transfer between the AWS GovCloud (US-East) and AWS GovCloud (US-West) Regions.
  - Transfer between an AWS GovCloud (US) Region and commercial AWS Region, such as US East (N. Virginia). This type of transfer requires an <u>agent</u> when transferring between Amazon EFS or Amazon FSx file systems.

#### ▲ Important

You pay for data transferred between AWS Regions. This transfer is billed as data transfer out from the source to destination Region. For more information, see <a href="AWS DataSync">AWS DataSync</a> Pricing.

## Determining if your transfer requires a DataSync agent

Depending on your transfer scenario, you might need a DataSync agent. For more information, see Do I need an AWS DataSync agent?

# Transferring to or from on-premises storage with AWS DataSync

With AWS DataSync, you can transfer files and objects between a number of on-premises or self-managed storage systems and the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

#### **Topics**

- Configuring AWS DataSync transfers with an NFS file server
- Configuring AWS DataSync transfers with an SMB file server
- Configuring AWS DataSync transfers with an HDFS cluster
- Configuring DataSync transfers with an object storage system

## Configuring AWS DataSync transfers with an NFS file server

With AWS DataSync, you can transfer data between your Network File System (NFS) file server and one of the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

To set up this kind of transfer, you create a <u>location</u> for your NFS file server. You can use this location as a transfer source or destination.

#### **Providing DataSync access to NFS file servers**

For DataSync to access your NFS file server, you need a DataSync <u>agent</u>. The agent mounts an export on your file server by using the NFS protocol.

#### **Topics**

- Configuring your NFS export
- Supported NFS versions

#### **Configuring your NFS export**

The export that DataSync needs for your transfer depends on if your NFS file server is a source or destination location and how your file server's permissions are configured.

If your file server is a source location, DataSync just has to read and traverse your files and folders. If it's a destination location, DataSync needs root access to write to the location and set ownership, permissions, and other metadata on the files and folders that you're copying. You can use the no\_root\_squash option to allow root access for your export.

The following examples describe how to configure an NFS export that provides access to DataSync.

#### When your NFS file server is a source location (root access)

Configure your export by using the following command, which provides DataSync read-only permissions (ro) and root access (no\_root\_squash):

```
export-path datasync-agent-ip-address(ro,no_root_squash)
```

#### When your NFS file server is a destination location

Configure your export by using the following command, which provides DataSync write permissions (rw) and root access (no\_root\_squash):

```
export-path datasync-agent-ip-address(rw,no_root_squash)
```

#### When your NFS file server is a source location (no root access)

Configure your export by using the following command, which specifies the POSIX user ID (UID) and group ID (GID) that you know would provide DataSync read-only permissions on the export:

```
export-path datasync-agent-ip-address(ro,all_squash,anonuid=uid,anongid=gid)
```

#### **Supported NFS versions**

By default, DataSync uses NFS version 4.1. DataSync also supports NFS 4.0 and 3.x.

## **Configuring your network for NFS transfers**

For your DataSync transfer, you must configure traffic for a few network connections:

- 1. Allow traffic on the following ports from your DataSync agent to your NFS file server:
  - For NFS version 4.1 and 4.0 TCP port 2049
  - For NFS version 3.x TCP ports 111 and 2049

Other NFS clients in your network should be able to mount the NFS export that you're using to transfer data. The export must also be accessible without Kerberos authentication.

- 2. Configure traffic for your service endpoint connection (such as a VPC, public, or FIPS endpoint).
- 3. Allow traffic from the DataSync service to the <u>AWS storage service</u> you're transferring to or from.

#### **Creating your NFS transfer location**

Before you begin, note the following:

- You need an NFS file server that you want to transfer data from.
- You need a DataSync agent that can access your file server.
- DataSync doesn't support copying NFS version 4 access control lists (ACLs).

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For Location type, choose Network File System (NFS).
- 4. For **Agents**, choose the DataSync agent that can connect to your NFS file server.

You can choose more than one agent. For more information, see <u>Using multiple DataSync</u> <u>agents</u>.

- 5. For **NFS server**, enter the Domain Name System (DNS) name or IP address of the NFS file server that your DataSync agent connects to.
- 6. For **Mount path**, enter the NFS export path that you want DataSync to mount.
  - This path (or a subdirectory of the path) is where DataSync transfers data to or from. For more information, see Configuring your NFS export.
- 7. (Optional) Expand **Additional settings** and choose a specific **NFS version** for DataSync to use when accessing your file server.
  - For more information, see Supported NFS versions.
- 8. (Optional) Choose **Add tag** to tag your NFS location.

Tags are key-value pairs that help you manage, filter, and search for your locations. We recommend creating at least a name tag for your location.

9. Choose Create location.

#### Using the AWS CLI

Use the following command to create an NFS location.

```
aws datasync create-location-nfs \
    --server-hostname nfs-server-address \
    --on-prem-config AgentArns=datasync-agent-arns \
    --subdirectory nfs-export-path
```

For more information on creating the location, see <u>Providing DataSync access to NFS file</u> servers.

DataSync automatically chooses the NFS version that it uses to read from an NFS location. To specify an NFS version, use the optional Version parameter in the <a href="MSSMountOptions">MSSMountOptions</a> API operation.

This command returns the Amazon Resource Name (ARN) of the NFS location, similar to the ARN shown following.

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:111222333444:location/
loc-0f01451b140b2af49"
}
```

To make sure that the directory can be mounted, you can connect to any computer that has the same network configuration as your agent and run the following command.

```
mount -t nfs -o nfsvers=<nfs-server-version <nfs-server-address:<nfs-export-path <test-folder
```

The following is an example of the command.

```
mount -t nfs -o nfsvers=3 198.51.100.123:/path_for_sync_to_read_from /
temp_folder_to_test_mount_on_local_machine
```

## Configuring AWS DataSync transfers with an SMB file server

With AWS DataSync, you can transfer data between your Server Message Block (SMB) file server and one of the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

To set up this kind of transfer, you create a <u>location</u> for your SMB file server. You can use this as a transfer source or destination.

#### **Providing DataSync access to SMB file servers**

DataSync connects to your file server using the SMB protocol and can authenticate with NTLM or Kerberos.

#### **Topics**

- Supported SMB versions
- Using NTLM authentication
- Using Kerberos authentication
- Required permissions
- DFS Namespaces

#### **Supported SMB versions**

By default, DataSync automatically chooses a version of the SMB protocol based on negotiation with your SMB file server.

You also can configure DataSync to use a specific SMB version, but we recommend doing this only if DataSync has trouble negotiating with the SMB file server automatically. DataSync supports SMB versions 1.0 and later. For security reasons, we recommend using SMB version 3.0.2 or later. Earlier versions, such as SMB 1.0, contain known security vulnerabilities that attackers can exploit to compromise your data.

See the following table for a list of options in the DataSync console and API:

Console option	API option	Description
Automatic	AUTOMATIC	DataSync and the SMB file server negotiate the highest version of SMB that they mutually support between 2.1 and 3.1.1.
		This is the default and recommended option. If you instead choose a specific version that your file server doesn't support, you may get an Operation Not Supported error.
SMB 3.0.2	SMB3	Restricts the protocol negotiation to only SMB version 3.0.2.
SMB 2.1	SMB2	Restricts the protocol negotiation to only SMB version 2.1.
SMB 2.0	SMB2_0	Restricts the protocol negotiation to only SMB version 2.0.
SMB 1.0	SMB1	Restricts the protocol negotiation to only SMB version 1.0.

#### **Using NTLM authentication**

To use NTLM authentication, you provide a user name and password that allows DataSync to access the SMB file server that you're transferring to or from. The user can be a local user on your file server or a domain user in your Microsoft Active Directory.

#### **Using Kerberos authentication**

To use Kerberos authentication, you provide a Kerberos principal, Kerberos key table (keytab) file, and Kerberos configuration file that allows DataSync to access the SMB file server that you're transferring to or from.

#### **Topics**

- Prerequisites
- DataSync configuration options for Kerberos

#### **Prerequisites**

You need to create a couple Kerberos artifacts and configure your network so that DataSync can access your SMB file server.

Create a Kerberos keytab file by using the ktpass or kutil utility.

The following example creates a keytab file by using ktpass. The Kerberos realm that you specify (MYDOMAIN.ORG) must be upper case.

```
ktpass /out C:\YOUR_KEYTAB.keytab /princ HOST/kerberosuser@MYDOMAIN.ORG /mapuser kerberosuser /pass * /crypto AES256-SHA1 /ptype KRB5_NT_PRINCIPAL
```

 Prepare a simplified version of the Kerberos configuration file (krb5.conf). Include information about the realm, the location of the domain admin servers, and mappings of hostnames onto a Kerberos realm.

Verify that the krb5.conf content is formatted with the correct mixed casing for the realms and domain realm names. For example:

```
[libdefaults]
  dns_lookup_realm = true
  dns_lookup_kdc = true
  forwardable = true
  default_realm = MYDOMAIN.ORG

[realms]
  MYDOMAIN.ORG = {
    kdc = mydomain.org
    admin_server = mydomain.org
```

```
}
[domain_realm]
.mydomain.org = MYDOMAIN.ORG
mydomain.org = MYDOMAIN.ORG
```

• In your network configuration, make sure that your Kerberos Key Distribution Center (KDC) server port is open. The KDC port is typically TCP port 88.

#### **DataSync configuration options for Kerberos**

When creating an SMB location that uses Kerberos, you configure the following options.

Console option	API option	Description
SMB server	ServerHostName	The domain name of the SMB file server that your DataSync agent will mount. For Kerberos, you can't specify the file server's IP address.
Kerberos principal	KerberosPrincipal	An identity in your Kerberos realm that has permission to access the files, folders, and file metadata in your SMB file server.  A Kerberos principal might look like HOST/kerb erosuser@MYDOMAIN.  ORG .  Principal names are case sensitive.
Keytab file	KerberosKeytab	A Kerberos key table (keytab) file, which includes mappings

Console option	API option	Description
		between your Kerberos principal and encryption keys.
Kerberos configuration file	KerberosKrbConf	A krb5.conf file that defines your Kerberos realm configuration.
DNS IP addresses (optional)	DnsIpAddresses	The IPv4 addresses for the DNS servers that your SMB file server belongs to.  If you have multiple domains in your environment, configuring this makes sure that DataSync connects to the right SMB file server.

#### **Required permissions**

The identity that you provide DataSync must have permission to mount and access your SMB file server's files, folders, and file metadata.

If you provide an identity in your Active Directory, it must be a member of an Active Directory group with one or both of the following user rights (depending the <u>metadata that you want</u> DataSync to copy):

User right	Description
<b>Restore files and directories (</b> SE_RESTOR E_NAME )	Allows DataSync to copy object ownership, permissions, file metadata, and NTFS discretio nary access lists (DACLs).
	This user right is usually granted to members of the <b>Domain Admins</b> and <b>Backup Operators</b> groups (both of which are default Active Directory groups).

User right	Description
Manage auditing and security log (SE_SECURITY_NAME )	Allows DataSync to copy NTFS system access control lists (SACLs).
	This user right is usually granted to members of the <b>Domain Admins</b> group.

If you want to copy Windows ACLs and are transferring between an SMB file server and another storage system that uses SMB (such as Amazon FSx for Windows File Server or FSx for ONTAP), the identity that you provide DataSync must belong to the same Active Directory domain or have an Active Directory trust relationship between their domains.

#### **DFS Namespaces**

DataSync doesn't support Microsoft Distributed File System (DFS) Namespaces. We recommend specifying an underlying file server or share instead when creating your DataSync location.

#### **Creating your SMB transfer location**

Before you begin, you need an SMB file server that you want to transfer data from.

#### Using the DataSync console

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For Location type, choose Server Message Block (SMB).

You configure this location as a source or destination later.

4. For **Agents**, choose the DataSync agent that can connect to your SMB file server.

You can choose more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

For SMB server, enter the domain name or IP address of the SMB file server that your DataSync agent will mount.

Remember the following with this setting:

• You can't specify an IP version 6 (IPv6) address.

- If you're using Kerberos authentication, you must specify a domain name.
- 6. For **Share name**, enter the name of the share exported by your SMB file server where DataSync will read or write data.

You can include a subdirectory in the share path (for example, /path/to/subdirectory). Make sure that other SMB clients in your network can also mount this path.

To copy all the data in the subdirectory, DataSync must be able to mount the SMB share and access all of its data. For more information, see Required permissions.

 (Optional) Expand Additional settings and choose an SMB Version for DataSync to use when accessing your file server.

By default, DataSync automatically chooses a version based on negotiation with the SMB file server. For information, see Supported SMB versions.

- 8. For **Authentication type**, choose **NTLM** or **Kerberos**.
- 9. Do one of the following depending on your authentication type:

#### NTLM

• For **User**, enter a user name that can mount your SMB file server and has permission to access the files and folders involved in your transfer.

For more information, see Required permissions.

- For **Password**, enter the password of the user who can mount your SMB file server and has permission to access the files and folders involved in your transfer.
- (Optional) For **Domain**, enter the Windows domain name that your SMB file server belongs to.

If you have multiple domains in your environment, configuring this setting makes sure that DataSync connects to the right SMB file server.

#### Kerberos

• For **Kerberos principal**, specify a principal in your Kerberos realm that has permission to access the files, folders, and file metadata in your SMB file server.

A Kerberos principal might look like HOST/kerberosuser@MYDOMAIN.ORG.

Principal names are case sensitive. Your DataSync task execution will fail if the principal that you specify for this setting doesn't exactly match the principal that you use to create the keytab file.

- For **Keytab file**, upload a keytab file that includes mappings between your Kerberos principal and encryption keys.
- For **Kerberos configuration file**, upload a krb5.conf file that defines your Kerberos realm configuration.
- (Optional) For **DNS IP addresses**, specify up to two IPv4 addresses for the DNS servers that your SMB file server belongs to.

If you have multiple domains in your environment, configuring this parameter makes sure that DataSync connects to the right SMB file server.

10. (Optional) Choose **Add tag** to tag your SMB location.

Tags are key-value pairs that help you manage, filter, and search for your locations. We recommend creating at least a name tag for your location.

11. Choose Create location.

#### Using the AWS CLI

The following instructions describe how to create SMB locations with NTLM or Kerberos authentication.

#### **NTLM**

1. Copy the following create-location-smb command.

```
aws datasync create-location-smb \
    --agent-arns datasync-agent-arns \
    --server-hostname smb-server-address \
    --subdirectory smb-export-path \
    --authentication-type "NTLM" \
    --user user-who-can-mount-share \
    --password user-password \
    --domain windows-domain-of-smb-server
```

2. For --agent-arns, specify the DataSync agent that can connect to your SMB file server.

You can choose more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

- 3. For --server-hostname, specify the domain name or IPv4 address of the SMB file server that your DataSync agent will mount.
- 4. For --subdirectory, specify the name of the share exported by your SMB file server where DataSync will read or write data.

You can include a subdirectory in the share path (for example, /path/to/subdirectory). Make sure that other SMB clients in your network can also mount this path.

To copy all the data in the subdirectory, DataSync must be able to mount the SMB share and access all of its data. For more information, see Required permissions.

5. For --user, specify a user name that can mount your SMB file server and has permission to access the files and folders involved in your transfer.

For more information, see Required permissions.

- 6. For --password, specify the password of the user who can mount your SMB file server and has permission to access the files and folders involved in your transfer.
- 7. (Optional) For --domain, specify the Windows domain name that your SMB file server belongs to.
  - If you have multiple domains in your environment, configuring this setting makes sure that DataSync connects to the right SMB file server.
- 8. (Optional) Add the --version option if you want DataSync to use a specific SMB version. For more information, see Supported SMB versions.
- 9. Run the create-location-smb command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "arn:aws:datasync:us-east-1:123456789012:location/loc-01234567890example"
}
```

#### Kerberos

Copy the following create-location-smb command.

```
aws datasync create-location-smb \
    --agent-arns datasync-agent-arns \
    --server-hostname smb-server-address \
    --subdirectory smb-export-path \
    --authentication-type "KERBEROS" \
    --kerberos-principal "HOST/kerberosuser@EXAMPLE.COM" \
    --kerberos-keytab "fileb://path/to/file.keytab" \
    --kerberos-krb5-conf "file://path/to/krb5.conf" \
    --dns-ip-addresses array-of-ipv4-addresses
```

2. For --agent-arns, specify the DataSync agent that can connect to your SMB file server.

You can choose more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

- 3. For --server-hostname, specify the domain name of the SMB file server that your DataSync agent will mount.
- 4. For --subdirectory, specify the name of the share exported by your SMB file server where DataSync will read or write data.

You can include a subdirectory in the share path (for example, /path/to/subdirectory). Make sure that other SMB clients in your network can also mount this path.

To copy all the data in the subdirectory, DataSync must be able to mount the SMB share and access all of its data. For more information, see <u>Required permissions</u>.

- 5. For the Kerberos options, do the following:
  - --kerberos-principal: Specify a principal in your Kerberos realm that has permission to access the files, folders, and file metadata in your SMB file server.

A Kerberos principal might look like HOST/kerberosuser@MYDOMAIN.ORG.

Principal names are case sensitive. Your DataSync task execution will fail if the principal that you specify for this option doesn't exactly match the principal that you use to create the keytab file.

 --kerberos-keytab: Specify a keytab file that includes mappings between your Kerberos principal and encryption keys.

 --kerberos-krb5-conf: Specify a krb5.conf file that defines your Kerberos realm configuration.

• (Optional) --dns-ip-addresses: Specify up to two IPv4 addresses for the DNS servers that your SMB file server belongs to.

If you have multiple domains in your environment, configuring this parameter makes sure that DataSync connects to the right SMB file server.

- 6. (Optional) Add the --version option if you want DataSync to use a specific SMB version. For more information, see Supported SMB versions.
- 7. Run the create-location-smb command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "arn:aws:datasync:us-east-1:123456789012:location/loc-01234567890example"
}
```

## Configuring AWS DataSync transfers with an HDFS cluster

With AWS DataSync, you can transfer data between your Hadoop Distributed File System (HDFS) cluster and one of the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

To set up this kind of transfer, you create a <u>location</u> for your HDFS cluster. You can use this location as a transfer source or destination.

#### **Providing DataSync access to HDFS clusters**

To connect to your HDFS cluster, DataSync uses an <u>agent that you deploy</u> as close as possible to your HDFS cluster. The DataSync agent acts as an HDFS client and communicates with the NameNodes and DataNodes in your cluster.

When you start a transfer task, DataSync queries the NameNode for locations of files and folders on the cluster. If you configure your HDFS location as a source location, DataSync reads files and folder data from the DataNodes in your cluster and copies that data to the destination. If you configure your HDFS location as a destination location, then DataSync writes files and folders from the source to the DataNodes in your cluster.

#### **Authentication**

When connecting to an HDFS cluster, DataSync supports simple authentication or Kerberos authentication. To use simple authentication, provide the user name of a user with rights to read and write to the HDFS cluster. To use Kerberos authentication, provide a Kerberos configuration file, a Kerberos key table (keytab) file, and a Kerberos principal name. The credentials of the Kerberos principal must be in the provided keytab file.

#### **Encryption**

When using Kerberos authentication, DataSync supports encryption of data as it's transmitted between the DataSync agent and your HDFS cluster. Encrypt your data by using the Quality of Protection (QOP) configuration settings on your HDFS cluster and by specifying the QOP settings when creating your HDFS location. The QOP configuration includes settings for data transfer protection and Remote Procedure Call (RPC) protection.

#### DataSync supports the following Kerberos encryption types:

- des-cbc-crc
- des-cbc-md4
- des-cbc-md5
- des3-cbc-sha1
- arcfour-hmac
- arcfour-hmac-exp
- aes128-cts-hmac-sha1-96
- aes256-cts-hmac-sha1-96

- aes128-cts-hmac-sha256-128
- aes256-cts-hmac-sha384-192
- camellia128-cts-cmac
- camellia256-cts-cmac

You can also configure HDFS clusters for encryption at rest using Transparent Data Encryption (TDE). When using simple authentication, DataSync reads and writes to TDE-enabled clusters. If you're using DataSync to copy data to a TDE-enabled cluster, first configure the encryption zones on the HDFS cluster. DataSync doesn't create encryption zones.

#### **Unsupported HDFS features**

The following HDFS capabilities aren't currently supported by DataSync:

- Transparent Data Encryption (TDE) when using Kerberos authentication
- Configuring multiple NameNodes
- Hadoop HDFS over HTTP (HttpFS)
- POSIX access control lists (ACLs)
- HDFS extended attributes (xattrs)
- HDFS clusters using Apache HBase

## **Creating your HDFS transfer location**

You can use your location as a source or destination for your DataSync transfer.

**Before you begin**: Verify network connectivity between your agent and Hadoop cluster by doing the following:

- Test access to the TCP ports listed in <u>Network requirements for on-premises</u>, self-managed, other cloud, and edge storage.
- Test access between your local agent and your Hadoop cluster. For instructions, see <u>Verifying</u> your agent's connection to your storage system.

#### Using the DataSync console

Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.

2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.

3. For Location type, choose Hadoop Distributed File System (HDFS).

You can configure this location as a source or destination later.

4. For **Agents**, choose the agent that can connect to your HDFS cluster.

You can choose more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

- 5. For **NameNode**, provide the domain name or IP address of your HDFS cluster's primary NameNode.
- 6. For **Folder**, enter a folder on your HDFS cluster that you want DataSync to use for the data transfer.

If your HDFS location is a source, DataSync copies the files in this folder to the destination. If your location is a destination, DataSync writes files to this folder.

7. To set the **Block size** or **Replication factor**, choose **Additional settings**.

The default block size is 128 MiB. The block sizes that you provide must be a multiple of 512 bytes.

The default replication factor is three DataNodes when transferring to the HDFS cluster.

- 8. In the **Security** section, choose the **Authentication type** used on your HDFS cluster.
  - **Simple** For **User**, specify the user name with the following permissions on the HDFS cluster (depending on your use case):
    - If you plan to use this location as a source location, specify a user that only has read permissions.
    - If you plan to use this location as a destination location, specify a user that has read and write permissions.

Optionally, specify the URI of the Key Management Server (KMS) of your HDFS cluster.

- Kerberos Specify the Kerberos Principal with access to your HDFS cluster. Next, provide
  the KeyTab file that contains the provided Kerberos principal. Then, provide the Kerberos
  configuration file. Finally, specify the type of encryption in transit protection in the RPC
  protection and Data transfer protection dropdown lists.
- 9. (Optional) Choose **Add tag** to tag your HDFS location.

Tags are key-value pairs that help you manage, filter, and search for your locations. We recommend creating at least a name tag for your location.

10. Choose Create location.

#### Using the AWS CLI

Copy the following create-location-hdfs command.

```
aws datasync create-location-hdfs --name-nodes [{"Hostname":"host1", "Port": 8020}]

--authentication-type "SIMPLE|KERBEROS" \
--agent-arns [arn:aws:datasync:us-east-1:123456789012:agent/
agent-01234567890example] \
--subdirectory "/path/to/my/data"
```

- 2. For the --name-nodes parameter, specify the hostname or IP address of your HDFS cluster's primary NameNode and the TCP port that the NameNode is listening on.
- 3. For the --authentication-type parameter, specify the type of authentication to use when connecting to the Hadoop cluster. You can specify SIMPLE or KERBEROS.
  - If you use SIMPLE authentication, use the --simple-user parameter to specify the user name of the user. If you use KERBEROS authentication, use the --kerberos-principal, --kerberos-keytab, and --kerberos-krb5-conf parameters. For more information, see create-location-hdfs.
- 4. For the --agent-arns parameter, specify the ARN of the DataSync agent that can connect to your HDFS cluster.
  - You can choose more than one agent. For more information, see <u>Using multiple DataSync</u> <u>agents</u>.
- 5. (Optional) For the --subdirectory parameter, specify a folder on your HDFS cluster that you want DataSync to use for the data transfer.
  - If your HDFS location is a source, DataSync copies the files in this folder to the destination. If your location is a destination, DataSync writes files to this folder.
- Run the create-location-hdfs command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "arn:aws:datasync:us-east-1:123456789012:location/loc-01234567890example"
}
```

## Configuring DataSync transfers with an object storage system

With AWS DataSync, you can transfer data between your object storage system and one of the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

To set up this kind of transfer, you create a <u>location</u> for your object storage system. You can use this location as a transfer source or destination. Transferring data to or from your on-premises object storage requires a DataSync agent.

## **Prerequisites**

Your object storage system must be compatible with the following <u>Amazon S3 API operations</u> for DataSync to connect to it:

- AbortMultipartUpload
- CompleteMultipartUpload
- CopyObject
- CreateMultipartUpload
- DeleteObject
- DeleteObjects

- DeleteObjectTagging
- GetBucketLocation
- GetObject
- GetObjectTagging
- HeadBucket
- HeadObject
- ListObjectsV2
- PutObject
- PutObjectTagging
- UploadPart

#### **Creating your object storage transfer location**

Before you begin, you need an object storage system that you plan to transfer data to or from.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For **Location type**, choose **Object storage**.

You configure this location as a source or destination later.

- 4. For **Server**, provide the domain name or IP address of the object storage server.
- 5. For **Bucket name**, enter the name of the object storage bucket involved in the transfer.
- 6. For **Folder**, enter an object prefix.

DataSync only copies objects with this prefix.

- 7. If your transfer requires an agent, choose **Use agents**, then choose the DataSync agent that connects to your object storage system.
  - Some transfers don't require agents. In other scenarios, you might want to use more than one agent. For more information, see <u>Situations when you don't need a DataSync agent</u> and <u>Using multiple DataSync agents</u>.
- 8. To configure the connection to the object storage server, expand **Additional settings** and do the following:

- a. For **Server protocol**, choose **HTTP** or **HTTPS**.
- b. For **Server port**, use a default port (**80** for HTTP or **443** for HTTPS) or specify a custom port if needed.

c. For **Certificate**, if your object storage system uses a private or self-signed certificate authority (CA), select **Choose file** and specify a single .pem file with a full certificate chain.

The certificate chain might include:

- The object storage system's certificate
- All intermediate certificates (if there are any)
- The root certificate of the signing CA

You can concatenate your certificates into a .pem file (which can be up to 32768 bytes before base64 encoding). The following example cat command creates an <code>object\_storage\_certificates.pem</code> file that includes three certificates:

```
cat object_server_certificate.pem intermediate_certificate.pem ca_root_certificate.pem
> object_storage_certificates.pem
```

9. If the object storage server requires credentials for access, select **Requires credentials** and enter the **Access key** you use to access the bucket. Then either enter the **Secret key** directly, or specify an AWS Secrets Manager secret that contains the key. For more information, see Providing credentials for storage locations.

The access key and secret key can be a user name and password, respectively.

10. (Optional) Choose **Add tag** to tag your object storage location.

Tags are key-value pairs that help you manage, filter, and search for your locations. We recommend creating at least a name tag for your location.

11. Choose Create location.

#### Using the AWS CLI

Copy the following create-location-object-storage command:

```
aws datasync create-location-object-storage \
    --server-hostname object-storage-server.example.com \
```

```
--bucket-name your-bucket \
--agent-arns arn:aws:datasync:us-east-1:123456789012:agent/
agent-01234567890deadfb
```

- 2. Specify the following required parameters in the command:
  - --server-hostname Specify the domain name or IP address of your object storage server.
  - --bucket-name Specify the name of the bucket on your object storage server that you're transferring to or from.
- 3. (Optional) Add any of the following parameters to the command:
  - --agent-arns Specify the DataSync agents that you want to connect to your object storage server.
  - --server-port Specifies the port that your object storage server accepts inbound network traffic on (for example, port 443).
  - --server-protocol Specifies the protocol (HTTP or HTTPS) which your object storage server uses to communicate.
  - --access-key Specifies the access key (for example, a user name) if credentials are required to authenticate with the object storage server.
  - --secret-key Specifies the secret key (for example, a password) if credentials are required to authenticate with the object storage server.

You can also provide additional parameters for securing your keys using AWS Secrets Manager. For more information, see <u>Providing credentials for storage locations</u>.

 --server-certificate – Specifies a certificate chain for DataSync to authenticate with your object storage system if the system uses a private or self-signed certificate authority (CA). You must specify a single .pem file with a full certificate chain (for example, file:///home/user/.ssh/object\_storage\_certificates.pem).

The certificate chain might include:

- The object storage system's certificate
- All intermediate certificates (if there are any)
- The root certificate of the signing CA

You can concatenate your certificates into a .pem file (which can be up to 32768 bytes before base64 encoding). The following example cat command creates an <code>object\_storage\_certificates</code>.pem file that includes three certificates:

```
cat object_server_certificate.pem intermediate_certificate.pem ca_root_certificate.pem
> object_storage_certificates.pem
```

• --subdirectory - Specifies the object prefix for your object storage server.

DataSync only copies objects with this prefix.

 --tags – Specifies the key-value pair that represents a tag that you want to add to the location resource.

Tags can help you manage, filter, and search for your resources. We recommend creating a name tag for your location.

4. Run the create-location-object-storage command.

You get a response that shows you the location ARN that you just created.

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:123456789012:location/
loc-01234567890abcdef"
}
```

## Transferring to or from AWS storage with AWS DataSync

With AWS DataSync, you can transfer data to or from a number of AWS storage services. For more information, see Where can I transfer my data with DataSync?

#### **Topics**

- Configuring AWS DataSync transfers with Amazon S3
- Configuring AWS DataSync transfers with Amazon EFS
- Configuring transfers with FSx for Windows File Server
- Configuring DataSync transfers with FSx for Lustre
- Configuring DataSync transfers with Amazon FSx for OpenZFS
- Configuring transfers with Amazon FSx for NetApp ONTAP

# **Configuring AWS DataSync transfers with Amazon S3**

To transfer data to or from your Amazon S3 bucket, you create an AWS DataSync transfer *location*. DataSync can use this location as a source or destination for transferring data.

## **Providing DataSync access to S3 buckets**

DataSync needs access to the S3 bucket that you're transferring to or from. To do this, you must create an AWS Identity and Access Management (IAM) role that DataSync assumes with the permissions required to access the bucket. You then specify this role when <u>creating your Amazon S3 location for DataSync</u>.

#### **Contents**

- Required permissions
- Creating an IAM role for DataSync to access your Amazon S3 location
- Accessing S3 buckets using server-side encryption
- Accessing restricted S3 buckets
- Accessing S3 buckets with restricted VPC access

## Required permissions

The permissions that your IAM role needs can depend on whether bucket is a DataSync source or destination location. Amazon S3 on Outposts requires a different set of permissions.

Amazon S3 (source location)

## Amazon S3 (destination location)

```
{
 "Version": "2012-10-17",
 "Statement": [
     {
         "Action": [
             "s3:GetBucketLocation",
             "s3:ListBucket",
             "s3:ListBucketMultipartUploads"
         ],
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
         "Condition": {
             "StringEquals": {
                 "aws:ResourceAccount": "123456789012"
             }
         }
     },
         "Action": [
             "s3:AbortMultipartUpload",
             "s3:DeleteObject",
             "s3:GetObject",
             "s3:GetObjectTagging",
             "s3:GetObjectVersion",
             "s3:GetObjectVersionTagging",
             "s3:ListMultipartUploadParts",
             "s3:PutObject",
```

#### Amazon S3 on Outposts

```
{
            "Version": "2012-10-17",
            "Statement": [{
                    "Action": [
                        "s3-outposts:ListBucket",
                        "s3-outposts:ListBucketMultipartUploads"
                    ],
                    "Effect": "Allow",
                    "Resource": [
                        "arn:aws:s3-outposts:region:account-id:outpost/outpost-
id/bucket/amzn-s3-demo-bucket",
                        "arn:aws:s3-outposts:region:account-id:outpost/outpost-
id/accesspoint/bucket-access-point-name"
                },
                {
                    "Action": [
                        "s3-outposts:AbortMultipartUpload",
                        "s3-outposts:DeleteObject",
                        "s3-outposts:GetObject",
                        "s3-outposts:GetObjectTagging",
                        "s3-outposts:GetObjectVersion",
                        "s3-outposts:GetObjectVersionTagging",
                        "s3-outposts:ListMultipartUploadParts",
                        "s3-outposts:PutObject",
                        "s3-outposts:PutObjectTagging"
                    ],
                    "Effect": "Allow",
```

#### Creating an IAM role for DataSync to access your Amazon S3 location

When <u>creating your Amazon S3 location</u> in the console, DataSync can automatically create and assume an IAM role that normally has the right permissions to access your S3 bucket.

In some situations, you might need to create this role manually (for example, accessing buckets with extra layers of security or transferring to or from a bucket in a different AWS accounts).

#### Manually creating an IAM role for DataSync

- 1. Open the IAM console at https://console.aws.amazon.com/iam/.
- 2. In the left navigation pane, under **Access management**, choose **Roles**, and then choose **Create role**.
- 3. On the **Select trusted entity** page, for **Trusted entity type**, choose **AWS service**.
- 4. For **Use case**, choose **DataSync** in the dropdown list and select **DataSync**. Choose **Next**.
- 5. On the **Add permissions** page, choose **Next**. Give your role a name and choose **Create role**.
- 6. On the **Roles** page, search for the role that you just created and choose its name.
- 7. On the role's details page, choose the **Permissions** tab. Choose **Add permissions** then **Create** inline policy.
- 8. Choose the **JSON** tab and <u>add the permissions required</u> to access your bucket into the policy editor.
- 9. Choose **Next**. Give your policy a name and choose **Create policy**.

10. (Recommended) To prevent the cross-service confused deputy problem, do the following:

- a. On the role's details page, choose the **Trust relationships** tab. Choose **Edit trust policy**.
- b. Update the trust policy by using the following example, which includes the aws:SourceArn and aws:SourceAccount global condition context keys:

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": "datasync.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                 "aws:SourceAccount": "account-id"
            },
            "StringLike": {
                 "aws:SourceArn": "arn:aws:datasync:region:account-id:*"
            }
        }
    }]
}
```

c. Choose **Update policy**.

You can specify this role when creating your Amazon S3 location.

## Accessing S3 buckets using server-side encryption

DataSync can transfer data to or from <u>S3 buckets that use server-side encryption</u>. The type of encryption key a bucket uses can determine if you need a custom policy allowing DataSync to access the bucket.

When using DataSync with S3 buckets that use server-side encryption, remember the following:

• If your S3 bucket is encrypted with an AWS managed key – DataSync can access the bucket's objects by default if all your resources are in the same AWS account.

If your S3 bucket is encrypted with a customer managed AWS Key Management Service (AWS KMS) key (SSE-KMS) – The key's policy must include the IAM role that DataSync uses to access the bucket.

- If your S3 bucket is encrypted with a customer managed SSE-KMS key and in a different AWS account DataSync needs permission to access the bucket in the other AWS account. You can set up this up by doing the following:
  - In the IAM role that DataSync uses, you must specify the cross-account bucket's SSE-KMS key by using the key's fully qualified Amazon Resource Name (ARN). This is the same key ARN that you use to configure the bucket's <u>default encryption</u>. You can't specify a key ID, alias name, or alias ARN in this situation.

Here's an example key ARN:

```
arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

For more information on specifying KMS keys in IAM policy statements, see the <u>AWS Key</u> Management Service Developer Guide.

- In the SSE-KMS key policy, specify the IAM role used by DataSync.
- If your S3 bucket is encrypted with a customer managed AWS KMS key (DSSE-KMS) for dual-layer server-side encryption The key's policy must include the IAM role that DataSync uses to access the bucket. (Keep in mind that DSSE-KMS doesn't support S3 Bucket Keys, which can reduce AWS KMS request costs.)
- If your S3 bucket is encrypted with a customer-provided encryption key (SSE-C) DataSync can't access this bucket.

## **Example: SSE-KMS key policy for DataSync**

The following example is a <u>key policy</u> for a customer-managed SSE-KMS key. The policy is associated with an S3 bucket that uses server-side encryption.

If you want to use this example, replace the following values with your own:

- account-id Your AWS account.
- admin-role-name The name of the IAM role that can administer the key.
- datasync-role-name The name of the IAM role that allows DataSync to use the key when
  accessing the bucket.

#### **JSON**

```
}
    "Id": "key-consolepolicy-3",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable IAM Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Allow access for Key Administrators",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:role/admin-role-name"
            },
            "Action": [
                "kms:Create*",
                "kms:Describe*",
                "kms:Enable*",
                "kms:List*",
                "kms:Put*",
                "kms:Update*",
                "kms:Revoke*",
                "kms:Disable*",
                "kms:Get*",
                "kms:Delete*",
                "kms:TagResource",
                "kms:UntagResource",
                "kms:ScheduleKeyDeletion",
                "kms:CancelKeyDeletion"
            ],
            "Resource": "*"
        },
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
```

#### **Accessing restricted S3 buckets**

If you need to transfer to or from an S3 bucket that typically denies all access, you can edit the bucket policy so that DataSync can access the bucket only for your transfer.

## Example: Allowing access based on IAM roles

Copy the following S3 bucket policy.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "Deny-access-to-bucket",
        "Effect": "Deny",
        "Principal": "*",
        "Action": "s3:*",
        "Resource": [
            "arn:aws:s3:::amzn-s3-demo-bucket",
            "arn:aws:s3:::amzn-s3-demo-bucket/*"
        ],
        "Condition": {
            "StringNotLike": {
                "aws:userid": [
                    "datasync-iam-role-id:*",
                    "your-iam-role-id"
                ]
            }
```

```
}
}]
```

- 2. In the policy, replace the following values:
  - amzn-s3-demo-bucket Specify the name of the restricted S3 bucket.
  - datasync-iam-role-id Specify the ID of the IAM role that DataSync uses to access the bucket.

Run the following AWS CLI command to get the IAM role ID:

```
aws iam get-role --role-name datasync-iam-role-name
```

In the output, look for the RoleId value:

```
"RoleId": "ANPAJ2UCCR6DPCEXAMPLE"
```

your-iam-role-id – Specify the ID of the IAM role that you use to create your DataSync location for the bucket.

Run the following command to get the IAM role ID:

```
aws iam get-role --role-name your-iam-role-name
```

In the output, look for the RoleId value:

```
"RoleId": "AIDACKCEVSQ6C2EXAMPLE"
```

- 3. Add this policy to your S3 bucket policy.
- 4. When you're done using DataSync with the restricted bucket, remove the conditions for both IAM roles from the bucket policy.

#### Accessing S3 buckets with restricted VPC access

An Amazon S3 bucket that <u>limits access to specific virtual private cloud (VPC) endpoints or VPCs</u> will deny DataSync from transferring to or from that bucket. To enable transfers in these situations, you can update the bucket's policy to include the IAM role that you <u>specify with your DataSync</u> location.

## Option 1: Allowing access based on DataSync location role ARN

In the S3 bucket policy, you can specify the Amazon Resource Name (ARN) of your DataSync location IAM role.

The following example is an S3 bucket policy that denies access from all but two VPCs (vpc-1234567890abcdef0 and vpc-abcdef01234567890). However, the policy also includes the <a href="mailto:ArnNotLikeIfExists">ArnNotLikeIfExists</a> condition and <a href="mailto:aws:PrincipalArn">aws:PrincipalArn</a> condition key, which allow the ARN of a DataSync location role to access the bucket.

```
}
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Access-to-specific-VPCs-only",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "StringNotEqualsIfExists": {
                     "aws:SourceVpc": [
                         "vpc-1234567890abcdef0",
                         "vpc-abcdef01234567890"
                     ]
                },
                "ArnNotLikeIfExists": {
                     "aws:PrincipalArn": [
                         "arn:aws:iam::account-id:role/datasync-location-role-
name"
                    ]
                }
            }
        }
    ]
}
```

Option 2: Allowing access based on DataSync location role tag

In the S3 bucket policy, you can specify a tag attached to your DataSync location IAM role.

The following example is an S3 bucket policy that denies access from all but two VPCs (vpc-1234567890abcdef0 and vpc-abcdef01234567890). However, the policy also includes the <a href="StringNotEqualsIfExists">StringNotEqualsIfExists</a> condition and <a href="aws:PrincipalTag">aws:PrincipalTag</a> condition key, which allow a principal with the tag key exclude-from-vpc-restriction and value true. You can try a similar approach in your bucket policy by specifying a tag attached to your DataSync location role.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Access-to-specific-VPCs-only",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "StringNotEqualsIfExists": {
                     "aws:SourceVpc": [
                         "vpc-1234567890abcdef0",
                         "vpc-abcdef01234567890"
                    ],
                     "aws:PrincipalTag/exclude-from-vpc-restriction": "true"
                }
            }
        }
    ]
}
```

## Storage class considerations with Amazon S3 transfers

When Amazon S3 is your destination location, DataSync can transfer your data directly into a specific Amazon S3 storage class.

Some storage classes have behaviors that can affect your Amazon S3 storage costs. When using storage classes that can incur additional charges for overwriting, deleting, or retrieving objects, changes to object data or metadata result in such charges. For more information, see <a href="Mazon S3"><u>Amazon S3</u></a> pricing.

## ▲ Important

New objects transferred to your Amazon S3 destination location are stored using the storage class that you specify when creating your location.

By default, DataSync preserves the storage class of existing objects in your destination location unless you configure your task to transfer all data. In those situations, the storage class that you specify when creating your location is used for all objects.

Amazon S3 storage class	Considerations
S3 Standard	Choose S3 Standard to store your frequently accessed files redundant ly in multiple Availability Zones that are geographically separated. This is the default if you don't specify a storage class.
S3 Intelligent-Tiering	Choose S3 Intelligent-Tiering to optimize storage costs by automatic ally moving data to the most cost-effective storage access tier.
	You pay a monthly charge per object stored in the S3 Intelligent- Tiering storage class. This Amazon S3 charge includes monitoring data access patterns and moving objects between tiers.
S3 Standard-IA	Choose S3 Standard-IA to store your infrequently accessed objects redundantly in multiple Availability Zones that are geographically separated.
	Objects stored in the S3 Standard-IA storage class can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 Standard-IA storage class.
	Objects less than 128 KB are smaller than the minimum capacity charge per object in the S3 Standard-IA storage class. These objects are stored in the S3 Standard storage class.

Amazon S3 storage class	Considerations
S3 One Zone-IA	Choose S3 One Zone-IA to store your infrequently accessed objects in a single Availability Zone.
	Objects stored in the S3 One Zone-IA storage class can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 One Zone-IA storage class.
	Objects less than 128 KB are smaller than the minimum capacity charge per object in the S3 One Zone-IA storage class. These objects are stored in the S3 Standard storage class.
S3 Glacier Instant Retrieval	Choose S3 Glacier Instant Retrieval to archive objects that are rarely accessed but require retrieval in milliseconds.
	Data stored in the S3 Glacier Instant Retrieval storage class offers cost savings compared to the S3 Standard-IA storage class with the same latency and throughput performance. S3 Glacier Instant Retrieval has higher data access costs than S3 Standard-IA, though.
	Objects stored in S3 Glacier Instant Retrieval can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 Glacier Instant Retrieval storage class.
	Objects less than 128 KB are smaller than the minimum capacity charge per object in the S3 Glacier Instant Retrieval storage class. These objects are stored in the S3 Standard storage class.

Amazon S3 storage class	Considerations
S3 Glacier Flexible Retrieval	Choose S3 Glacier Flexible Retrieval for more active archives.  Objects stored in S3 Glacier Flexible Retrieval can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 Glacier Flexible Retrieval storage class.  The S3 Glacier Flexible Retrieval storage class requires 40 KB of additional metadata for each archived object. DataSync puts objects that are less than 40 KB in the S3 Standard storage class. You must restore objects archived in this storage class before DataSync can read them. For information, see Working with archived objects in the Amazon S3 User Guide.  When using S3 Glacier Flexible Retrieval, choose the Verify only the data transferred task option to compare data and metadata checksums at the end of the transfer. You can't use the Verify all data in the destination option for this storage class because it requires retrieving all existing objects from the destination.

Amazon S3 storage class	Considerations
S3 Glacier Deep Archive	Choose S3 Glacier Deep Archive to archive your objects for long-term data retention and digital preservation where data is accessed once or twice a year.
	Objects stored in S3 Glacier Deep Archive can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 Glacier Deep Archive storage class.
	The S3 Glacier Deep Archive storage class requires 40 KB of additional metadata for each archived object. DataSync puts objects that are less than 40 KB in the S3 Standard storage class.
	You must restore objects archived in this storage class before DataSync can read them. For information, see <u>Working with archived objects</u> in the <i>Amazon S3 User Guide</i> .
	When using S3 Glacier Deep Archive, choose the <b>Verify only the data transferred</b> task option to compare data and metadata checksums at the end of the transfer. You can't use the <b>Verify all data in the destination</b> option for this storage class because it requires retrieving all existing objects from the destination.
S3 Outposts	The storage class for Amazon S3 on Outposts.

# **Evaluating S3 request costs when using DataSync**

With Amazon S3 locations, you incur costs related to S3 API requests made by DataSync. This section can help you understand how DataSync uses these requests and how they might affect your Amazon S3 costs.

## **Topics**

- S3 requests made by DataSync
- Cost considerations

## S3 requests made by DataSync

The following table describes the S3 requests that DataSync can make when you're copying data to or from an Amazon S3 location.

S3 request	How DataSync uses it
<u>ListObjectV2</u>	DataSync makes at least one LIST request for every object ending in a forward slash (/) to list the objects that start with that prefix. This request is called during a task's preparing phase.
HeadObject	DataSync makes HEAD requests to retrieve object metadata during a task's <u>preparing</u> and <u>verifying</u> phases. There can be multiple HEAD requests per object depending on how you want DataSync to <u>verify the integrity of the data it transfers</u> .
GetObject	DataSync makes GET requests to read data from an object during a task's <u>transferring</u> phase. There can be multiple GET requests for large objects.
GetObjectTagging	If you configure your task to copy object tags, DataSync makes these GET requests to check for object tags during the task's preparing and transferring phases.
PutObject	DataSync makes PUT requests to create objects and prefixes in a destination S3 bucket during a task's <u>transferring</u> phase. Since DataSync uses the <u>Amazon S3 multipart</u> upload feature, there can be multiple PUT

S3 request	How DataSync uses it
	requests for large objects. To help minimize storage costs, we recommend using a <u>lifecycle</u> configuration to stop incomplete multipart uploads.
<u>PutObjectTagging</u>	If your source objects have tags and you configure your task to copy object tags, DataSync makes these PUT requests when transferring those tags.
CopyObject	DataSync makes a COPY request to create a copy of an object only if that object's metadata changes. This can happen if you originally copied data to the S3 bucket using another service or tool that didn't carry over its metadata.

#### **Cost considerations**

DataSync makes S3 requests on S3 buckets every time you run your task. This can lead to charges adding up in certain situations. For example:

- You're frequently transferring objects to or from an S3 bucket.
- You may not be transferring much data, but your S3 bucket has lots of objects in it. You can still
  see high charges in this scenario because DataSync makes S3 requests on each of the bucket's
  objects.
- You're transferring between S3 buckets, so DataSync is making S3 requests on the source and destination.

To help minimize S3 request costs related to DataSync, consider the following:

## **Topics**

- What S3 storage classes am I using?
- How often do I need to transfer my data?

## What S3 storage classes am I using?

S3 request charges can vary based on the Amazon S3 storage class your objects are using, particularly for classes that archive objects (such as S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval, and S3 Glacier Deep Archive).

Here are some scenarios in which storage classes can affect your S3 request charges when using DataSync:

- Each time you run a task, DataSync makes HEAD requests to retrieve object metadata. These
  requests result in charges even if you aren't moving any objects. How much these requests affect
  your bill depends on the storage class your objects are using along with the number of objects
  that DataSync scans.
- If you moved objects into the S3 Glacier Instant Retrieval storage class (either directly or through a bucket lifecycle configuration), requests on objects in this class are more expensive than objects in other storage classes.
- If you configure your DataSync task to <u>verify that your source and destination locations are fully synchronized</u>, there will be GET requests for each object in all storage classes (except S3 Glacier Flexible Retrieval and S3 Glacier Deep Archive).
- In addition to GET requests, you incur data retrieval costs for objects in the S3 Standard-IA, S3
   One Zone-IA, or S3 Glacier Instant Retrieval storage class.

For more information, see Amazon S3 pricing.

#### How often do I need to transfer my data?

If you need to move data on a recurring basis, think about a <u>schedule</u> that doesn't run more tasks than you need.

You may also consider limiting the scope of your transfers. For example, you can configure DataSync to focus on objects in certain prefixes or <u>filter what data gets transferred</u>. These options can help reduce the number of S3 requests made each time you run your DataSync task.

# **Object considerations with Amazon S3 transfers**

- If you're transferring from an S3 bucket, use <u>S3 Storage Lens</u> to determine how many objects you're moving.
- When transferring between S3 buckets, we recommend using <u>Enhanced task mode</u> because you aren't subject to DataSync task <u>quotas</u>.

• DataSync might not transfer an object with nonstandard characters in its name. For more information, see the object key naming guidelines in the *Amazon S3 User Guide*.

- When using DataSync with an S3 bucket that uses versioning, remember the following:
  - When transferring to an S3 bucket, DataSync creates a new version of an object if that object is modified at the source. This results in additional charges.
  - An object has different version IDs in the source and destination buckets.
- After initially transferring data from an S3 bucket to a file system (for example, NFS or Amazon FSx), subsequent runs of the same DataSync task won't include objects that have been modified but are the same size they were during the first transfer.

## Creating your transfer location for an Amazon S3 general purpose bucket

To create a location for your transfer, you need an existing S3 general purpose bucket. If you don't have one, see the *Amazon S3 User Guide*.

## ▲ Important

Before you create your location, make sure that you read the following sections:

- Storage class considerations with Amazon S3 transfers
- Evaluating S3 request costs when using DataSync

## **Using the DataSync console**

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For **Location type**, choose **Amazon S3**, and then choose **General purpose bucket**.
- 4. For **S3 URI**, enter or choose the bucket and prefix that you want to use for your location.

## **Marning**

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

/photos

- photos//2006/January
- photos/./2006/February
- photos/../2006/March

5. For **S3 storage class when used as a destination**, choose a storage class that you want your objects to use when Amazon S3 is a transfer destination.

For more information, see <u>Storage class considerations with Amazon S3 transfers</u>.

- 6. For **IAM role**, do one of the following:
  - Choose **Autogenerate** for DataSync to automatically create an IAM role with the permissions required to access the S3 bucket.

If DataSync previously created an IAM role for this S3 bucket, that role is chosen by default.

- Choose a custom IAM role that you created. For more information, see <u>Creating an IAM role</u> for DataSync to access your Amazon S3 location.
- 7. (Optional) Choose **Add new tag** to tag your Amazon S3 location.

Tags can help you manage, filter, and search for your resources. We recommend creating a name tag for your location.

8. Choose **Create location**.

## Using the AWS CLI

Copy the following create-location-s3 command:

```
aws datasync create-location-s3 \
    --s3-bucket-arn 'arn:aws:s3:::amzn-s3-demo-bucket' \
    --s3-storage-class 'your-S3-storage-class' \
    --s3-config 'BucketAccessRoleArn=arn:aws:iam::account-id:role/role-allowing-datasync-operations' \
    --subdirectory /your-prefix-name
```

- 2. For --s3-bucket-arn, specify the ARN of the S3 bucket that you want to use as a location.
- 3. For --s3-storage-class, specify a storage class that you want your objects to use when Amazon S3 is a transfer destination.
- 4. For --s3-config, specify the ARN of the IAM role that DataSync needs to access your bucket.

For more information, see Creating an IAM role for DataSync to access your Amazon S3 location.

5. For --subdirectory, specify a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

## Marning

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February
- photos/../2006/March
- Run the create-location-s3 command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:111222333444:location/
loc-0b3017fc4ba4a2d8d"
}
```

You can use this location as a source or destination for your DataSync task.

## Creating your transfer location for an S3 on Outposts bucket

To create a location for your transfer, you need an existing Amazon S3 on Outposts bucket. If you don't have one, see the Amazon S3 on Outposts User Guide.

You also need a DataSync agent. For more information, see Deploying your agent on AWS Outposts.

When transferring from an S3 on Outposts bucket prefix that contains a large dataset (such as hundreds of thousands or millions of objects), your DataSync task might time out. To avoid this,

consider using a <u>DataSync manifest</u>, which lets you specify the exact objects that you need to transfer.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For **Location type**, choose **Amazon S3**, and then choose **Outposts bucket**.
- 4. For **S3 bucket**, choose an Amazon S3 access point that can access your S3 on Outposts bucket.
  - For more information, see the *Amazon S3 User Guide*.
- 5. For **S3** storage class when used as a destination, choose a storage class that you want your objects to use when Amazon S3 is a transfer destination.
  - For more information, see <u>Storage class considerations with Amazon S3 transfers</u>. DataSync by default uses the S3 Outposts storage class for Amazon S3 on Outposts.
- 6. For **Agents**, specify the Amazon Resource Name (ARN) of the DataSync agent on your Outpost.
- 7. For **Folder**, enter a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

## Marning

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February
- photos/../2006/March
- 8. For **IAM role**, do one of the following:
  - Choose **Autogenerate** for DataSync to automatically create an IAM role with the permissions required to access the S3 bucket.

If DataSync previously created an IAM role for this S3 bucket, that role is chosen by default.

• Choose a custom IAM role that you created. For more information, see <u>Creating an IAM role</u> for DataSync to access your Amazon S3 location.

9. (Optional) Choose **Add new tag** to tag your Amazon S3 location.

Tags can help you manage, filter, and search for your resources. We recommend creating a name tag for your location.

10. Choose Create location.

## **Using the AWS CLI**

1. Copy the following create-location-s3 command:

```
aws datasync create-location-s3 \
    --s3-bucket-arn 'bucket-access-point' \
    --s3-storage-class 'your-S3-storage-class' \
    --s3-config 'BucketAccessRoleArn=arn:aws:iam::account-id:role/role-allowing-datasync-operations' \
    --subdirectory /your-folder \
    --agent-arns 'arn:aws:datasync:your-region:account-id::agent/agent-agent-id'
```

2. For --s3-bucket-arn, specify the ARN an Amazon S3 access point that can access your S3 on Outposts bucket.

For more information, see the *Amazon S3 User Guide*.

3. For --s3-storage-class, specify a storage class that you want your objects to use when Amazon S3 is a transfer destination.

For more information, see <u>Storage class considerations with Amazon S3 transfers</u>. DataSync by default uses the S3 Outposts storage class for S3 on Outposts.

4. For --s3-config, specify the ARN of the IAM role that DataSync needs to access your bucket.

For more information, see <u>Creating an IAM role for DataSync to access your Amazon S3</u> location.

5. For --subdirectory, specify a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

#### Marning

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February
- photos/../2006/March
- 6. For --agent-arns, specify the ARN of the DataSync agent on your Outpost.
- 7. Run the create-location-s3 command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:111222333444:location/
loc-0b3017fc4ba4a2d8d"
}
```

You can use this location as a source or destination for your DataSync task.

#### Amazon S3 transfers across AWS accounts

With DataSync, you can move data to or from S3 buckets in different AWS accounts. For more information, see the following tutorials:

- Transferring data from on-premises storage to Amazon S3 across AWS accounts
- Transferring data from Amazon S3 to Amazon S3 across AWS accounts

## Amazon S3 transfers between commercial and AWS GovCloud (US) Regions

By default, DataSync doesn't transfer between S3 buckets in commercial and AWS GovCloud (US) Regions. You can still set up this kind of transfer, though, by creating an object storage location for one of the S3 buckets in your transfer. You can perform this type of transfer with or without

an agent. If you use an agent, your task must be configured for **Basic** mode. To transfer without an agent, you must use **Enhanced** mode.

**Before you begin**: Make sure that you understand the cost implications of transferring between Regions. For more information, see AWS DataSync Pricing.

#### **Contents**

- Providing DataSync access to your object storage location's bucket
- Creating your DataSync agent (optional)
- Creating an object storage location for your S3 bucket

## Providing DataSync access to your object storage location's bucket

When creating the object storage location for this transfer, you must provide DataSync the credentials of an IAM user with permission to access the location's S3 bucket. For more information, see Required permissions.



## Marning

IAM users have long-term credentials, which presents a security risk. To help mitigate this risk, we recommend that you provide these users with only the permissions they require to perform the task and that you remove these users when they are no longer needed.

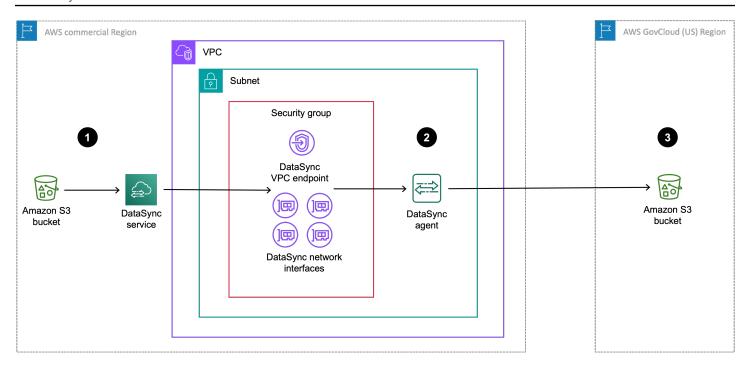
## Creating your DataSync agent (optional)

If you want to run your transfer using Basic mode, then you will need to use an agent. Because you're transferring between a commercial and AWS GovCloud (US) Region, you deploy your DataSync agent as an Amazon EC2 instance in one of the Regions. We recommend that your agent use a VPC service endpoint to avoid data transfer charges out to the public internet. For more information, see Amazon EC2 Data Transfer pricing.

Choose one of the following scenarios that describe how to create an agent based on the Region where you plan to run your DataSync task.

## When running a DataSync task in a commercial Region

The following diagram shows a transfer where your DataSync task and agent are in the commercial Region.



Reference	Description
1	In the commercial Region where you're running a DataSync task, data transfers from the source S3 bucket. The source bucket is configured as an <a href="Mailto:Amazon S3 location">Amazon S3 location</a> in the commercial Region.
2	Data transfers through the DataSync agent, which is in the same VPC and subnet where the VPC service endpoint and <a href="network interfaces">network interfaces</a> are located.
3	Data transfers to the destination S3 bucket in the AWS GovCloud (US) Region. The destination bucket is configured as an <u>object storage location</u> in the commercial Region.

You can use this same setup to transfer the opposite direction, too, from the AWS GovCloud (US) Region to the commercial Region.

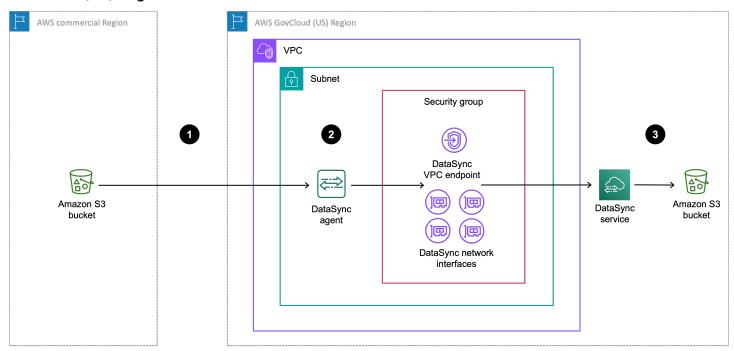
## To create your DataSync agent

- 1. Deploy an Amazon EC2 agent in your commercial Region.
- 2. Configure your agent to use a VPC service endpoint.

## 3. Activate your agent.

## When running a DataSync task in a GovCloud (US) Region

The following diagram shows a transfer where your DataSync task and agent are in the AWS GovCloud (US) Region.



Reference	Description
1	Data transfers from the source S3 bucket in the commercial Region to the AWS GovCloud (US) Region where you're running a DataSync task. The source bucket is configured as an <u>object storage location</u> in the AWS GovCloud (US) Region.
2	In the AWS GovCloud (US) Region, data transfers through the DataSync agent in the same VPC and subnet where the VPC service endpoint and <a href="network interfaces">network interfaces</a> are located.
3	Data transfers to the destination S3 bucket in the AWS GovCloud (US) Region. The destination bucket is configured as an <a href="Mailto:Amazon S3 location">Amazon S3 location</a> in the AWS GovCloud (US) Region.

You can use this same setup to transfer the opposite direction, too, from the AWS GovCloud (US) Region to the commercial Region.

## To create your DataSync agent

- 1. Deploy an Amazon EC2 agent in your AWS GovCloud (US) Region.
- Configure your agent to use a VPC service endpoint.
- 3. Activate your agent.

If your dataset is highly compressible, you might see reduced costs by instead creating your agent in a commercial Region while running a task in an AWS GovCloud (US) Region. There's more setup than normal for creating this agent, including preparing the agent for use in a commercial Region. For information about creating an agent for this setup, see the Move data in and out of AWS GovCloud (US) with AWS DataSync blog.

## Creating an object storage location for your S3 bucket

You need an object storage location for the S3 bucket that's in the Region where you aren't running your DataSync task.

## Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. Make sure that you're in the same Region where you plan to run your task.
- 3. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 4. For **Location type**, choose **Object storage**.
- 5. For **Agents**, choose the DataSync agent that you created for this transfer.
- 6. For **Server**, enter an Amazon S3 endpoint for your bucket by using one of the following formats:
  - Commercial Region bucket: s3. your-region. amazonaws.com
  - AWS GovCloud (US) Region bucket: s3. your-gov-region.amazonaws.com

For a list of Amazon S3 endpoints, see the AWS General Reference.

- 7. For **Bucket** name, enter the name of the S3 bucket.
- 8. For **Folder**, enter a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

## Marning

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February
- photos/../2006/March
- Select **Requires credentials** and do the following:
  - For Access key, enter the access key for an IAM user that can access the bucket.
  - For **Secret key**, enter the same IAM user's secret key.
- 10. (Optional) Choose **Add tag** to tag your location.

Tags can help you manage, filter, and search for your resources. We recommend creating a name tag for your location.

11. Choose Create location.

#### Using the AWS CLI

Copy the following create-location-object-storage command:

```
aws datasync create-location-object-storage \
    --server-hostname s3-endpoint \
    --bucket-name amzn-s3-demo-bucket \
    --agent-arns arn:aws:datasync:your-region:123456789012:agent/
agent-01234567890deadfb
```

- For the --server-hostname parameter, specify an Amazon S3 endpoint for your bucket by using one of the following formats:
  - Commercial Region bucket: s3. your-region. amazonaws.com
  - AWS GovCloud (US) Region bucket: s3. your-gov-region.amazonaws.com

For the Region in the endpoint, make sure that you specify the same Region where you plan to run your task.

For a list of Amazon S3 endpoints, see the AWS General Reference.

- 3. For the --bucket-name parameter, specify the name of the S3 bucket.
- 4. For the --agent-arns parameter, specify the DataSync agent that you created for this transfer.
- 5. For the --access-key parameter, specify the access key for an <u>IAM user</u> that can access the bucket.
- 6. For the --secret-key parameter, enter the same IAM user's secret key.
- 7. (Optional) For the --subdirectory parameter, specify a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

## Marning

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February
- photos/../2006/March
- 8. (Optional) For the --tags parameter, specify key-value pairs that represent tags for the location resource.

Tags can help you manage, filter, and search for your resources. We recommend creating a name tag for your location.

9. Run the create-location-object-storage command.

You get a response that shows you the location ARN that you just created.

{

```
"LocationArn": "arn:aws:datasync:us-east-1:123456789012:location/loc-01234567890abcdef"
}
```

You can use this location as a source or destination for your DataSync task. For the other S3 bucket in this transfer, create an Amazon S3 location.

## **Next steps**

Some possible next steps include:

- 1. If needed, create your other location. For more information, see Where can I transfer my data with AWS DataSync?
- 2. <u>Configure DataSync task settings</u>, such as what files to transfer, how to handle metadata, among other options.
- 3. Set a schedule for your DataSync task.
- 4. Configure monitoring for your DataSync task.
- 5. Start your task.

## Configuring AWS DataSync transfers with Amazon EFS

To transfer data to or from your Amazon EFS file system, you must create an AWS DataSync transfer *location*. DataSync can use this location as a source or destination for transferring data.

## Providing DataSync access to Amazon EFS file systems

<u>Creating a location</u> involves understanding how DataSync can access your storage. For Amazon EFS, DataSync mounts your file system as a root user from your virtual private cloud (VPC) using network interfaces.

#### **Contents**

- Determining the subnet and security groups for your mount target
- Accessing restricted file systems
  - Creating a DataSync IAM role for file system access
  - Example file system policy allowing DataSync access

## Determining the subnet and security groups for your mount target

When creating your location, you specify the subnet and security groups that allow DataSync to connect to one of your Amazon EFS file system's mount targets.

The subnet that you specify must be located:

- In the same VPC as your file system.
- In the same Availability Zone as at least one mount target for your file system.



## Note

You don't need to specify a subnet that includes a file system mount target.

The security groups that you specify must allow inbound traffic on Network File System (NFS) port 2049. For information on creating and updating security groups for your mount targets, see the Amazon EFS User Guide.

## Specifying security groups associated with a mount target

You can specify a security group that's associated with one of your file system's mount targets. We recommend this approach from a network management standpoint.

## Specifying security groups that aren't associated with a mount target

You also can specify a security group that isn't associated with one of your file system's mount targets. However, this security group must be able to communicate with a mount target's security group.

For example, here's how you might create a relationship between security group D (for DataSync) and security group M (for the mount target):

- Security group D, which you specify when creating your location, must have a rule that allows outbound connections on NFS port 2049 to security group M.
- Security group M, which you associate with the mount target, must allow inbound access on NFS port 2049 from security group D.

## To find a mount target's security group

The following instructions can help you identify the security group of an Amazon EFS file system mount target that you want DataSync to use for your transfer.

1. In the AWS CLI, run the following describe-mount-targets command.

```
aws efs describe-mount-targets \
    --region file-system-region \
    --file-system-id file-system-id
```

This command returns information about your file system's mount targets (similar to the following example output).

- 2. Take note of the MountTargetId value that you want to use.
- 3. Run the following describe-mount-target-security-groups command using the MountTargetId to see the security group of your mount target.

```
aws efs describe-mount-target-security-groups \
    --region file-system-region \
    --mount-target-id mount-target-id
```

You specify this security group when creating your location.

## Accessing restricted file systems

DataSync can transfer to or from Amazon EFS file systems that restrict access through access points and IAM policies.



#### Note

If DataSync accesses a destination file system through an access point that enforces user identity, the POSIX user and group IDs for your source data aren't preserved if you configure your DataSync task to copy ownership. Instead, the transferred files and folders are set to the access point's user and group IDs. When this happens, task verification fails because DataSync detects a mismatch between metadata in the source and destination locations.

#### **Contents**

- Creating a DataSync IAM role for file system access
- Example file system policy allowing DataSync access

## Creating a DataSync IAM role for file system access

If you have an Amazon EFS file system that restricts access through an IAM policy, you can create an IAM role that provides DataSync permission to read from or write data to the file system. You then might need to specify that role in your file system policy.

## To create the DataSync IAM role

- 1. Open the IAM console at https://console.aws.amazon.com/iam/.
- In the left navigation pane, under Access management, choose Roles, and then choose Create 2. role.
- 3. On the **Select trusted entity** page, for **Trusted entity type**, choose **Custom trust policy**.
- Paste the following JSON into the policy editor: 4.

**JSON** 

```
"Version": "2012-10-17",

"Statement": [{
        "Effect": "Allow",
        "Principal": {
              "Service": "datasync.amazonaws.com"
        },
        "Action": "sts:AssumeRole"
    }]
}
```

- 5. Choose **Next**. On the **Add permissions** page, choose **Next**.
- 6. Give your role a name and choose **Create role**.

You specify this role when creating your location.

#### Example file system policy allowing DataSync access

The following example file system policy shows how access to an Amazon EFS file system (identified in the policy as fs-1234567890abcdef0) is restricted but still allows access to DataSync through an IAM role named MyDataSyncRole:

**JSON** 

```
}
    "Version": "2012-10-17",
    "Id": "ExampleEFSFileSystemPolicy",
    "Statement": [{
        "Sid": "AccessEFSFileSystem",
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::111122223333:role/MyDataSyncRole"
       },
        "Action": [
            "elasticfilesystem:ClientMount",
            "elasticfilesystem:ClientWrite",
            "elasticfilesystem:ClientRootAccess"
       ],
        "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/fs-1234567890abcdef0",
        "Condition": {
            "Bool": {
```

- Principal Specifies an IAM role that gives DataSync permission to access the file system.
- Action Gives DataSync root access and allows it to read from and write to the file system.
- aws: SecureTransport Requires NFS clients to use TLS when connecting to the file system.
- elasticfilesystem: AccessPointArn Allows access to the file system only through a specific access point.

#### **Network considerations with Amazon EFS transfers**

VPCs that you use with DataSync must have default tenancy. VPCs with dedicated tenancy aren't supported.

#### Performance considerations with Amazon EFS transfers

Your Amazon EFS file system's throughput mode can affect transfer duration and file system performance during the transfer. Consider the following:

- For best results, we recommend using Elastic throughput mode. If you don't use Elastic throughput mode, your transfer might take longer.
- If you use Bursting throughput mode, the performance of your file system's applications might be affected because DataSync consumes file system burst credits.
- How you <u>configure DataSync to verify your transferred data</u> can affect file system performance and data access costs.

For more information, see <u>Amazon EFS performance</u> in the *Amazon Elastic File System User Guide* and the Amazon EFS Pricing page.

## **Creating your Amazon EFS transfer location**

To create the transfer location, you need an existing Amazon EFS file system. If you don't have one, see Getting started with Amazon EFS in the Amazon Elastic File System User Guide.

#### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For **Location type**, choose **Amazon EFS file system**.

You configure this location as a source or destination later.

- For **File system**, choose the Amazon EFS file system that you want to use as a location. 4.
- For **Mount path**, enter a mount path for your Amazon EFS file system. 5.

This specifies where DataSync reads or writes data (depending on if this is a source or destination location) on your file system.

By default, DataSync uses the root directory (or access point if you provide one for the EFS access point setting). You can also specify subdirectories using forward slashes (for example, / path/to/directory).

For **Subnet** choose a subnet where you want DataSync to create the network interfaces for managing your data transfer traffic.

The subnet must be located:

- In the same VPC as your file system.
- In the same Availability Zone as at least one file system mount target.



#### Note

You don't need to specify a subnet that includes a file system mount target.

For **Security groups**, choose the security group associated with your Amazon EFS file system's 7. mount target. You can choose more than one security group.



#### Note

The security groups that you specify must allow inbound traffic on NFS port 2049. For more information, see Determining the subnet and security groups for your mount target.

For In-transit encryption, choose whether you want DataSync to use Transport Layer Security (TLS) encryption when it transfers data to or from your file system.



#### Note

You must enable this setting to configure an access point, IAM role, or both with your Amazon EFS location.

(Optional) For **EFS access point**, choose an access point that DataSync can use to mount your file system.

For more information, see Accessing restricted file systems.

10. (Optional) For IAM role, specify a role that allows DataSync to access your file system.

For information on creating this role, see Creating a DataSync IAM role for file system access.

11. (Optional) Select **Add tag** to tag your file system.

A tag is a key-value pair that helps you manage, filter, and search for your locations.

12. Choose Create location.

#### Using the AWS CLI

Copy the following create-location-efs command:

```
aws datasync create-location-efs \
    --efs-filesystem-arn 'arn:aws:elasticfilesystem:region:account-id:file-
system/file-system-id' \
    --subdirectory /path/to/your/subdirectory \
    --ec2-config SecurityGroupArns='arn:aws:ec2:region:account-id:security-
group/security-group-id', SubnetArn='arn:aws:ec2:region:account-id:subnet/subnet-id'
    --in-transit-encryption TLS1_2 \
```

```
--access-point-arn 'arn:aws:elasticfilesystem: region: account-id: access-
point/access-point-id' \
    --file-system-access-role-arn 'arn:aws:iam::account-id:role/datasync-efs-
access-role
```

For --efs-filesystem-arn, specify the Amazon Resource Name (ARN) of the Amazon EFS file system that you're transferring to or from.

For --subdirectory, specify a mount path for your file system.

This is where DataSync reads or writes data (depending on if this is a source or destination location) on your file system.

By default, DataSync uses the root directory (or access point if you provide one with -access-point-arn). You can also specify subdirectories using forward slashes (for example, /path/to/directory).

- For --ec2-config, do the following:
  - For SecurityGroupArns, specify the ARN of the security group associated with your file system's mount target. You can specify more than one security group.

#### Note

The security groups that you specify must allow inbound traffic on NFS port 2049. For more information, see Determining the subnet and security groups for your mount target.

 For SubnetArn, specify the ARN of the subnet where you want DataSync to create the network interfaces for managing your data transfer traffic.

The subnet must be located:

- In the same VPC as your file system.
- In the same Availability Zone as at least one file system mount target.



#### Note

You don't need to specify a subnet that includes a file system mount target.

For --in-transit-encryption, specify whether you want DataSync to use Transport Layer Security (TLS) encryption when it transfers data to or from your file system.



#### Note

You must set this to TLS1\_2 to configure an access point, IAM role, or both with your Amazon EFS location.

(Optional) For --access-point-arn, specify the ARN of an access point that DataSync can use to mount your file system.

For more information, see Accessing restricted file systems.

(Optional) For --file-system-access-role-arn, specify the ARN of an IAM role that allows DataSync to access your file system.

For information on creating this role, see Creating a DataSync IAM role for file system access.

Run the create-location-efs command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:111222333444:location/
loc-0b3017fc4ba4a2d8d"
}
```

# Configuring transfers with FSx for Windows File Server

To transfer data to or from your Amazon FSx for Windows File Server file system, you must create an AWS DataSync transfer *location*. DataSync can use this location as a source or destination for transferring data.

# Providing DataSync access to FSx for Windows File Server file systems

DataSync connects to your FSx for Windows File Server file system with the Server Message Block (SMB) protocol and mounts it from your virtual private cloud (VPC) using network interfaces.



#### Note

VPCs that you use with DataSync must have default tenancy. VPCs with dedicated tenancy aren't supported.

#### **Topics**

- Required permissions
- Required authentication protocols
- DFS Namespaces

#### Required permissions

You must provide DataSync a user with the necessary rights to mount and access your FSx for Windows File Server files, folders, and file metadata.

We recommend that this user belong to a Microsoft Active Directory group for administering your file system. The specifics of this group depends on your Active Directory setup:

- If you're using AWS Directory Service for Microsoft Active Directory with FSx for Windows File Server, the user must be a member of the AWS Delegated FSx Administrators group.
- If you're using self-managed Active Directory with FSx for Windows File Server, the user must be a member of one of two groups:
  - The **Domain Admins** group, which is the default delegated administrators group.
  - A custom delegated administrators group with user rights that allow DataSync to copy object ownership permissions and Windows access control lists (ACLs).



#### Important

You can't change the delegated administrators group after the file system has been deployed. You must either redeploy the file system or restore it from a backup to use the custom delegated administrator group with the following user rights that DataSync needs to copy metadata.

User right	Description
<b>Restore files and directories (</b> SE_RESTOR E_NAME )	Allows DataSync to copy object ownership , permissions, file metadata, and NTFS discretionary access lists (DACLs).
	This user right is usually granted to members of the <b>Domain Admins</b> and <b>Backup Operators</b> groups (both of which are default Active Directory groups).
Manage auditing and security log (SE_SECURITY_NAME )	Allows DataSync to copy NTFS system access control lists (SACLs).
	This user right is usually granted to members of the <b>Domain Admins</b> group.

• If you want to copy Windows ACLs and are transferring between an SMB file server and FSx for Windows File Server file system or between FSx for Windows File Server file systems, the users that you provide DataSync must belong to the same Active Directory domain or have an Active Directory trust relationship between their domains.



#### Marning

Your FSx for Windows File Server file system's SYSTEM user must have Full control permissions on all folders in your file system. Do not change the NTFS ACL permissions for this user on your folders. If you do, DataSync can change your file system's permissions in a way that makes your file share inaccessible and prevents file system backups from being usable. For more information on file- and folder-level access, see the *Amazon FSx for* Windows File Server User Guide.

### **Required authentication protocols**

Your FSx for Windows File Server must use NTLM authentication for DataSync to access it. DataSync can't access a file server that uses Kerberos authentication.

#### **DFS Namespaces**

DataSync doesn't support Microsoft Distributed File System (DFS) Namespaces. We recommend specifying an underlying file server or share instead when creating your DataSync location.

For more information, see Grouping multiple file systems with DFS Namespaces in the Amazon FSx for Windows File Server User Guide.

# **Creating your FSx for Windows File Server transfer location**

Before you begin, make sure that you have an existing FSx for Windows File Server in your AWS Region. For more information, see Getting started with Amazon FSx in the Amazon FSx for Windows File Server User Guide.

### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- For **Location type**, choose **Amazon FSx**. 3.
- For **FSx file system**, choose the FSx for Windows File Server file system that you want to use as 4. a location.
- For **Share name**, enter a mount path for your FSx for Windows File Server using forward slashes.

This specifies the path where DataSync reads or writes data (depending on if this is a source or destination location).

You can also include subdirectories (for example, /path/to/directory).

For **Security groups**, choose up to five Amazon EC2 security groups that provide access to your file system's preferred subnet.

The security groups that you choose must be able to communicate with your file system's security groups. For information about configuring security groups for file system access, see the Amazon FSx for Windows File Server User Guide.



#### Note

If you choose a security group that doesn't allow connections from within itself, do one of the following:

- Configure the security group to allow it to communicate within itself.
- Choose a different security group that can communicate with the mount target's security group.

7. For **User**, enter the name of a user that can access your FSx for Windows File Server.

For more information, see Required permissions.

- 8. For **Password**, enter password of the user name.
- 9. (Optional) For **Domain**, enter the name of the Windows domain that your FSx for Windows File Server file system belongs to.

If you have multiple Active Directory domains in your environment, configuring this setting makes sure that DataSync connects to the right file system.

10. (Optional) Enter values for the **Key** and **Value** fields to tag the FSx for Windows File Server.

Tags help you manage, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

11. Choose Create location.

#### **Using the AWS CLI**

#### To create an FSx for Windows File Server location by using the AWS CLI

• Use the following command to create an Amazon FSx location.

```
aws datasync create-location-fsx-windows \
    --fsx-filesystem-arn arn:aws:fsx:region:account-id:file-system/filesystem-id \
    --security-group-arns arn:aws:ec2:region:account-id:security-group/group-id \
    --user smb-user --password password
```

In the create-location-fsx-windows command, do the following:

- fsx-filesystem-arn Specify the Amazon Resource Name (ARN) of the file system that you want to transfer to or from.
- security-group-arns Specify the ARNs of up to five Amazon EC2 security groups that provide access to your file system's preferred subnet.

The security groups that you specify must be able to communicate with your file system's security groups. For information about configuring security groups for file system access, see the Amazon FSx for Windows File Server User Guide.



#### Note

If you choose a security group that doesn't allow connections from within itself, do one of the following:

- Configure the security group to allow it to communicate within itself.
- Choose a different security group that can communicate with the mount target's security group.
- The AWS Region The Region that you specify is the one where your target Amazon FSx file system is located.

The preceding command returns a location ARN similar to the one shown following.

```
{
    "LocationArn": "arn:aws:datasync:us-west-2:111222333444:location/
loc-07db7abfc326c50fb"
}
```

# Configuring DataSync transfers with FSx for Lustre

To transfer data to or from your Amazon FSx for Lustre file system, you must create an AWS DataSync transfer location. DataSync can use this location as a source or destination for transferring data.

# Providing DataSync access to FSx for Lustre file systems

DataSync accesses your FSx for Lustre file system using the Lustre client. DataSync requires access to all data on your FSx for Lustre file system. To have this level of access, DataSync mounts your file system as the root user using a user ID (UID) and group ID (GID) of 0.

DataSync mounts your file system from your virtual private cloud (VPC) using network interfaces. DataSync fully manages the creation, the use, and the deletion of these network interfaces on your behalf.



#### Note

VPCs that you use with DataSync must have default tenancy. VPCs with dedicated tenancy aren't supported.

# **Creating your FSx for Lustre transfer location**

To create the transfer location, you need an existing FSx for Lustre file system. For more information, see Getting started with Amazon FSx for Lustre in the Amazon FSx for Lustre User Guide.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**. 2.
- For Location type, choose Amazon FSx. 3.

You configure this location as a source or destination later.

- For **FSx file system**, choose the FSx for Lustre file system that you want to use as a location.
- 5. For **Mount path**, enter the mount path for your FSx for Lustre file system.
  - The path can include a subdirectory. When the location is used as a source, DataSync reads data from the mount path. When the location is used as a destination, DataSync writes all data to the mount path. If a subdirectory isn't provided, DataSync uses the root directory (/).
- For **Security groups**, choose up to five security groups that provide access to your FSx for Lustre file system.
  - The security groups must be able to access the file system's ports. The file system must also allow access from the security groups.
  - For more information about security groups, see File System Access Control with Amazon VPC in the Amazon FSx for Lustre User Guide.
- (Optional) Enter values for the **Key** and **Value** fields to tag the FSx for Lustre file system.
  - Tags help you manage, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.
- Choose Create location. 8.

#### Using the AWS CLI

#### To create an FSx for Lustre location by using the AWS CLI

• Use the following command to create an FSx for Lustre location.

```
aws datasync create-location-fsx-lustre \
    --fsx-filesystem-arn arn:aws:fsx:region:account-id:file-system:filesystem-id \
    --security-group-arns arn:aws:ec2:region:account-id:security-group/group-id
```

The following parameters are required in the create-location-fsx-lustre command.

- fsx-filesystem-arn The fully qualified Amazon Resource Name (ARN) of the file system that you want to read from or write to.
- security-group-arns The ARN of an Amazon EC2 security group to apply to the network interfaces of the file system's preferred subnet.

The preceding command returns a location ARN similar to the following.

```
{
    "LocationArn": "arn:aws:datasync:us-west-2:111222333444:location/
loc-07sb7abfc326c50fb"
}
```

# Configuring DataSync transfers with Amazon FSx for OpenZFS

To transfer data to or from your Amazon FSx for OpenZFS file system, you must create an AWS DataSync transfer *location*. DataSync can use this location as a source or destination for transferring data.

# Providing DataSync access to FSx for OpenZFS file systems

DataSync mounts your FSx for OpenZFS file system from your virtual private cloud (VPC) using <a href="network interfaces">network interfaces</a>. DataSync fully manages the creation, the use, and the deletion of these network interfaces on your behalf.



#### Note

VPCs that you use with DataSync must have default tenancy. VPCs with dedicated tenancy aren't supported.

# Configuring FSx for OpenZFS file system authorization

DataSync accesses your FSx for OpenZFS file system as an NFS client, mounting the file system as a root user with a user ID (UID) and group ID (GID) of 0.

For DataSync to copy all of your file metadata, you must configure the NFS export settings on your file system volumes using no\_root\_squash. However, you can limit this level of access to only a specific DataSync task.

For more information, see Volume properties in the Amazon FSx for OpenZFS User Guide.

#### Configuring NFS exports specific to DataSync (recommended)

You can configure an NFS export specific to each volume that's accessed only by your DataSync task. Do this for the most recent ancestor volume of the mount path that you specify when creating your FSx for OpenZFS location.

#### To configure an NFS export specific to DataSync

1. Create your DataSync task.

This creates the task's network interfaces that you specify in your NFS export settings.

- Locate the private IP addresses of the task's network interfaces by using the Amazon EC2 console or AWS CLI.
- For your FSx for OpenZFS file system volume, configure the following NFS export settings for each of the task's network interfaces:
  - Client address: Enter the network interface's private IP address (for example, 10.24.34.0).
  - NFS options: Enter rw, no\_root\_squash.

#### Configuring NFS exports for all clients

You can specify an NFS export that allows root access to all clients.

#### To configure an NFS export for all clients

- For your FSx for OpenZFS file system volume, configure the following NFS export settings:
  - Client address: Enter \*.
  - NFS options: Enter rw, no\_root\_squash.

## **Creating your FSx for OpenZFS transfer location**

To create the location, you need an existing FSx for OpenZFS file system. If you don't have one, see Getting started with Amazon FSx for OpenZFS in the Amazon FSx for OpenZFS User Guide.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Locations**, and then choose **Create location**.
- 3. For **Location type**, choose **Amazon FSx**.

You configure this location as a source or destination later.

- 4. For **FSx file system**, choose the FSx for OpenZFS file system that you want to use as a location.
- 5. For **Mount path**, enter the mount path for your FSx for OpenZFS file system.
  - The path must begin with /fsx and can be any existing directory path in the file system. When the location is used as a source, DataSync reads data from the mount path. When the location is used as a destination, DataSync writes all data to the mount path. If a subdirectory isn't provided, DataSync uses the root volume directory (for example, /fsx).
- 6. For **Security groups**, choose up to five security groups that provide network access to your FSx for OpenZFS file system.
  - The security groups must provide access to the network ports that are used by the FSx for OpenZFS file system. The file system must allow network access from the security groups.
  - For more information about security groups, see <u>File system access control with Amazon VPC</u> in the *Amazon FSx for OpenZFS User Guide*.
- 7. (Optional) Expand **Additional settings** and for **NFS version** choose the NFS version that DataSync uses to access your file system.
  - By default, DataSync uses NFS version 4.1.

8. (Optional) Enter values for the **Key** and **Value** fields to tag the FSx for OpenZFS file system.

Tags help you manage, filter, and search for your location. We recommend creating at least a name tag for your location.

Choose Create location.

#### **Using the AWS CLI**

#### To create an FSx for OpenZFS location by using the AWS CLI

1. Copy the following create-location-fsx-open-zfs command:

```
aws datasync create-location-fsx-open-zfs \
    --fsx-filesystem-arn arn:aws:fsx:region:account-id:file-system/filesystem-id \
    --security-group-arns arn:aws:ec2:region:account-id:security-group/group-id \
    --protocol NFS={}
```

- 2. Specify the following required options in the command:
  - For fsx-filesystem-arn, specify the location file system's fully qualified Amazon Resource Name (ARN). This includes the AWS Region where your file system resides, your AWS account, and the file system ID.
  - For security-group-arns, specify the ARN of the Amazon EC2 security group that
    provides access to the <u>network interfaces</u> of your FSx for OpenZFS file system's preferred
    subnet. This includes the AWS Region where your Amazon EC2 instance resides, your AWS
    account, and the security group ID.

For more information about security groups, see <u>File System Access Control with Amazon</u> VPC in the *Amazon FSx for OpenZFS User Guide*.

- For protocol, specify the protocol that DataSync uses to access your file system. (DataSync currently supports only NFS.)
- 3. Run the command. You get a response showing the location that you just created.

```
{
    "LocationArn": "arn:aws:datasync:us-west-2:123456789012:location/loc-abcdef01234567890"
}
```

# Configuring transfers with Amazon FSx for NetApp ONTAP

To transfer data to or from your Amazon FSx for NetApp ONTAP file system, you must create an AWS DataSync transfer location. DataSync can use this location as a source or destination for transferring data.

# Providing DataSync access to FSx for ONTAP file systems

To access an FSx for ONTAP file system, DataSync mounts a storage virtual machine (SVM) on your file system using network interfaces in your virtual private cloud (VPC). DataSync creates these network interfaces in your file system's preferred subnet only when you create a task that includes your FSx for ONTAP location.



#### Note

VPCs that you use with DataSync must have default tenancy. VPCs with dedicated tenancy aren't supported.

DataSync can connect to an FSx for ONTAP file system's SVM and copy data by using the Network File System (NFS) or Server Message Block (SMB) protocol.

#### **Topics**

- Using the NFS protocol
- Using the SMB protocol
- Unsupported protocols
- Choosing the right protocol
- Accessing SnapLock volumes

# Using the NFS protocol

With the NFS protocol, DataSync uses the AUTH\_SYS security mechanism with a user ID (UID) and group ID (GID) of 0 to authenticate with your SVM.



#### Note

DataSync currently only supports NFS version 3 with FSx for ONTAP locations.

#### Using the SMB protocol

With the SMB protocol, DataSync uses credentials that you provide to authenticate with your SVM.

#### **Supported SMB versions**

By default, DataSync automatically chooses a version of the SMB protocol based on negotiation with your SMB file server. You also can configure DataSync to use a specific version, but we recommend doing this only if DataSync has trouble negotiating with the SMB file server automatically. For security reasons, we recommend using SMB version 3.0.2 or later.

See the following table for a list of options in the DataSync console and API for configuring an SMB version with your FSx for ONTAP location:

Console option	API option	Description
Automatic	AUTOMATIC	DataSync and the SMB file server negotiate the highest version of SMB that they mutually support between 2.1 and 3.1.1.
		This is the default and recommended option. If you instead choose a specific version that your file server doesn't support, you may get an Operation Not Supported error.
SMB 3.0.2	SMB3	Restricts the protocol negotiation to only SMB version 3.0.2.
SMB 2.1	SMB2	Restricts the protocol negotiation to only SMB version 2.1.
SMB 2.0	SMB2_0	Restricts the protocol negotiation to only SMB version 2.0.

#### **Required permissions**

You must provide DataSync a local user in your SVM or a domain user in your Microsoft Active Directory with the necessary rights to mount and access your files, folders, and file metadata.

If you provide a user in your Active Directory, note the following:

• If you're using AWS Directory Service for Microsoft Active Directory, the user must be a member of the AWS Delegated FSx Administrators group.

- If you're using a self-managed Active Directory, the user must be a member of one of two groups:
  - The **Domain Admins** group, which is the default delegated administrators group.
  - A custom delegated administrators group with user rights that allow DataSync to copy object ownership permissions and Windows access control lists (ACLs).

#### Important

You can't change the delegated administrators group after the file system has been deployed. You must either redeploy the file system or restore it from a backup to use the custom delegated administrator group with the following user rights that DataSync needs to copy metadata.

User right	Description
Act as part of the operating system (SE_TCB_NAME )	Allows DataSync to copy object ownership , permissions, file metadata, and NTFS discretionary access lists (DACLs).
	This user right is usually granted to members of the <b>Domain Admins</b> and <b>Backup Operators</b> groups (both of which are default Active Directory groups).
Manage auditing and security log (SE_SECURITY_NAME )	Allows DataSync to copy NTFS system access control lists (SACLs).
	This user right is usually granted to members of the <b>Domain Admins</b> group.

 If you want to copy Windows ACLs and are transferring between FSx for ONTAP file systems using SMB (or other types of file systems using SMB), the users that you provide DataSync must belong to the same Active Directory domain or have an Active Directory trust relationship between their domains.

#### **Required authentication protocols**

For DataSync to access your SMB share, your FSx for ONTAP file system must use NTLM authentication. DataSync can't access FSx for ONTAP file systems that use Kerberos authentication.

#### **DFS Namespaces**

DataSync doesn't support Microsoft Distributed File System (DFS) Namespaces. We recommend specifying an underlying file server or share instead when creating your DataSync location.

### **Unsupported protocols**

DataSync can't access FSx for ONTAP file systems using the iSCSI (Internet Small Computer Systems Interface) protocol.

#### Choosing the right protocol

To preserve file metadata in FSx for ONTAP migrations, configure your DataSync source and destination locations to use the same protocol. Between the supported protocols, SMB preserves metadata with the highest fidelity (see <u>Understanding how DataSync handles file and object metadata</u> for details).

When migrating from a Unix (Linux) server or network-attached storage (NAS) share that serves users through NFS, do the following:

- 1. <u>Create an NFS location</u> for the Unix (Linux) server or NAS share. (This will be your source location.)
- 2. Configure the FSx for ONTAP volume you're transferring data to with the Unix security style.
- 3. Create a location for your FSx for ONTAP file system that's configured for NFS. (This will be your destination location.)

When migrating from a Windows server or NAS share that serves users through SMB, do the following:

- 1. Create an SMB location for the Windows server or NAS share. (This will be your source location.)
- 2. Configure the FSx for ONTAP volume you're transferring data to with the NTFS security style.
- 3. Create a location for your FSx for ONTAP file system that's configured for SMB. (This will be your destination location.)

If your FSx for ONTAP environment uses multiple protocols, we recommend working with an AWS storage specialist. To learn about best practices for multiprotocol access, see Enabling multiprotocol workloads with Amazon FSx for NetApp ONTAP.

#### **Accessing SnapLock volumes**

If you're transferring data to a SnapLock volume on an FSx for ONTAP file system, make sure the SnapLock settings Autocommit and Volume append mode are disabled on the volume during your transfer. You can re-enable these settings when you're done transferring data.

# Creating your FSx for ONTAP transfer location

To create the location, you need an existing FSx for ONTAP file system. If you don't have one, see Getting started with Amazon FSx for NetApp ONTAP in the Amazon FSx for NetApp ONTAP User Guide.

### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**. 2.
- 3. For Location type, choose Amazon FSx.
  - You configure this location as a source or destination later.
- 4. For **FSx file system**, choose the FSx for ONTAP file system that you want to use as a location.
- For **Storage virtual machine**, choose a storage virtual machine (SVM) in your file system where 5. you want to copy data to or from.
- For **Mount path**, specify a path to the file share in that SVM where you'll copy your data.

You can specify a junction path (also known as a mount point), gtree path (for NFS file shares), or share name (for SMB file shares). For example, your mount path might be /vol1, /vol1/ tree1, or /share1.



#### (i) Tip

Don't specify a path in the SVM's root volume. For more information, see Managing FSx for ONTAP storage virtual machines in the Amazon FSx for NetApp ONTAP User Guide.

For Security groups, choose up to five Amazon EC2 security groups that provide access to your 7. file system's preferred subnet.

The security groups must allow outbound traffic on the following ports (depending on the protocol you use):

- NFS TCP ports 111, 635, and 2049
- **SMB** TCP port 445

Your file system's security groups must also allow inbound traffic on the same ports.

8. For **Protocol**, choose the data transfer protocol that DataSync uses to access your file system's SVM.

For more information, see Choosing the right protocol.

**NFS** 

DataSync uses NFS version 3.

**SMB** 

Configure an SMB version, user, password, and Active Directory domain name (if needed) to access the SVM.

• (Optional) Expand **Additional settings** and choose an **SMB version** for DataSync to use when accessing your SVM.

By default, DataSync automatically chooses a version based on negotiation with the SMB file server. For more information, see Using the SMB protocol.

• For **User**, enter a user name that can mount and access the files, folders, and metadata that you want to transfer in the SVM.

For more information, see <u>Using the SMB protocol</u>.

- For **Password**, enter the password of the user that you specified that can access the SVM.
- (Optional) For Active Directory domain name, enter the fully qualified domain name (FQDN) of the Active Directory that your SVM belongs to.

If you have multiple domains in your environment, configuring this setting makes sure that DataSync connects to the right SVM.

9. (Optional) Enter values for the **Key** and **Value** fields to tag the FSx for ONTAP file system.

Tags help you manage, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

10. Choose Create location.

#### Using the AWS CLI

#### To create an FSx for ONTAP location by using the AWS CLI

1. Copy the following create-location-fsx-ontap command:

```
aws datasync create-location-fsx-ontap \
    --storage-virtual-machine-arn arn:aws:fsx:region:account-id:storage-virtual-
machine/fs-file-system-id \
    --security-group-arns arn:aws:ec2:region:account-id:security-group/group-id \
    --protocol data-transfer-protocol={}
```

- 2. Specify the following required options in the command:
  - For storage-virtual-machine-arn, specify the fully qualified Amazon Resource Name (ARN) of a storage virtual machine (SVM) in your file system where you want to copy data to or from.

This ARN includes the AWS Region where your file system resides, your AWS account, and the file system and SVM IDs.

• For security-group-arns, specify the ARNs of the Amazon EC2 security groups that provide access to the network interfaces of your file system's preferred subnet.

This includes the AWS Region where your Amazon EC2 instance resides, your AWS account, and your security group IDs. You can specify up to five security group ARNs.

For more information about security groups, see <u>File System Access Control with Amazon VPC</u> in the *Amazon FSx for NetApp ONTAP User Guide*.

- For protocol, configure the protocol that DataSync uses to access your file system's SVM.
  - For NFS, you can use the default configuration:

```
--protocol NFS={}
```

For SMB, you must specify a user name and password that can access the SVM:

```
--protocol SMB={User=smb-user, Password=smb-password}
```

3. Run the command.

You get a response that shows the location that you just created.

```
{
    "LocationArn": "arn:aws:datasync:us-west-2:123456789012:location/loc-
abcdef01234567890"
}
```

# Transferring to or from other cloud storage with AWS DataSync

With AWS DataSync, you can transfer data between some other cloud providers and AWS storage services. For more information, see Where can I transfer my data with DataSync?

# **Topics**

- Planning transfers to or from third-party cloud storage systems
- Configuring AWS DataSync transfers with Google Cloud Storage
- Configuring transfers with Microsoft Azure Blob Storage
- Configuring AWS DataSync transfers with Microsoft Azure Files SMB shares
- Configuring transfers with other cloud object storage

# Planning transfers to or from third-party cloud storage systems

When planning cross-cloud data transfers, consider the following:

• **Using an agent:** An agent is only required to access storage in other clouds when using Basic mode tasks. Enhanced mode tasks do not require an agent. If you decide to use an agent, you can deploy it as an Amazon EC2 instance when transferring from a cloud providers' S3-compatible object storage, or as a Google Compute Engine or Azure Virtual Machine for transfers from those specific storage services, respectively. When transferring from filesystems in Google and Azure, we recommend deploying the agent as a Google or Azure VM so that the agent is as close to the filesystem as possible. Additionally, DataSync compresses the data from the agent to AWS, which can help reduce egress costs. DataSync provides a list of validated cloud locations that provide the required Amazon S3 API compatibility.

• The other cloud's object storage endpoint: The storage endpoint for a third-party cloud provider is typically region or account specific. The regional endpoint is used as the server in the DataSync object storage location, together with a specified bucket name.

- Storage classes of the source objects: Like Amazon S3, some cloud providers support an archive tier that requires a restore before being able to access the archived objects. For example, objects in the Azure Blob archive tier must be retrieved for standard access prior to a data transfer. Objects in the Google Cloud Storage archive tier can be accessed immediately and do not require restore, but there are retrieval costs associated with direct archive tier access. Review your cross-cloud storage class documentation to determine access requirements and retrieval fees prior to beginning your data transfer. For more information about restoring archived objects in Amazon S3, see Restoring an archived object in the Amazon Simple Storage Service User Guide.
- Object storage access: Transferring data between third-party cloud providers requires access
  to the other cloud's object storage in the form of authentication keys. For example, to provide
  access to Google Cloud Storage, you configure a DataSync object storage location that
  connects to the Google Cloud Storage XML API and authenticates using a Hash-based Message
  Authentication Code (HMAC) key for your service account. For Azure Blob storage, you configure
  a dedicated Azure Blob DataSync location that authenticates using SAS tokens. DataSync uses
  AWS Secrets Manager to securely store your object storage credentials. For more information,
  see Securing storage location credentials.

### • Object tag support:

- Unlike Amazon S3, not all cloud providers support <u>object tags</u>. DataSync tasks can fail while
  attempting to read tags from the source location if the cloud provider does not support object
  tags through the Amazon S3 API, or if the credentials you provide are insufficient to retrieve
  the tags. DataSync provides a task option to turn off <u>reading and copying object tags</u> during
  a transfer if object tags are not supported, or you don't want to retain the tags. Review your
  cloud provider documentation to determine if object tags are supported, and verify your
  transfer task's object tag settings before initiating the transfer.
- You can use the Amazon S3 API to check whether a cloud provider will return a get-object-tagging request. For more information, see <u>get-object-tagging</u> in the AWS CLI Command Reference.

A cloud provider that supports object tags will return a response similar to the following example:

```
aws s3api get-object-tagging --bucket BUCKET_NAME --endpoint- url=https://
BUCKET_ENDPOINT --key prefix/file1
```

```
{
    "TagSet": []
}
```

A cloud provider that doesn't support get-object-tagging will return the following message:

```
aws s3api get-object-tagging --bucket BUCKET_NAME --endpoint- url=https://
BUCKET_ENDPOINT --key prefix/file1
An error occurred (OperationNotSupported) when calling the GetObjectTagging operation: The operation is not supported for this resource
```

- Associated costs for requests and data egress: Transferring data from cloud object storage has
  request and egress costs associated with reading data and data transfer out. Request charges
  vary between cloud providers and between storage classes where applicable. Consult your cloud
  provider documentation regarding specific costs for requests relative to the storage class you
  plan to read from. For an overview of request charges that DataSync makes for data transfers,
  see Evaluating S3 request costs when using DataSync and AWS DataSync pricing. Transferring
  data out of specific cloud providers results in egress charges. Data transfer costs vary between
  cloud providers and are also dependent on the region where the data is stored.
- Object storage request rates: Cloud providers have various performance and request rate
  characteristics for their object storage platforms. Review your other cloud provider's request
  rates and determine where the request limits are applied. Plan ahead for highly parallelized
  transfers consisting of multiple agents, where specific partitioning or performance increases
  might be required.

Amazon S3 has documented request rates that you can build your solution around. Amazon S3 request rates are per partitioned prefix and are scalable across multiple prefixes. For more information, see <a href="Messages Best practices design patterns">Best practices design patterns</a>: optimizing Amazon S3 performance in the Amazon Simple Storage Service User Guide.

# Configuring AWS DataSync transfers with Google Cloud Storage

The following tutorial shows how you can use AWS DataSync to migrate objects from a Google Cloud Storage (GCS) bucket to an Amazon S3 bucket.

#### Overview

DataSync uses the <u>Google Cloud Storage XML API</u> for data transfers. This API provides an Amazon S3-compatible interface for reading and writing data with GCS buckets.

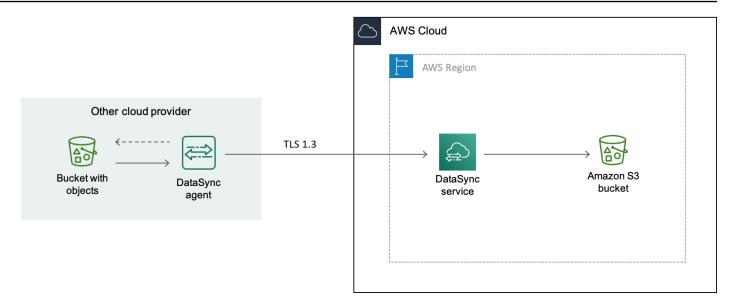
DataSync supports transfers between GCS and AWS storage services. When copying between GCS and Amazon EFS or Amazon FSx, an agent is required and your task must use **Basic** mode. When copying between GCS and Amazon S3, an agent is only required when using **Basic** mode.

If you use an agent for your transfer, then you can deploy the agent in Google Cloud or in your Amazon VPC.

#### Agent in Google Cloud

- 1. You deploy a DataSync agent in your Google Cloud environment.
- 2. The agent reads your Google Cloud Storage bucket by using a Hash-based Message Authentication Code (HMAC) key.
- 3. The objects from your Google Cloud Storage bucket transfer securely through TLS 1.3 into the AWS Cloud by using a public endpoint.
- 4. The DataSync service writes the data to your S3 bucket.

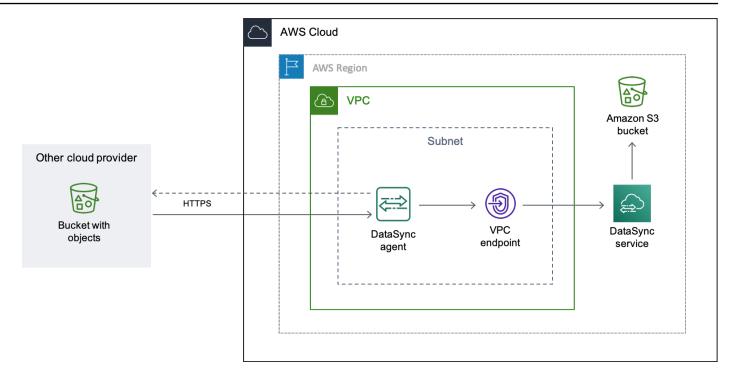
The following diagram illustrates the transfer.



#### Agent in your VPC

- 1. You deploy a DataSync agent in a virtual private cloud (VPC) in your AWS environment.
- 2. The agent reads your Google Cloud Storage bucket by using a Hash-based Message Authentication Code (HMAC) key.
- 3. The objects from your Google Cloud Storage bucket transfer securely through TLS 1.3 into the AWS Cloud by using a private VPC endpoint.
- 4. The DataSync service writes the data to your S3 bucket.

The following diagram illustrates the transfer.



#### Costs

The fees associated with this migration might include:

- Running a Google <u>Compute Engine</u> virtual machine (VM) instance (if you deploy your DataSync agent in Google Cloud)
- Running an Amazon EC2 instance (if you deploy your DataSync agent in a VPC within AWS)
- Transferring the data by using <u>DataSync</u>, including request charges related to <u>Google Cloud</u> Storage and Amazon S3 (if S3 is one of your transfer locations)
- Transferring data out of Google Cloud Storage
- Storing data in Amazon S3

# **Prerequisites**

Before you begin, do the following if you haven't already:

- Create a Google Cloud Storage bucket with the objects that you want to transfer to AWS.
- · Sign up for an AWS account.
- Create an Amazon S3 bucket for storing your objects after they're in AWS.

## Creating an HMAC key for your Google Cloud Storage bucket

DataSync uses an HMAC key that's associated with your Google service account to authenticate with and read the bucket that you're transferring data from. (For detailed instructions on how to create HMAC keys, see the Google Cloud Storage documentation.)

#### To create an HMAC key

- 1. Create an HMAC key for your Google service account.
- 2. Make sure that your Google service account has at least Storage Object Viewer permissions.
- 3. Save your HMAC key's access ID and secret in a secure location.

You'll need these items later to configure your DataSync source location.

# **Step 2: Configure your network**

Network configuration is required only when using a DataSync agent with your transfer. The network requirements for this migration depend on where you choose to deploy your agent.

### For a DataSync agent in Google Cloud

If you want to host your DataSync agent in Google Cloud, configure your network to <u>allow</u> DataSync transfers through a public endpoint.

## For a DataSync agent in your VPC

If you want to host your agent in AWS, you need a VPC with an interface endpoint. DataSync uses the VPC endpoint to facilitate the transfer.

# To configure your network for a VPC endpoint

- 1. If you don't have one, create a VPC in the same AWS Region as your S3 bucket.
- 2. Create a private subnet for your VPC.
- 3. Create a VPC service endpoint for DataSync.
- 4. Configure your network to allow DataSync transfers through a VPC service endpoint.

To do this, modify the security group that's associated with your VPC service endpoint.

## Step 3: Create a DataSync agent (optional)

A DataSync agent is only required when using **Basic** mode tasks. If you are using **Enhanced** mode to transfer between Google Cloud Storage (GCS) and Amazon S3, then no agent is required. If you want to use **Basic** mode, then you need a DataSync agent that can access your GCS bucket.

#### For Google Cloud

In this scenario, the DataSync agent runs in your Google Cloud environment.

Before you begin: Install the Google Cloud CLI.

#### To create the agent for Google Cloud

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Agents**, then choose **Create agent**.
- For Hypervisor, choose VMware ESXi, then choose Download the image to download a .zip file that contains the agent.
- 4. Open a terminal. Unzip the image by running the following command:

```
unzip AWS-DataSync-Agent-VMWare.zip
```

5. Extract the contents of the agent's .ova file beginning with aws-datasync by running the following command:

```
tar -xvf aws-datasync-2.0.1655755445.1-x86_64.xfs.gpt.ova
```

6. Import the agent's .vmdk file into Google Cloud by running the following Google Cloud CLI command:

```
gcloud compute images import aws-datasync-2-test \
    --source-file INCOMPLETE-aws-datasync-2.0.1655755445.1-x86_64.xfs.gpt-disk1.vmdk
    --os centos-7
```

# Note

Importing the .vmdk file might take up to two hours.

7. Create and start a VM instance for the agent image that you just imported.

The instance needs the following configurations for your agent. (For detailed instructions on how to create an instance, see the Google Cloud Compute Engine documentation.)

- For the machine type, choose one of the following:
  - e2-standard-8 For DataSync task executions working with up to 20 million objects.
  - e2-standard-16 For DataSync task executions working with more than 20 million objects.
- For the boot disk settings, go to the custom images section. Then choose the DataSync agent image that you just imported.
- For the service account setting, choose your Google service account (the same account that you used in Step 1).
- For the firewall setting, choose the option to allow HTTP (port 80) traffic.

To activate your DataSync agent, port 80 must be open on the agent. The port doesn't need to be publicly accessible. Once activated, DataSync closes the port.

8. After the VM instance is running, take note of its public IP address.

You'll need this IP address to activate the agent.

- 9. Go back to the DataSync console. On the **Create agent** screen where you downloaded the agent image, do the following to activate your agent:
  - For Endpoint type, choose the public service endpoints option (for example, Public service endpoints in US East Ohio).
  - For Activation key, choose Automatically get the activation key from your agent.
  - For **Agent address**, enter the public IP address of the agent VM instance that you just created.
  - · Choose Get key.
- 10. Give your agent a name, and then choose **Create agent**.

Your agent is online and ready to transfer data.

#### For your VPC

In this scenario, the agent runs as an Amazon EC2 instance in a VPC that's associated with your AWS account.

Before you begin: Set up the AWS Command Line Interface (AWS CLI).

#### To create the agent for your VPC

1. Open a terminal. Make sure to configure your AWS CLI profile to use the account that's associated with your S3 bucket.

2. Copy the following command. Replace *vpc-region* with the AWS Region where your VPC resides (for example, us-east-1).

```
aws ssm get-parameter --name /aws/service/datasync/ami --region vpc-region
```

3. Run the command. In the output, take note of the "Value" property.

This value is the DataSync Amazon Machine Image (AMI) ID of the Region that you specified. For example, an AMI ID could look like ami-1234567890abcdef0.

4. Copy the following URL. Again, replace *vpc-region* with the AWS Region where your VPC resides. Then, replace *ami-id* with the AMI ID that you noted in the previous step.

```
https://console.aws.amazon.com/ec2/v2/home?region=vpc-
region#LaunchInstanceWizard:ami=ami-id
```

5. Paste the URL into a browser.

The Amazon EC2 instance launch page in the AWS Management Console displays.

- 6. For **Instance type**, choose one of the <u>recommended Amazon EC2 instances for DataSync</u> agents.
- 7. For **Key pair**, choose an existing key pair, or create a new one.
- 8. For **Network settings**, choose the VPC and subnet where you want to deploy the agent.
- 9. Choose Launch instance.
- 10. Once the Amazon EC2 instance is running, choose your VPC endpoint.
- 11. Activate your agent.

# Step 4: Create a DataSync source location for your Google Cloud Storage bucket

To set up a DataSync location for your Google Cloud Storage bucket, you need the access ID and secret for the HMAC key that you created in Step 1.

#### To create the DataSync source location

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For **Location type**, choose **Object storage**.
- 4. For Server, enter storage.googleapis.com.
- 5. For **Bucket name**, enter the name of your Google Cloud Storage bucket.
- 6. For **Folder**, enter an object prefix.
  - DataSync only copies objects with this prefix.
- 7. If your transfer requires an agent, choose **Use agents**, then choose the agent that you created in Step 3.
- 8. Expand Additional settings. For Server protocol, choose HTTPS. For Server port, choose 443.
- 9. Scroll down to the **Authentication** section. Make sure that the **Requires credentials** check box is selected, and then do the following:
  - For Access key, enter your HMAC key's access ID.
  - For Secret key, either enter your HMAC key's secret key directly, or specify an AWS Secrets
    Manager secret that contains the key. For more information, see <u>Providing credentials for</u>
    storage locations.
- 10. Choose Create location.

# Step 5: Create a DataSync destination location for your S3 bucket

You need a DataSync location for where you want your data to end up.

## To create the DataSync destination location

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. Create a DataSync location for the S3 bucket.

If you deployed the DataSync agent in your VPC, this tutorial assumes that the S3 bucket is in the same AWS Region as your VPC and DataSync agent.

# Step 6: Create and start a DataSync task

With your source and destinations locations configured, you can start moving your data into AWS.

#### To create and start the DataSync task

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** 2. task.
- On the **Configure source location** page, do the following:
  - Choose **Choose an existing location**. a.
  - b. Choose the source location that you created in Step 4, then choose Next.
- On the **Configure destination location** page, do the following:
  - Choose Choose an existing location. a.
  - Choose the destination location that you created in Step 5, then choose **Next**.
- On the **Configure settings** page, do the following: 5.
  - Under Data transfer configuration, expand Additional settings and clear the Copy object a. tags check box.

#### Important

Because the Google Cloud Storage XML API does not support reading or writing object tags, your DataSync task might fail if you try to copy object tags.

- Configure any other task settings that you want, and then choose **Next**.
- On the **Review** page, review your settings, and then choose **Create task**.
- 7. On the task's details page, choose **Start**, and then choose one of the following:
  - To run the task without modification, choose **Start with defaults**.
  - To modify the task before running it, choose **Start with overriding options**.

When your task finishes, you'll see the objects from your Google Cloud Storage bucket in your S3 bucket.

# **Configuring transfers with Microsoft Azure Blob Storage**

With AWS DataSync, you can transfer data between Microsoft Azure Blob Storage (including Azure Data Lake Storage Gen2 blob storage) and the following AWS storage services:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

To set up this kind of transfer, you create a <u>location</u> for your Azure Blob Storage. You can use this location as a transfer source or destination. A DataSync agent is required only when transferring data between Azure Blob and Amazon EFS or Amazon FSx, or when using **Basic** mode tasks. You don't need an agent to transfer data between Azure Blob and Amazon S3 using **Enhanced** mode.

## **Providing DataSync access to your Azure Blob Storage**

How DataSync accesses your Azure Blob Storage depends on several factors, including whether you're transferring to or from blob storage and what kind of <u>shared access signature (SAS) token</u> you're using. Your objects also must be in an access tier that DataSync can work with.

#### **Topics**

- SAS tokens
- Access tiers

#### **SAS** tokens

A SAS token specifies the access permissions for your blob storage. (For more information about SAS, see the Azure Blob Storage documentation.)

You can generate SAS tokens to provide different levels of access. DataSync supports tokens with the following access levels:

Account

#### Container

The access permissions that DataSync needs depends on the scope of your token. Not having the correct permissions can cause your transfer to fail. For example, your transfer won't succeed if you're moving objects with tags to Azure Blob Storage but your SAS token doesn't have tag permissions.

#### **Topics**

- SAS token permissions for account-level access
- SAS token permissions for container-level access
- SAS expiration policies

#### SAS token permissions for account-level access

DataSync needs an account-level access token with the following permissions (depending on whether you're transferring to or from Azure Blob Storage).

Transfers from blob storage

- Allowed services Blob
- Allowed resource types Container, Object

If you don't include these permissions, DataSync can't transfer your object metadata, including object tags.

- Allowed permissions Read, List
- Allowed blob index permissions Read/Write (if you want DataSync to copy object tags)

#### Transfers to blob storage

- Allowed services Blob
- Allowed resource types Container, Object

If you don't include these permissions, DataSync can't transfer your object metadata, including object tags.

• Allowed permissions – Read, Write, List, Delete (if you want DataSync to remove files that aren't in your transfer source)

Allowed blob index permissions – Read/Write (if you want DataSync to copy object tags)

#### SAS token permissions for container-level access

DataSync needs a container-level access token with the following permissions (depending on whether you're transferring to or from Azure Blob Storage).

Transfers from blob storage

- Read
- List
- Tag (if you want DataSync to copy object tags)



#### Note

You can't add the tag permission when generating a SAS token in the Azure portal. To add the tag permission, instead generate the token by using the Azure Storage Explorer app or generate a SAS token that provides account-level access.

#### Transfers to blob storage

- Read
- Write
- List
- Delete (if you want DataSync to remove files that aren't in your transfer source)
- Tag (if you want DataSync to copy object tags)



#### Note

You can't add the tag permission when generating a SAS token in the Azure portal. To add the tag permission, instead generate the token by using the Azure Storage Explorer app or generate a SAS token that provides account-level access.

#### SAS expiration policies

Make sure that your SAS doesn't expire before you expect to finish your transfer. For information about configuring a SAS expiration policy, see the Azure Blob Storage documentation.

If the SAS expires during the transfer, DataSync can no longer access your Azure Blob Storage location. (You might see a Failed to open directory error.) If this happens, <u>update your location</u> with a new SAS token and restart your DataSync task.

#### **Access tiers**

When transferring from Azure Blob Storage, DataSync can copy objects in the hot and cool tiers. For objects in the archive access tier, you must rehydrate those objects to the hot or cool tier before you can copy them.

When transferring to Azure Blob Storage, DataSync can copy objects into the hot, cool, and archive access tiers. If you're copying objects into the archive access tier, DataSync can't verify the transfer if you're trying to verify all data in the destination.

DataSync doesn't support the cold access tier. For more information about access tiers, see the Azure Blob Storage documentation.

### **Considerations with Azure Blob Storage transfers**

When planning to transfer data to or from Azure Blob Storage with DataSync, there are some things to keep in mind.

#### **Topics**

- Costs
- Blob types
- AWS Region availability
- Copying object tags
- Transferring to Amazon S3
- Deleting directories in a transfer destination
- Limitations

#### Costs

The fees associated with moving data in or out of Azure Blob Storage can include:

- Running an Azure virtual machine (VM) (if you deploy a DataSync agent in Azure)
- Running an Amazon EC2 instance (if you deploy a DataSync agent in a VPC within AWS)
- Transferring the data by using <u>DataSync</u>, including request charges related to <u>Azure Blob Storage</u> and <u>Amazon S3</u> (if S3 is one of your transfer locations)
- Transferring data in or out of Azure Blob Storage
- Storing data in an AWS storage service supported by DataSync

#### **Blob types**

How DataSync works with blob types depends on whether you're transferring to or from Azure Blob Storage. When you're moving data into blob storage, the objects or files that DataSync transfers can only be block blobs. When you're moving data out of blob storage, DataSync can transfer block, page, and append blobs.

For more information about blob types, see the Azure Blob Storage documentation.

#### **AWS Region availability**

You can create an Azure Blob Storage transfer location in any <u>AWS Region that's supported by</u> DataSync.

#### **Copying object tags**

The ability for DataSync to preserve object tags when transferring to or from Azure Blob Storage depends on the following factors:

- The size of an object's tags DataSync can't transfer an object with tags that exceed 2 KB.
- Whether DataSync is configured to copy object tags DataSync copies object tags by default.
- The namespace that your Azure storage account uses DataSync can copy object tags if your Azure storage account uses a flat namespace but not if your account uses a hierarchical namespace (a feature of Azure Data Lake Storage Gen2). Your DataSync task will fail if you try to copy object tags and your storage account uses a hierarchical namespace.
- Whether your SAS token authorizes tagging The permissions that you need to copy object
  tags vary depending on the level of access that your token provides. Your task will fail if you
  try to copy object tags and your token doesn't have the right permissions for tagging. For more
  information, check the permission requirements for account-level access tokens or containerlevel access tokens.

#### **Transferring to Amazon S3**

When transferring to Amazon S3, DataSync won't transfer Azure Blob Storage objects larger than 5 TB or objects with metadata larger than 2 KB.

#### Deleting directories in a transfer destination

When transferring to Azure Blob Storage, DataSync can <u>remove objects in your blob storage that</u> <u>aren't present in your transfer source</u>. (You can configure this option by clearing the **Keep deleted files** setting in the DataSync console. Your SAS token must also have delete permissions.)

When you configure your transfer this way, DataSync won't delete directories in your blob storage if your Azure storage account is using a hierarchical namespace. In this case, you must manually delete the directories (for example, by using Azure Storage Explorer).

#### Limitations

Remember the following limitations when transferring data to or from Azure Blob Storage:

- DataSync <u>creates some directories</u> in a location to help facilitate your transfer. If Azure Blob
  Storage is a destination location and your storage account uses a hierarchical namespace, you
  might notice task-specific subdirectories (such as task-000011112222abcde) in the /.awsdatasync folder. DataSync typically deletes these subdirectories following a transfer. If that
  doesn't happen, you can delete these task-specific directories yourself as long as a task isn't
  running.
- DataSync doesn't support using a SAS token to access only a specific folder in your Azure Blob Storage container.
- You can't provide DataSync a user delegation SAS token for accessing your blob storage.

### **Creating your DataSync agent (optional)**

A DataSync agent is required only when transferring data between Azure Blob and Amazon EFS or Amazon FSx, or when using **Basic** mode tasks. You don't need an agent to transfer data between Azure Blob and Amazon S3 using **Enhanced** mode. This section describes how to deploy and activate an agent.



#### (i) Tip

Although you can deploy your agent on an Amazon EC2 instance, using a Microsoft Hyper-V agent might result in decreased network latency and more data compression.

#### Microsoft Hyper-V agents

You can deploy your DataSync agent directly in Azure with a Microsoft Hyper-V image.



#### (i) Tip

Before you continue, consider using a shell script that might help you deploy your Hyper-V agent in Azure quicker. You can get more information and download the code on GitHub. If you use the script, you can skip ahead to the section about Getting your agent's activation key.

#### **Topics**

- Prerequisites
- Downloading and preparing your agent
- Deploying your agent in Azure
- Getting your agent's activation key
- Activating your agent

### **Prerequisites**

To prepare your DataSync agent and deploy it in Azure, you must do the following:

- Enable Hyper-V on your local machine.
- Install PowerShell (including the Hyper-V Module).
- Install the Azure CLI.
- Install AzCopy.

#### Downloading and preparing your agent

Download an agent from the DataSync console. Before you can deploy the agent in Azure, you must convert it to a fixed-size virtual hard disk (VHD). For more information, see the <u>Azure</u> documentation.

#### To download and prepare your agent

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Agents**, and then choose **Create agent**.
- 3. For **Hypervisor**, choose **Microsoft Hyper-V**, and then choose **Download the image**.

The agent downloads in a .zip file that contains a .vhdx file.

- 4. Extract the . vhdx file on your local machine.
- 5. Open PowerShell and do the following:
  - a. Copy the following Convert-VHD cmdlet:

```
Convert-VHD -Path .\land local-path-to-vhdx-file\aws-datasync-2.0.1686143940.1- x86_64.xfs.gpt.vhdx `-DestinationPath .\land local-path-to-vhdx-file\aws-datasync-2016861439401- x86_64.vhd -VHDType Fixed
```

- b. Replace each instance of *local-path-to-vhdx-file* with the location of the .vhdx file on your local machine.
- c. Run the command.

Your agent is now a fixed-size VHD (with a .vhd file format) and ready to deploy in Azure.

### Deploying your agent in Azure

Deploying your DataSync agent in Azure involves:

- · Creating a managed disk in Azure
- Uploading your agent to that managed disk
- Attaching the managed disk to a Linux virtual machine

#### To deploy your agent in Azure

- 1. In PowerShell, go to the directory that contains your agent's . vhd file.
- 2. Run the 1s command and save the Length value (for example, 85899346432).

This is the size of your agent image in bytes, which you need when creating a managed disk that can hold the image.

- 3. Do the following to create a managed disk:
  - a. Copy the following Azure CLI command:

```
az disk create -n your-managed-disk `
-g your-resource-group `
-l your-azure-region `
--upload-type Upload `
--upload-size-bytes agent-size-bytes `
--sku standard_lrs
```

- b. Replace *your-managed-disk* with a name for your managed disk.
- c. Replace *your-resource-group* with the name of the Azure resource group that your storage account belongs to.
- d. Replace *your-azure-region* with the Azure region where your resource group is located.
- e. Replace *agent-size-bytes* with the size of your agent image.
- f. Run the command.

This command creates an empty managed disk with a <u>standard SKU</u> where you can upload your DataSync agent.

- 4. To generate a shared access signature (SAS) that allows write access to the managed disk, do the following:
  - a. Copy the following Azure CLI command:

```
az disk grant-access -n your-managed-disk `
-g your-resource-group `
--access-level Write `
--duration-in-seconds 86400
```

b. Replace your-managed-disk with the name of the managed disk that you created.

- c. Replace *your-resource-group* with the name of the Azure resource group that your storage account belongs to.
- d. Run the command.

In the output, take note of the SAS URI. You need this URI when uploading the agent to Azure.

The SAS allows you to write to the disk for up to an hour. This means that you have an hour to upload your agent to the managed disk.

- 5. To upload your agent to your managed disk in Azure, do the following:
  - a. Copy the following AzCopy command:

```
.\azcopy copy local-path-to-vhd-file sas-uri --blob-type PageBlob
```

- b. Replace *local-path-to-vhd-file* with the location of the agent's .vhd file on your local machine.
- c. Replace sas-uri with the SAS URI that you got when you ran the az disk grant-access command.
- d. Run the command.
- 6. When the agent upload finishes, revoke access to your managed disk. To do this, copy the following Azure CLI command:

```
az disk revoke-access -n your-managed-disk -g your-resource-group
```

- a. Replace *your-resource-group* with the name of the Azure resource group that your storage account belongs to.
- b. Replace *your-managed-disk* with the name of the managed disk that you created.
- c. Run the command.
- 7. Do the following to attach your managed disk to a new Linux VM:
  - a. Copy the following Azure CLI command:

```
az vm create --resource-group your-resource-group `
--location eastus `
```

```
--name your-agent-vm `
--size Standard_E4as_v4 `
--os-type linux `
--attach-os-disk your-managed-disk
```

b. Replace *your-resource-group* with the name of the Azure resource group that your storage account belongs to.

- c. Replace your-agent-vm with a name for the VM that you can remember.
- d. Replace *your-managed-disk* with the name of the managed disk that you're attaching to the VM.
- e. Run the command.

You've deployed your agent. Before you can start configuring your data transfer, you must activate the agent.

### Getting your agent's activation key

To manually get your DataSync agent's activation key, follow these steps.

Alternatively, <u>DataSync can automatically get the activation key for you</u>, but this approach requires some network configuration.

#### To get your agent's activation key

1. In the Azure portal, <u>enable boot diagnostics for the VM for your agent</u> by choosing the **Enable** with custom storage account setting and specifying your Azure storage account.

After you've enabled the boot diagnostics for your agent's VM, you can access your agent's local console to get the activation key.

- 2. While still in the Azure portal, go to your VM and choose **Serial console**.
- 3. In the agent's local console, log in by using the following default credentials:
  - Username admin
  - Password password

We recommend at some point changing at least the agent's password. In the agent's local console, enter 5 on the main menu, then use the password command to change the password.

4. Enter **0** to get the agent's activation key.

- 5. Enter the AWS Region where you're using DataSync (for example, us-east-1).
- 6. Choose the service endpoint that the agent will use to connect with AWS.
- 7. Save the value of the Activation key output.

#### **Activating your agent**

After you have the activation key, you can finish creating your DataSync agent.

#### To activate your agent

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose Agents, and then choose Create agent.
- 3. For **Hypervisor**, choose **Microsoft Hyper-V**.
- 4. For **Endpoint type**, choose the same type of service endpoint that you specified when you got your agent's activation key (for example, choose **Public service endpoints in** *Region name*).
- 5. Configure your network to work with the service endpoint type that your agent is using. For service endpoint network requirements, see the following topics:
  - VPC endpoints
  - · Public endpoints
  - Federal Information Processing Standard (FIPS) endpoints
- 6. For **Activation key**, do the following:
  - a. Choose Manually enter your agent's activation key.
  - b. Enter the activation key that you got from the agent's local console.
- 7. Choose **Create agent**.

Your agent is ready to connect with your Azure Blob Storage. For more information, see <u>Creating</u> your Azure Blob Storage transfer location.

#### **Amazon EC2 agents**

You can deploy your DataSync agent on an Amazon EC2 instance.

#### To create an Amazon EC2 agent

Deploy an Amazon EC2 agent.

2. Choose a service endpoint that the agent uses to communicate with AWS.

In this situation, we recommend using a virtual private cloud (VPC) service endpoint.

- Configure your network to work with VPC service endpoints.
- 4. Activate the agent.

### **Creating your Azure Blob Storage transfer location**

You can configure DataSync to use your Azure Blob Storage as a transfer source or destination.

#### Before you begin

Make sure that you know <u>how DataSync accesses Azure Blob Storage</u> and works with <u>access tiers</u> and <u>blob types</u>. You also need a <u>DataSync agent</u> that can connect to your Azure Blob Storage container.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For Location type, choose Microsoft Azure Blob Storage.
- 4. For **Container URL**, enter the URL of the container that's involved in your transfer.
- 5. (Optional) For **Access tier when used as a destination**, choose the <u>access tier</u> that you want your objects or files transferred into.
- 6. For **Folder**, enter path segments if you want to limit your transfer to a virtual directory in your container (for example, /my/images).
- 7. If your transfer requires an agent, choose **Use agents**, then choose the DataSync agent that can connect with your Azure Blob Storage container.
- 8. For **SAS token**, provide the credentials necessary for DataSync to access your blob storage. Some public datasets on Azure Blob storage do not require credentials. You can enter a SAS token directly, or specify an AWS Secrets Manager secret that contains the token. For more information, see Providing credentials for storage locations.

Your SAS token is part of the SAS URI string that comes after your storage resource URI and a question mark (?). A token looks something like this:

9. (Optional) Enter values for the **Key** and **Value** fields to tag the location.

Tags help you manage, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

10. Choose Create location.

#### **Using the AWS CLI**

Copy the following create-location-azure-blob command:

```
aws datasync create-location-azure-blob \
--container-url "https://path/to/container" \
--authentication-type "SAS" \
--sas-configuration '{
    "Token": "your-sas-token"
    }' \
--agent-arns my-datasync-agent-arn \
--subdirectory "/path/to/my/data" \
--access-tier "access-tier-for-destination" \
--tags [{"Key": "key1","Value": "value1"}]
```

- For the --container-url parameter, specify the URL of the Azure Blob Storage container that's involved in your transfer.
- 3. For the --authentication-type parameter, specify SAS. If you are accessing a public dataset that does not require authentication, specify NONE.
- 4. For the --sas-configuration parameter's Token option, specify the SAS token that allows DataSync to access your blob storage.

You can also provide additional parameters for securing your keys using AWS Secrets Manager. For more information, see Providing credentials for storage locations.

Your SAS token is part of the SAS URI string that comes after your storage resource URI and a question mark (?). A token looks something like this:

5. (Optional) For the --agent-arns parameter, specify the Amazon Resource Name (ARN) of the DataSync agent that can connect to your container.

```
Here's an example agent ARN: arn:aws:datasync:us-east-1:123456789012:agent/agent-01234567890aaabfb
```

You can specify more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

- 6. For the --subdirectory parameter, specify path segments if you want to limit your transfer to a virtual directory in your container (for example, /my/images).
- 7. (Optional) For the --access-tier parameter, specify the <u>access tier</u> (HOT, COOL, or ARCHIVE) that you want your objects or files transferred into.

This parameter applies only when you're using this location as a transfer destination.

(Optional) For the --tags parameter, specify key-value pairs that can help you manage, filter, and search for your location.

We recommend creating a name tag for your location.

9. Run the create-location-azure-blob command.

If the command is successful, you get a response that shows you the ARN of the location that you created. For example:

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:123456789012:location/
loc-12345678abcdefgh"
}
```

# **Viewing your Azure Blob Storage transfer location**

You can get details about the existing DataSync transfer location for your Azure Blob Storage.

#### Using the DataSync console

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.

- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose your Azure Blob Storage location.

You can see details about your location, including any DataSync transfer tasks that are using it.

#### **Using the AWS CLI**

1. Copy the following describe-location-azure-blob command:

```
aws datasync describe-location-azure-blob \
   --location-arn "your-azure-blob-location-arn"
```

- 2. For the --location-arn parameter, specify the ARN for the Azure Blob Storage location that you created (for example, arn:aws:datasync:us-east-1:123456789012:location/loc-12345678abcdefgh).
- 3. Run the describe-location-azure-blob command.

You get a response that shows you details about your location. For example:

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:123456789012:location/
loc-12345678abcdefgh",
    "LocationUri": "azure-blob://my-user.blob.core.windows.net/container-1",
    "AuthenticationType": "SAS",
    "Subdirectory": "/my/images",
    "AgentArns": ["arn:aws:datasync:us-east-1:123456789012:agent/
agent-01234567890deadfb"],
}
```

### **Updating your Azure Blob Storage transfer location**

If needed, you can modify your location's configuration in the console or by using the AWS CLI.

### **Using the AWS CLI**

1. Copy the following update-location-azure-blob command:

```
aws datasync update-location-azure-blob \
    --location-arn "your-azure-blob-location-arn" \
    --authentication-type "SAS" \
```

```
--sas-configuration '{
    "Token": "your-sas-token"
}' \
--agent-arns my-datasync-agent-arn \
--subdirectory "/path/to/my/data" \
--access-tier "access-tier-for-destination"
```

2. For the --location-arn parameter, specify the ARN for the Azure Blob Storage location that you're updating (for example, arn:aws:datasync:us-east-1:123456789012:location/loc-12345678abcdefgh).

- 3. For the --authentication-type parameter, specify SAS.
- 4. For the --sas-configuration parameter's Token option, specify the SAS token that allows DataSync to access your blob storage.

The token is part of the SAS URI string that comes after the storage resource URI and a question mark (?). A token looks something like this:

```
sp=r&st=2022-12-20T14:54:52Z&se=2022-12-20T22:54:52Z&spr=https&sv=2021-06-08&sr=c&sig=qCBKE %2FXTI9E%2F%2Fmq171%2BZU178wcwqU%3D
```

5. For the --agent-arns parameter, specify the Amazon Resource Name (ARN) of the DataSync agent that you want to connect to your container.

Here's an example agent ARN: arn: aws:datasync: us-east-1:123456789012: agent/agent-01234567890aaabfb

You can specify more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

- 6. For the --subdirectory parameter, specify path segments if you want to limit your transfer to a virtual directory in your container (for example, /my/images).
- 7. (Optional) For the --access-tier parameter, specify the <u>access tier</u> (HOT, COOL, or ARCHIVE) that you want your objects to be transferred into.

This parameter applies only when you're using this location as a transfer destination.

### **Next steps**

After you finish creating a DataSync location for your Azure Blob Storage, you can continue setting up your transfer. Here are some next steps to consider:

1. If you haven't already, create another location where you plan to transfer your data to or from your Azure Blob Storage.

- 2. Learn how DataSync handles metadata and special files, particularly if your transfer locations don't have a similar metadata structure.
- 3. Configure how your data gets transferred. For example, you can transfer only a subset of your data or delete files in your blob storage that aren't in your source location (as long as your SAS token has delete permissions).
- 4. Start your transfer.

# Configuring AWS DataSync transfers with Microsoft Azure Files SMB shares

You can configure AWS DataSync to transfer data to or from a Microsoft Azure Files Server Message Block (SMB) share.



#### (i) Tip

For a full walkthrough on moving data from Azure Files SMB shares to AWS, see the AWS Storage Blog.

# **Providing DataSync access to SMB shares**

DataSync connects to your SMB share using the SMB protocol and authenticates with credentials that you provide it.

### **Topics**

- Supported SMB protocol versions
- Required permissions

### **Supported SMB protocol versions**

By default, DataSync automatically chooses a version of the SMB protocol based on negotiation with your SMB file server.

You also can configure DataSync to use a specific SMB version, but we recommend doing this only if DataSync has trouble negotiating with the SMB file server automatically. DataSync supports

SMB versions 1.0 and later. For security reasons, we recommend using SMB version 3.0.2 or later. Earlier versions, such as SMB 1.0, contain known security vulnerabilities that attackers can exploit to compromise your data.

See the following table for a list of options in the DataSync console and API:

Console option	API option	Description
Automatic	AUTOMATIC	DataSync and the SMB file server negotiate the highest version of SMB that they mutually support between 2.1 and 3.1.1.  This is the default and recommended option. If you
		instead choose a specific version that your file server doesn't support, you may get an Operation Not Supported error.
SMB 3.0.2	SMB3	Restricts the protocol negotiation to only SMB version 3.0.2.
SMB 2.1	SMB2	Restricts the protocol negotiation to only SMB version 2.1.
SMB 2.0	SMB2_0	Restricts the protocol negotiation to only SMB version 2.0.
SMB 1.0	SMB1	Restricts the protocol negotiation to only SMB version 1.0.

#### **Required permissions**

DataSync needs a user who has permission to mount and access your SMB location. This can be a local user on your Windows file server or a domain user that's defined in your Microsoft Active Directory.

To set object ownership, DataSync requires the SE\_RESTORE\_NAME privilege, which is usually granted to members of the built-in Active Directory groups **Backup Operators** and **Domain Admins**. Providing a user to DataSync with this privilege also helps ensure sufficient permissions to files, folders, and file metadata, except for NTFS system access control lists (SACLs).

Additional privileges are required to copy SACLs. Specifically, this requires the Windows SE SECURITY NAME privilege, which is granted to members of the **Domain Admins** group. If you configure your task to copy SACLs, make sure that the user has the required privileges. To learn more about configuring a task to copy SACLs, see Configuring how to handle files, objects, and metadata.

When you copy data between an SMB file server and Amazon FSx for Windows File Server file system, the source and destination locations must belong to the same Microsoft Active Directory domain or have an Active Directory trust relationship between their domains.

### Creating your Azure Files transfer location by using the console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For Location type, choose Server Message Block (SMB).

You configure this location as a source or destination later.

- For **Agents**, choose one or more DataSync agents that you want to connect to your SMB share.
  - If you choose more than one agent, make sure you understand using multiple agents for a location.
- 5. For **SMB Server**, enter the Domain Name System (DNS) name or IP address of the SMB share that your DataSync agent will mount.



#### Note

You can't specify an IP version 6 (IPv6) address.

- For **Share name**, enter the name of the share exported by your SMB share where DataSync will read or write data.
  - You can include a subdirectory in the share path (for example, /path/to/subdirectory). Make sure that other SMB clients in your network can also mount this path.
  - To copy all the data in the subdirectory, DataSync must be able to mount the SMB share and access all of its data. For more information, see Required permissions.
- (Optional) Expand Additional settings and choose an SMB Version for DataSync to use when accessing your SMB share.

By default, DataSync automatically chooses a version based on negotiation with the SMB share. For information, see Supported SMB versions.

For User, enter a user name that can mount your SMB share and has permission to access the files and folders involved in your transfer.

For more information, see Required permissions.

- 9. For **Password**, enter the password of the user who can mount your SMB share and has permission to access the files and folders involved in your transfer.
- 10. (Optional) For **Domain**, enter the Windows domain name that your SMB share belongs to.

If you have multiple domains in your environment, configuring this setting makes sure that DataSync connects to the right share.

11. (Optional) Choose Add tag to tag your location.

Tags are key-value pairs that help you manage, filter, and search for your locations. We recommend creating at least a name tag for your location.

12. Choose Create location.

# Configuring transfers with other cloud object storage

With AWS DataSync, you can transfer data between <u>AWS storage services</u> and the following cloud object storage providers:

- Wasabi Cloud Storage
- DigitalOcean Spaces
- Oracle Cloud Infrastructure Object Storage
- Cloudflare R2 Storage
- Backblaze B2 Cloud Storage
- NAVER Cloud Object Storage
- Alibaba Cloud Object Storage Service
- IBM Cloud Object Storage
- Seagate Lyve Cloud

A DataSync agent is required only when transferring data between storage systems in other clouds and Amazon EFS or Amazon FSx, or when using **Basic** mode tasks. You don't need an agent to transfer data between storage systems in other clouds and Amazon S3 using **Enhanced** mode.

Regardless of whether you use an agent, you must also create a transfer <u>location</u> for your cloud object storage (specifically an **Object storage** location). DataSync can use this location as a source or destination for your transfer.

#### Providing DataSync access to your other cloud object storage

How DataSync accesses your cloud object storage depends on several factors, including whether your storage is compatible with the Amazon S3 API and the permissions and credentials that DataSync needs to access your storage.

#### **Topics**

- Amazon S3 API compatibility
- Storage permissions and endpoints
- Storage credentials

#### **Amazon S3 API compatibility**

Your cloud object storage must be compatible with the following <u>Amazon S3 API operations</u> for DataSync to connect to it:

- AbortMultipartUpload
- CompleteMultipartUpload
- CopyObject
- CreateMultipartUpload
- DeleteObject
- DeleteObjects
- DeleteObjectTagging
- GetBucketLocation
- GetObject
- GetObjectTagging

- HeadBucket
- HeadObject
- ListObjectsV2
- PutObject
- PutObjectTagging
- UploadPart

#### Storage permissions and endpoints

You must configure the permissions that allow DataSync to access your cloud object storage. If your object storage is a source location, DataSync needs read and list permissions for the bucket that you're transferring data from. If your object storage is a destination location, DataSync needs read, list, write, and delete permissions for the bucket.

DataSync also needs an endpoint (or server) to connect to your storage. The following table describes the endpoints that DataSync can use to access other cloud object storage:

Other cloud provider	Endpoint	
Wasabi Cloud Storage	S3. region.wasabisys.com	
DigitalOcean Spaces	region.digitaloceanspaces.com	
Oracle Cloud Infrastructure Object Storage	<pre>namespace .compat.objectstor age. region.oraclecloud.com</pre>	
Cloudflare R2 Storage	<pre>account-id .r2.cloudflarestor age.com</pre>	
Backblaze B2 Cloud Storage	S3. region.backblazeb2.com	
NAVER Cloud Object Storage	<pre>region.object.ncloudstorage.com (most regions)</pre>	
Alibaba Cloud Object Storage Service	region.aliyuncs.com	
IBM Cloud Object Storage	s3. region.cloud-object-stor age.appdomain.cloud	

Other cloud provider	Endpoint	
Seagate Lyve Cloud	s3. region.lyvecloud.seagate.com	

#### 

For details on how to configure bucket permissions and updated information on storage endpoints, see your cloud provider's documentation.

#### Storage credentials

DataSync also needs the credentials to access the object storage bucket involved in your transfer. This might be an access key and secret key or something similar depending on how your cloud storage provider refers to these credentials.

For more information, see your cloud provider's documentation.

### Considerations when transferring from other cloud object storage

When planning to transfer objects to or from another cloud storage provider by using DataSync, there are some things to keep in mind.

#### **Topics**

- Costs
- Storage classes
- Object tags
- Transferring to Amazon S3

#### Costs

The fees associated with moving data in and out of another cloud storage provider can include:

- Running an Amazon EC2 instance for your DataSync agent
- Transferring the data by using DataSync, including request charges related to your cloud object storage and Amazon S3 (if S3 is your transfer destination)
- Transferring data in or out of your cloud storage (check your cloud provider's pricing)

- Storing data in an AWS storage service supported by DataSync
- Storing data in another cloud provider (check your cloud provider's pricing)

#### Storage classes

Some cloud storage providers have storage classes (similar to <u>Amazon S3</u>) which DataSync can't read without being restored first. For example, Oracle Cloud Infrastructure Object Storage has an archive storage class. You need to restore objects in that storage class before DataSync can transfer them. For more information, see your cloud provider's documentation.

#### **Object tags**

Not all cloud providers support object tags. The ones that do might not allow querying tags through the Amazon S3 API. In either situation, your DataSync transfer task might fail if you try to copy object tags.

You can avoid this by clearing the **Copy object tags** checkbox in the DataSync console when creating, starting, or updating your task.

#### **Transferring to Amazon S3**

When transferring to Amazon S3, DataSync can't transfer objects larger than 5 TB. DataSync also can only copy object metadata up to 2 KB.

# **Creating your DataSync agent**

A DataSync agent is required only when transferring data between storage systems in other clouds and Amazon EFS or Amazon FSx, or when using **Basic** mode tasks. You don't need an agent to transfer data between storage systems in other clouds and Amazon S3 using **Enhanced** mode. This section desribes how to deploy and activate an agent on an Amazon EC2 instance in your virtual private cloud (VPC) in AWS.

#### To create an Amazon EC2 agent

- Deploy an Amazon EC2 agent.
- 2. Choose a service endpoint that the agent uses to communicate with AWS.

In this situation, we recommend using a VPC service endpoint.

Configure your network to work with <u>VPC service endpoints</u>.

#### 4. Activate the agent.

### Creating a transfer location for your other cloud object storage

You can configure DataSync to use your cloud object storage as a source or destination location.

### Before you begin

Make sure that you know <u>how DataSync accesses your cloud object storage</u>. You also need a DataSync agent that can connect to your cloud object storage.

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations** and **Create location**.
- 3. For Location type, choose Object storage.
- 4. For **Server**, enter the <u>endpoint</u> that DataSync can use to access your cloud object storage:
  - Wasabi Cloud Storage S3. region. wasabisys.com
  - **DigitalOcean Spaces** *region*.digitaloceanspaces.com
  - Oracle Cloud Infrastructure Object Storage –
     namespace.compat.objectstorage.region.oraclecloud.com
  - Cloudflare R2 Storage account-id.r2.cloudflarestorage.com
  - Backblaze B2 Cloud Storage S3. region. backblazeb2.com
  - NAVER Cloud Object Storage region.object.ncloudstorage.com (most regions)
  - Alibaba Cloud Object Storage Service region. aliyuncs.com
  - IBM Cloud Object Storage s3. region.cloud-object-storage.appdomain.cloud
  - Seagate Lyve Cloud s3. region.lyvecloud.seagate.com
- 5. For **Bucket name**, enter the name of the object storage bucket that you're transferring data to or from.
- 6. For **Folder**, enter an object prefix. DataSync only transfers objects with this prefix.
- 7. If your transfer requires an agent, choose **Use agents**, then choose the DataSync agent that can connect with your cloud object storage.
- 8. Expand Additional settings. For Server protocol, choose HTTPS. For Server port, choose 443.
- 9. Scroll down to the **Authentication** section. Make sure that the **Requires credentials** check box is selected, and then provide DataSync your <u>storage credentials</u>.

- For Access key, enter the ID to access your cloud object storage.
- For **Secret key**, provide the secret key to access your cloud object storage. You can either enter the key directly, or specify an AWS Secrets Manager secret that contains the key. For more information, see Providing credentials for storage locations.
- 10. (Optional) Enter values for the **Key** and **Value** fields to tag the location.

Tags help you manage, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

11. Choose Create location.

### **Next steps**

After you finish creating a DataSync location for your cloud object storage, you can continue setting up your transfer. Here are some next steps to consider:

- 1. If you haven't already, create another location where you plan to transfer your data to or from in AWS.
- 2. Learn how DataSync handles metadata and special files for object storage locations.
- 3. Configure how your data gets transferred. For example, maybe you only want to transfer a subset of your data.



#### Important

Make sure that you configure how DataSync copies object tags correctly. For more information, see considerations with object tags.

4. Start your transfer.

# Transferring to or from S3 compatible storage on Snowball **Edge**

With AWS DataSync, you can transfer objects between Amazon S3 compatible storage on an AWS Snowball Edge device or cluster and any of the following AWS storage services:

Amazon S3

- Amazon Elastic File System (Amazon EFS)
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

# **Prerequisites**

Before you get started, make sure that you've done the following:

- Created an AWS storage resource in the AWS Region where you plan to transfer data to or from. For example, this could be an S3 bucket or Amazon EFS file system in US East (N. Virginia).
- Established a wide-area network (WAN) connection for traffic into and out of your on-premises storage environment. For example, you can establish this kind of connection with <u>AWS Direct</u> Connect.

When you <u>create your DataSync agent</u>, you'll configure this WAN connection so that DataSync can transfer data between your Amazon S3 compatible storage that's on-premises and your storage resource in AWS.

Downloaded and installed the Snowball Edge client.

# Providing DataSync access to S3 compatible storage

To access your Amazon S3 compatible storage bucket, DataSync needs the following:

- User credentials on your Snowball Edge device or cluster that can access the bucket that you're transferring data to or from.
- An HTTPS certificate that allows DataSync to verify the authenticity of the connection between the DataSync agent and the s3api endpoint on your device or cluster.

#### **Topics**

- Getting the user credentials to access your S3 bucket
- Getting a certificate for the s3api endpoint connection

Prerequisites 191

### Getting the user credentials to access your S3 bucket

DataSync needs the access key and secret key for a user who can access the bucket that you're working with on your Snowball Edge device or cluster.

#### To get the user credentials to access your bucket

Open a terminal and run the Snowball Edge client.

For more information about running the Snowball Edge client, see <u>Using the Snowball Edge</u> client in the *AWS Snowball Edge Developer Guide*.

2. To get the access keys associated with your device or cluster, run the following snowballEdge command:

```
snowballEdge list-access-keys
```

- In the output, locate the access key for the bucket that DataSync will work with (for example, AKIAIOSFODNN7EXAMPLE).
- 4. To get the secret access key, run the following snowballEdge command. Replace accesskey-for-datasync with the access key that you located in the prior step.

```
snowballEdge get-secret-access-key --access-key-id access-key-for-datasync
```

The output includes the access key's corresponding secret key (for example, wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY).

5. Save the access key and secret key somewhere that you can remember.

You will need these keys when you're <u>configuring the DataSync source location</u> for your transfer.

### Getting a certificate for the s3api endpoint connection

You need an HTTPS certificate that can verify the authenticity of the connection between your DataSync agent and an s3api endpoint on your Snowball Edge device or cluster.

#### To get a certificate for the s3api endpoint connection

1. In the Snowball Edge client, run the following list-certificates command:

```
snowballEdge list-certificates
```

In the output, take note of the CertificateArn value. This is the certificate's Amazon Resource Name (ARN). You need the ARN to get the certificate's contents.

2. Run the following get-certificate command that specifies the certificate ARN that you just retrieved:

```
snowballEdge get-certificate --certificate-arn arn:aws:snowball-
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7
```

3. Copy the output, including the BEGIN CERTIFICATE and END CERTIFICATE lines, and save it as a .pem file.

#### Example of get-certificate output:

```
----BEGIN CERTIFICATE----

Certificate
----END CERTIFICATE----
```

You specify this . pem file when creating the DataSync source location for your transfer.

# Creating a DataSync agent in your on-premises storage environment

During a transfer, DataSync uses an <u>agent</u> to read from or write to the Amazon S3 compatible storage on your Snowball Edge device or cluster.

This agent must be deployed in your on-premises storage environment where it can connect to your device or cluster through your network. For example, you can run the agent on a VMware ESXi hypervisor that has local network access to your cluster.

#### To create a DataSync agent in your on-premises storage environment

- 1. Make sure that the <u>DataSync agent can run on your hypervisor</u> and that you <u>allocate the agent enough virtual machine (VM) resources</u>.
- 2. Deploy the agent in your on-premises environment.

For instructions, see one of the following topics, depending on the type of hypervisor that you're deploying the agent on:

- Deploy your agent on VMware
- Deploy your agent on Linux Kernel-based Machine (KVM)
- Deploy your agent on Microsoft Hyper-V
- Deploy your agent on Amazon EC2



#### Marning

We don't recommend deploying an agent on Amazon EC2 agent to access onpremises storage because of increased network latency.

Configure your network to allow the following traffic between the agent and your Amazon S3 compatible storage:

From	То	Protocol and port
DataSync agent	A virtual network interface (VNI) for an s3api endpoint on your device or cluster. If you have a cluster, it can be any s3api endpoint VNI.	TCP 443 (HTTPS)

If you need to find a VNI on your device or cluster, see describing your virtual network interfaces on Snowball Edge.

- Choose a service endpoint that the agent uses to communicate with the DataSync service. 4.
- 5. Activate your agent.

# Configuring the source location for your transfer

After you create your agent, you can configure the source location for your DataSync transfer.



#### Note

The following instructions assume that you're transferring from Amazon S3 compatible storage, but you can also use this location for a transfer destination.

#### To configure the source location by using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**. Choose **Tasks**, and then choose **Create task**.
- 3. On the **Configure source location** page, choose **Create a new location**.
- 4. For **Location type**, choose **Object storage**.
- 5. For **Agents**, choose the DataSync agent that you created in your on-premises storage environment.
- 6. For **Server**, enter the VNI for the s3api endpoint that's used by your Amazon S3 compatible storage.
  - If you have a Snowball Edge cluster instead of a single device, you can specify any of the cluster's s3api endpoint VNIs.
- 7. For **Bucket name**, enter the name of the Amazon S3 compatible storage bucket that you're transferring objects from.
- 8. For **Folder**, enter an object prefix.
  - DataSync only transfers objects with this prefix.
- 9. To configure the DataSync connection to the Snowball Edge device or cluster, expand **Additional settings** and do the following:
  - a. For **Server protocol**, choose **HTTPS**.
  - b. For **Server port**, enter **443**.
  - c. For **Certificate**, choose the certificate file for the s3api endpoint connection.
- 10. Select **Requires credentials**, and enter the **Access key** and **Secret key** to <u>access the Amazon S3</u> compatible storage bucket on your Snowball Edge device or cluster.
- 11. Choose Next.

# Configuring the destination location for your transfer

Your transfer's destination location must be in the same AWS Region and AWS account where you created your agent.

**Before you begin**: Make sure you've configured the source location for your transfer.

#### To configure the destination location for your transfer by using the DataSync console

 On the Configure destination location page, choose Create a new location or Choose an existing location for the AWS storage resource where you're transferring objects to.

If you're creating a new location, see one of the following topics:

- Amazon S3
- Amazon EFS
- FSx for Windows File Server
- FSx for Lustre
- FSx for OpenZFS
- FSx for ONTAP
- 2. When you're done configuring the destination location, choose **Next**.

# **Configuring your transfer settings**

With DataSync, you can specify a transfer schedule, customize how your data integrity is verified, and specify whether you want to transfer only a subset of objects, among other options.

Before you begin: Make sure you've configured the destination location for your transfer.

#### To configure your transfer settings by using the DataSync console

1. On the **Configure settings** page, change the transfer settings or use the defaults.

For more information about these settings, see Choosing what AWS DataSync transfers.

- 2. Choose Next.
- 3. Review your transfer details, and then choose **Create task**.

# Starting your transfer

After you create your transfer task, you're ready to start moving data. For instructions on starting a task by using the DataSync console or AWS CLI, see <u>Starting your task</u>.

### Limitations

 If your source storage system uses the NFS protocol (such as Amazon EFS), DataSync can't transfer files with hard links to a Snowball Edge device.

• DataSync can't transfer objects that are longer than 1,024 bytes from a Snowball Edge device to an S3 bucket. For more information, see the Amazon S3 User Guide.

# Creating a task for transferring your data

A task describes where and how AWS DataSync transfers data. A task consists of the following:

- Source location The storage system or service where DataSync transfers data from.
- Destination location The storage system or service where DataSync transfers data to.
- Task options Settings such as what files to transfer, how data gets verified, when the task runs, and more.
- Task executions When you run a task, it's called a task execution.

# **Creating your task**

When you create a DataSync task, you specify your source and destination locations. You also can customize your task by choosing which files to transfer, how metadata gets handled, setting up a schedule, and more.

Before you create your task, make sure that you understand how DataSync transfers work and review the task quotas.



#### Important

If you're planning to transfer data to or from an Amazon S3 location, review how DataSync can affect your S3 request charges and the DataSync pricing page before you begin.

#### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- Make sure you're in one of the AWS Regions where you plan to transfer data. 2.

Limitations 197

3. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.

- 4. On the **Configure source location** page, create or choose a source location, then choose **Next**.
- 5. On the **Configure destination location** page, <u>create</u> or choose a destination location, then choose **Next**.
- 6. (Recommended) On the **Configure settings** page, give your task a name that you can remember.
- 7. While still on the **Configure settings** page, choose your task options or use the default settings.

You might be interested in some of the following options:

- Specify the task mode that you want to use.
- Specify what data to transfer by using a <u>manifest</u> or <u>filters</u>.
- Configure how to handle file metadata and verify data integrity.
- Monitor your transfer with <u>task reports</u> or <u>Amazon CloudWatch</u>. We recommend setting up some kind of monitoring for your task.

When you're done, choose Next.

8. Review your task configuration, then choose **Create task**.

You're ready to start your task.

#### Using the AWS CLI

Once you create your DataSync source and destination locations, you can create your task.

- 1. In your AWS CLI settings, make sure that you're using one of the AWS Regions where you plan to transfer data.
- Copy the following create-task command:

```
aws datasync create-task \
    --source-location-arn "arn:aws:datasync:us-east-1:account-id:location/location-
id" \
    --destination-location-arn "arn:aws:datasync:us-east-1:account-
id:location/location-id" \
```

Creating your task 198

```
--name "task-name"
```

3. For --source-location-arn, specify the Amazon Resource Name (ARN) of your source location.

4. For --destination-location-arn, specify the ARN of your destination location.

If you're transferring across AWS Regions or accounts, make sure that the ARN includes the other Region or account ID.

- 5. (Recommended) For --name, specify a name for your task that you can remember.
- 6. Specify other task options as needed. You might be interested in some of the following options:
  - Specify what data to transfer by using a manifest or filters.
  - Configure how to handle file metadata and verify data integrity.
  - Monitor your transfer with <u>task reports</u> or <u>Amazon CloudWatch</u>. We recommend setting up some kind of monitoring for your task.

For more options, see <u>create-task</u>. Here's an example <u>create-task</u> command that specifies several options:

```
aws datasync create-task \
    --source-location-arn "arn:aws:datasync:us-east-1:account-id:location/location-
id" \
    --destination-location-arn "arn:aws:datasync:us-east-1:account-
id:location/location-id" \
    --cloud-watch-log-group-arn "arn:aws:logs:region:account-id" \
    --name "task-name" \
    --options
VerifyMode=NONE,OverwriteMode=NEVER,Atime=BEST_EFFORT,Mtime=PRESERVE,Uid=INT_VALUE,Gid=INT_VALUE,Gid=INT_VALUE,Gid=INT_VALUE
```

7. Run the create-task command.

If the command is successful, you get a response that shows you the ARN of the task that you created. For example:

```
{
    "TaskArn": "arn:aws:datasync:us-east-1:111222333444:task/
    task-08de6e6697796f026"
}
```

Creating your task 199

You're ready to start your task.

#### Task statuses

When you create a DataSync task, you can check its status to see if it's ready to run.

Console status	API status	Description
Available	AVAILABLE	The task is ready to start transferring data.
Running	RUNNING	A task execution is in progress. For more information, see <u>Task execution statuses</u> .
Unavailable	UNAVAILABLE	A DataSync agent used by the task is offline. For more information, see What do I do if my agent is offline?
Queued	QUEUED	Another task execution that uses the same DataSync agent is in progress. For more information, see <a href="Knowing when your task is queued">Knowing when your task is queued</a> .

# Partitioning large datasets with multiple tasks

If you're transferring a large dataset, such as <u>migrating</u> millions of files or objects, we recommend partitioning your dataset with multiple DataSync tasks. Partitioning your source data across multiple tasks (and possibly <u>agents</u>, depending on your locations) helps reduce the time it takes DataSync to prepare and transfer your data.

Consider some of the ways that you can partition a large dataset across several DataSync tasks:

- Create tasks that transfer separate folders. For example, you might create two tasks that target /
  FolderA and /FolderB, respectively, in your source storage.
- Create tasks that transfer subsets of files, objects, and folders by using a <u>manifest</u> or <u>filters</u>.

Be mindful that this approach can increase the I/O operations on your storage and affect your network bandwidth. For more information, see the blog on <a href="How to accelerate your data transfers">How to accelerate your data transfers</a> with DataSync scale out architectures.

Task statuses 200

# Segmenting transferred data with multiple tasks

If you're transferring different sets of data to the same destination, you can create multiple tasks to help segment the data that you transfer.

For example, if you're transferring to the same S3 bucket named MyBucket, you can create different prefixes in the bucket that correspond to each task. This approach prevents file name conflicts the datasets and allows you to set different permissions for each prefix. Here's how you might set this up:

- Create three prefixes in the destination MyBucket named task1, task2, and task3:
  - s3://MyBucket/task1
  - s3://MyBucket/task2
  - s3://MyBucket/task3
- Create three DataSync tasks named task1, task2, and task3 that transfer to the corresponding prefix in MyBucket.

# Choosing a task mode for your data transfer

Your AWS DataSync task can run in one of the following modes:

- Enhanced mode Transfer virtually unlimited numbers of objects with higher performance than Basic mode. Enhanced mode tasks optimize the data transfer process by listing, preparing, transferring, and verifying data in parallel. Enhanced mode is currently available for transfers between Amazon S3 locations, transfers between Azure Blob and Amazon S3 without an agent, and transfers between other clouds and Amazon S3 without an agent.
- Basic mode Transfer files or objects between AWS storage and all other supported DataSync locations. Basic mode tasks are subject to <u>quotas</u> on the number of files, objects, and directories in a dataset. Basic mode sequentially prepares, transfers, and verifies data, making it slower than Enhanced mode for most workloads.

# **Understanding task mode differences**

The following information can help you determine which task mode to use.

Capability	Enhanced mode behavior	Basic mode behavior
<u>Performance</u>	DataSync lists, prepares, transfers, and verifies your data in parallel. Provides higher performance than Basic mode for most workloads (such as transferr ing large objects)	DataSync prepares, transfers , and verifies your data sequentially. Performance is slower than Enhanced mode for most workloads
Number of items in a dataset that DataSync can work with per task execution	Virtually unlimited numbers of objects	Quotas apply
Data transfer <u>counters</u> and <u>metrics</u>	More counters and metrics than Basic mode, such as the number of objects that DataSync finds at your source location and how many objects are prepared during each task execution	Less counters and metrics than Enhanced mode
Logging	Structured logs (JSON format)	Unstructured logs
Supported locations	Currently for transfers between Amazon S3 locations , transfers between Azure Blob and Amazon S3 without an agent, and transfers between other clouds and Amazon S3 without an agent.	For transfers between all locations that DataSync supports
Data verification options	DataSync verifies only transferred data	DataSync verifies all data by default
Bandwidth limits	Not applicable	Supported

Capability	Enhanced mode behavior	Basic mode behavior
Cost	For more information, see the DataSync pricing page	For more information, see the <u>DataSync pricing</u> page
Failure handling for unsupported object tags	For cloud storage transfers to or from locations that don't support object tagging, task execution will fail immediate ly if the ObjectTags option is unspeficied or set to PRESERVE.	For cloud storage transfers to or from locations that don't support object tagging, task execution will run normally, but will report per-object failures for tagged objects if the ObjectTags option is unspecified or set to PRESERVE.

# Choosing a task mode

You can choose Enhanced mode only for transfers between Amazon S3 locations, transfers between Azure Blob and Amazon S3 without an agent, and transfers between other clouds and Amazon S3 without an agent. Otherwise, you must use Basic mode. For example, a transfer from an on-premises NFS location to an S3 location requires Basic mode.

Your task options and performance might vary depending on the task mode you choose. Once you create your task, you can't change the task mode.

# **Required permissions**

To create an Enhanced mode task, the IAM role that you're using DataSync with must have the iam:CreateServiceLinkedRole permission.

For your DataSync user permissions, consider using <u>AWSDataSyncFullAccess</u>. This is an AWS managed policy that provides a user full access to DataSync and minimal access to its dependencies.

### Using the DataSync console

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.

2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.

3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- 4. For **Task mode**, choose one of the following options:
  - Enhanced
  - Basic

For more information, see Understanding task mode differences.

5. While still on the **Configure settings** page, choose other task options or use the default settings.

You might be interested in some of the following options:

- Specify what data to transfer by using a manifest or filters.
- Configure how to <u>handle file metadata</u> and <u>verify data integrity</u>.
- Monitor your transfer with task reports or Amazon CloudWatch Logs.

When you're done, choose Next.

6. Review your task configuration, then choose **Create task**.

# **Using the AWS CLI**

- 1. In your AWS CLI settings, make sure that you're using one of the AWS Regions where you plan to transfer data.
- 2. Copy the following create-task command:

```
aws datasync create-task \
    --source-location-arn "arn:aws:datasync:us-east-1:account-id:location/location-
id" \
    --destination-location-arn "arn:aws:datasync:us-east-1:account-
id:location/location-id" \
    --task-mode "ENHANCED-or-BASIC"
```

3. For --source-location-arn, specify the Amazon Resource Name (ARN) of your source location.

4. For --destination-location-arn, specify the ARN of your destination location.

If you're transferring across AWS Regions or accounts, make sure that the ARN includes the other Region or account ID.

5. For --task-mode, specify ENHANCED or BASIC.

For more information, see Understanding task mode differences.

- 6. Specify other task options as needed. You might be interested in some of the following options:
  - Specify what data to transfer by using a manifest or filters.
  - Configure how to handle file metadata and verify data integrity.
  - Monitor your transfer with task reports or Amazon CloudWatch Logs.

For more options, see <u>create-task</u>. Here's an example <u>create-task</u> command that specifies Enhanced mode and several other options:

```
aws datasync create-task \
    --source-location-arn "arn:aws:datasync:us-east-1:account-id:location/location-
id" \
    --destination-location-arn "arn:aws:datasync:us-east-1:account-
id:location/location-id" \
    --name "task-name" \
    --task-mode "ENHANCED" \
    --options
TransferMode=CHANGED, VerifyMode=ONLY_FILES_TRANSFERRED, ObjectTags=PRESERVE, LogLevel=TRANSF
```

7. Run the create-task command.

If the command is successful, you get a response that shows you the ARN of the task that you created. For example:

```
{
    "TaskArn": "arn:aws:datasync:us-east-1:111222333444:task/
    task-08de6e6697796f026"
}
```

#### Using the DataSync API

You can specify the DataSync task mode by configuring the TaskMode parameter in the <u>CreateTask</u> operation.

# **Choosing what AWS DataSync transfers**

AWS DataSync lets you choose what to transfer and how you want your data handled. Some options include:

- Transferring an exact list of files or object by using a manifest.
- Including or excluding certain types of data in your transfer by using a filter.
- For recurring transfers, moving only the data that's changed since the last transfer
- Overwriting data in the destination location to match what's in the source location.
- Choosing which file or object metadata to preserve between your storage locations.

### **Topics**

- Transferring specific files or objects by using a manifest
- Transferring specific files, objects, and folders by using filters
- Understanding how DataSync handles file and object metadata
- Links and directories copied by AWS DataSync
- Configuring how to handle files, objects, and metadata

# Transferring specific files or objects by using a manifest

A *manifest* is a list of files or objects that you want AWS DataSync to transfer. For example, instead of having to transfer everything in an S3 bucket with potentially millions of objects, DataSync transfers only the objects that you list in your manifest.

Manifests are similar to <u>filters</u> but let you identify exactly which files or objects to transfer instead of data that matches a filter pattern.

# **Creating your manifest**

A manifest is a comma-separated values (CSV)-formatted file that lists the files or objects in your source location that you want DataSync to transfer. If your source is an S3 bucket, you can also include which version of an object to transfer.

#### **Topics**

- Guidelines
- Example manifests

#### **Guidelines**

Use these guidelines to help you create a manifest that works with DataSync.

Do

• Specify the full path of each file or object that you want to transfer.

You can't specify only a directory or folder with the intention of transferring all of its contents. For these situations, consider using an include filter instead of a manifest.

 Make sure that each file or object path is relative to the mount path, folder, directory, or prefix that you specified when configuring your DataSync source location.

For example, let's say you <u>configure an S3 location</u> with a prefix named photos. That prefix includes an object my-picture.png that you want to transfer. In the manifest, you then only need to specify the object (my-picture.png) instead of the prefix and object (photos/my-picture.png).

 To specify Amazon S3 object version IDs, separate the object's path and version ID by using a comma.

The following example shows a manifest entry with two fields. The first field includes an object named picture1.png. The second field is separated by a comma and includes a version ID of 111111:

```
picture1.png,111111
```

- Use quotes in the following situations:
  - When a path contains special characters (commas, quotes, and line endings):

```
"filename, with, commas.txt"
```

When a path spans multiple lines:

```
"this
is
```

```
a filename.txt"
```

• When a path includes quotes:

```
filename""with""quotes.txt
```

This represents a path named filename "with" quotes.txt.

These quote rules also apply to version ID fields. In general, if a manifest field has a quote, you must escape it with another quote.

• Separate each file or object entry with a new line.

You can separate lines by using Linux (line feed or carriage return) or Windows (carriage return followed by a line feed) style line breaks.

- Save your manifest (for example, my-manifest.csv or my-manifest.txt).
- Upload the manifest to an S3 bucket that DataSync can access.

This bucket doesn't have to be in the same AWS Region or account where you're using DataSync.

#### Don't

• Specify only a directory or folder with the intention of transferring all of its contents.

A manifest can only include full paths to the files or objects that you want to transfer. If you configure your source location to use a specific mount path, folder, directory, or prefix, you don't have to include that in your manifest.

- Specify a file or object path that exceeds 4,096 characters.
- Specify a file path, object path, or Amazon S3 object version ID that exceeds 1,024 bytes.
- Specify duplicate file or object paths.
- Include an object version ID if your source location isn't an S3 bucket.
- Include more than two fields in a manifest entry.

An entry can include only a file or object path and (if applicable) an Amazon S3 object version ID.

Include characters that don't conform to UTF-8 encoding.

• Include unintentional spaces in your entry fields outside of quotes.

### **Example manifests**

Use these examples to help you create a manifest that works with DataSync.

# Manifest with full file or object paths

The following example shows a manifest with full file or object paths to transfer.

```
photos/picture1.png
photos/picture2.png
photos/picture3.png
```

## Manifest with only object keys

The following example shows a manifest with objects to transfer from an Amazon S3 source location. Since the <u>location is configured</u> with the prefix photos, only the object keys are specified.

```
picture1.png
picture2.png
picture3.png
```

## Manifest with object paths and version IDs

The first two entries in the following manifest example include specific Amazon S3 object versions to transfer.

```
photos/picture1.png,111111
photos/picture2.png,121212
photos/picture3.png
```

#### **Manifest with UTF-8 characters**

The following example shows a manifest with files that include UTF-8 characters.

```
documents/résumé1.pdf
documents/résumé2.pdf
```

documents/résumé3.pdf

#### Providing DataSync access to your manifest

You need an AWS Identity and Access Management (IAM) role that gives DataSync access to your manifest in its S3 bucket. This role must include the following permissions:

• s3:GetObject

s3:GetObjectVersion

You can generate this role automatically in the DataSync console or create the role yourself.



#### Note

If your manifest is in a different AWS account, you must create this role manually.

### Creating the IAM role automatically

When creating or starting a transfer task in the console, DataSync can create an IAM role for you with the s3:GetObject and s3:GetObjectVersion permissions that you need to access your manifest.

## Required permissions to automatically create the role

To automatically create the role, make sure that the role that you're using to access the DataSync console has the following permissions:

• iam:CreateRole

iam:CreatePolicy

iam:AttachRolePolicy

## Creating the IAM role (same account)

You can manually create the IAM role that DataSync needs to access your manifest. The following instructions assume that you're in the same AWS account where you use DataSync and your manifest's S3 bucket is located.

- 1. Open the IAM console at https://console.aws.amazon.com/iam/.
- 2. In the left navigation pane, under **Access management**, choose **Roles**, and then choose **Create role**.
- 3. On the **Select trusted entity** page, for **Trusted entity type**, choose **AWS service**.
- 4. For **Use case**, choose **DataSync** in the dropdown list and select **DataSync**. Choose **Next**.
- 5. On the **Add permissions** page, choose **Next**. Give your role a name and choose **Create role**.
- 6. On the **Roles** page, search for the role that you just created and choose its name.
- 7. On the role's details page, choose the **Permissions** tab. Choose **Add permissions** then **Create** inline policy.
- 8. Choose the **JSON** tab and paste the following sample policy into the policy editor:

**JSON** 

- 9. In the sample policy that you just pasted, replace the following values with your own:
  - a. Replace amzn-s3-demo-bucket with the name of the S3 bucket that's hosting your manifest.
  - b. Replace my-manifest.csv with the file name of your manifest.
- 10. Choose **Next**. Give your policy a name and choose **Create policy**.
- 11. (Recommended) To prevent the cross-service confused deputy problem, do the following:
  - a. On the role's details page, choose the **Trust relationships** tab. Choose **Edit trust policy**.
  - b. Update the trust policy by using the following example, which includes the aws:SourceArn and aws:SourceAccount global condition context keys:

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": "datasync.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "account-id"
            },
            "StringLike": {
                "aws:SourceArn": "arn:aws:datasync:region:account-id:*"
            }
        }
    }]
}
```

- Replace each instance account-id with the AWS account ID where you're using DataSync.
- Replace *region* with the AWS Region where you're using DataSync.
- c. Choose **Update policy**.

You've created an IAM role that allows DataSync to access your manifest. Specify this role when creating or starting your task.

#### Creating the IAM role (different account)

If your manifest is in an S3 bucket that belongs to a different AWS account, you must manually create the IAM role that DataSync uses to access the manifest. Then, in the AWS account where your manifest is located, you need to include the role in the S3 bucket policy.

#### **Creating the role**

1. Open the IAM console at https://console.aws.amazon.com/iam/.

2. In the left navigation pane, under **Access management**, choose **Roles**, and then choose **Create role**.

- 3. On the **Select trusted entity** page, for **Trusted entity type**, choose **AWS service**.
- 4. For **Use case**, choose **DataSync** in the dropdown list and select **DataSync**. Choose **Next**.
- 5. On the **Add permissions** page, choose **Next**. Give your role a name and choose **Create role**.
- 6. On the **Roles** page, search for the role that you just created and choose its name.
- 7. On the role's details page, choose the **Permissions** tab. Choose **Add permissions** then **Create inline policy**.
- 8. Choose the **JSON** tab and paste the following sample policy into the policy editor:

**JSON** 

- 9. In the sample policy that you just pasted, replace the following values with your own:
  - a. Replace amzn-s3-demo-bucket with the name of the S3 bucket that's hosting your manifest.
  - b. Replace my-manifest.csv with the file name of your manifest.
- 10. Choose **Next**. Give your policy a name and choose **Create policy**.
- 11. (Recommended) To prevent the cross-service confused deputy problem, do the following:
  - a. On the role's details page, choose the **Trust relationships** tab. Choose **Edit trust policy**.
  - b. Update the trust policy by using the following example, which includes the aws:SourceArn and aws:SourceAccount global condition context keys:

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": "datasync.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                 "aws:SourceAccount": "account-id"
            },
            "StringLike": {
                 "aws:SourceArn": "arn:aws:datasync:region:account-id:*"
            }
        }
    }]
}
```

- Replace each instance of account-id with the AWS account ID where you're using DataSync.
- Replace <u>region</u> with the AWS Region where you're using DataSync.
- c. Choose **Update policy**.

You created the IAM role that you can include in your S3 bucket policy.

### Updating your S3 bucket policy with the role

Once you've created the IAM role, you must add it to the S3 bucket policy in the other AWS account where your manifest is located.

- 1. In the AWS Management Console, switch over to the account with your manfiest's S3 bucket.
- 2. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
- 3. On the bucket's detail page, choose the **Permissions** tab.
- 4. Under **Bucket policy**, choose **Edit** and do the following to modify your S3 bucket policy:

a. Update what's in the editor to include the following policy statements:

- b. Replace account id with the AWS account ID for the account that you're using DataSync with.
- Replace datasync-role with the IAM role that you just created that allows DataSync to access your manifest.
- d. Replace amzn-s3-demo-bucket with the name of the S3 bucket that's hosting your manifest in the other AWS account.

#### Choose Save changes.

You've created an IAM role that allows DataSync to access your manifest in the other account. Specify this role when <u>creating</u> or <u>starting</u> your task.

#### Specifying your manifest when creating a task

You can specify the manifest that you want DataSync to use when creating a task.

### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Tasks**, and then choose **Create task**.

3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

4. For Contents to scan, choose Specific files, objects, and folders, then select Using a manifest.

5. For **S3 URI**, choose your manifest that's hosted on an S3 bucket.

Alternatively, you can enter the URI (for example, s3://bucket/prefix/my-manifest.csv).

6. For **Object version**, choose the version of the manifest that you want DataSync to use.

By default, DataSync uses the latest version of the object.

- 7. For Manifest access role, do one of the following:
  - Choose **Autogenerate** for DataSync to automatically create an IAM role with the permissions required to access your manifest in its S3 bucket.
  - Choose an existing IAM role that can access your manifest.

For more information, see Providing DataSync access to your manifest.

- 8. Configure any other task settings you need, then choose **Next**.
- Choose Create task.

#### Using the AWS CLI

Copy the following create-task command:

}

2. For the --source-location-arn parameter, specify the Amazon Resource Name (ARN) of the location that you're transferring data from.

- 3. For the --destination-location-arn parameter, specify the ARN of the location that you're transferring data to.
- 4. For the --manifest-config parameter, do the following:
  - ManifestObjectPath Specify the S3 object key of your manifest.
  - BucketAccessRoleArn Specify the IAM role that allows DataSync to access your manifest in its S3 bucket.

For more information, see Providing DataSync access to your manifest.

- S3BucketArn Specify the ARN of the S3 bucket that's hosting your manifest.
- ManifestObjectVersionId Specify the version of the manifest that you want DataSync to use.

By default, DataSync uses the latest version of the object.

5. Run the create-task command to create your task.

When you're ready, you can start your transfer task.

# Specifying your manifest when starting a task

You can specify the manifest that you want DataSync to use when executing a task.

# Using the DataSync console

- Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, choose Tasks, and then choose the task that you want to start.
- 3. In the task overview page, choose **Start**, and then choose **Start with overriding options**.
- 4. For Contents to scan, choose Specific files, objects, and folders, then select Using a manifest.
- 5. For S3 URI, choose your manifest that's hosted on an S3 bucket.

Alternatively, you can enter the URI (for example, s3://bucket/prefix/my-manifest.csv).

6. For **Object version**, choose the version of the manifest that you want DataSync to use.

By default, DataSync uses the latest version of the object.

- 7. For **Manifest access role**, do one of the following:
  - Choose Autogenerate for DataSync to automatically create an IAM role to access your manifest in its S3 bucket.
  - Choose an existing IAM role that can access your manifest.

For more information, see Providing DataSync access to your manifest.

8. Choose **Start** to begin your transfer.

### Using the AWS CLI

Copy the following start-task-execution command:

```
aws datasync start-task-execution \
    --task-arn arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh \
    --manifest-config {
        "Source": {
            "ManifestObjectPath": "s3-object-key-of-manifest",
            "BucketAccessRoleArn": "bucket-iam-role",
            "S3BucketArn": "amzn-s3-demo-bucket-arn",
            "ManifestObjectVersionId": "manifest-version-to-use"
        }
    }
}
```

- 2. For the --task-arn parameter, specify the Amazon Resource Name (ARN) of the task that you're starting.
- 3. For the --manifest-config parameter, do the following:
  - ManifestObjectPath Specify the S3 object key of your manifest.
  - BucketAccessRoleArn Specify the IAM role that allows DataSync to access your manifest in its S3 bucket.

For more information, see <u>Providing DataSync access to your manifest</u>.

- S3BucketArn Specify the ARN of the S3 bucket that's hosting your manifest.
- ManifestObjectVersionId Specify the version of the manifest that you want DataSync to use.

By default, DataSync uses the latest version of the object.

4. Run the start-task-execution command to begin your transfer.

#### Limitations

- You can't use a manifest together with filters.
- You can't specify only a directory or folder with the intention of transferring all of its contents. For these situations, consider using an include filter instead of a manifest.
- You can't use the Keep deleted files task option (PreserveDeletedFiles in the API) to
  maintain files or objects in the destination that aren't in the source. DataSync only transfers
  what's listed in your manifest and doesn't delete anything in the destination.

### **Troubleshooting**

If you're transferring objects with specific version IDs from an S3 bucket, you might see an error related to HeadObject or GetObjectTagging. For example, here's an error related to GetObjectTagging:

```
[WARN] Failed to read metadata for file /picture1.png (versionId: 111111): S3 Get Object Tagging Failed
[ERROR] S3 Exception: op=GetObjectTagging photos/picture1.png, code=403, type=15, exception=AccessDenied,
msg=Access Denied req-hdrs: content-type=application/xml, x-amz-api-version=2006-03-01 rsp-hdrs: content-type=application/xml,
date=Wed, 07 Feb 2024 20:16:14 GMT, server=AmazonS3, transfer-encoding=chunked, x-amz-id-2=IOWQ4fDEXAMPLEQM+ey7N9WgVhSnQ6JEXAMPLEZb7hSQDASK+Jd1vEXAMPLEa3Km, x-amz-request-id=79104EXAMPLEB723
```

If you see either of these errors, validate that the IAM role that DataSync uses to access your S3 source location has the following permissions:

• s3:GetObjectVersion

### s3:GetObjectVersionTagging

If you need to update your role with these permissions, see Creating an IAM role for DataSync to access your Amazon S3 location.

### **Next steps**

If you haven't already, start your task. Otherwise, monitor your task's activity.

# Transferring specific files, objects, and folders by using filters

AWS DataSync lets you apply filters to include or exclude data from your source location in a transfer. For example, if you don't want to transfer temporary files that end with . tmp, you can create an exclude filter so that these files don't make their way to your destination location.

You can use a combination of exclude and include filters in the same transfer task. If you modify a task's filters, those changes are applied the next time you run the task.

#### Filtering terms, definitions, and syntax

Familiarize yourself with the concepts related to DataSync filtering:

#### **Filter**

The whole string that makes up a particular filter (for example, \*.tmp|\*.temp or /folderA|/ folderB).

Filters are made up of patterns delimited by using a pipe (I). You don't need a delimiter when you add patterns in the DataSync console because you add each pattern separately.



#### (i) Note

Filters are case sensitive. For example, filter /folderA won't match /FolderA.

#### Pattern

A pattern within a filter. For example, \*.tmp is a pattern that's part of the \*.tmp | \*.temp filter. If your filter has multiple patterns, you delimit each pattern by using a pipe ().

#### **Folders**

All filters are relative to the source location path. For example, suppose that you specify /
my\_source/ as the source path when you create your source location and task and specify
the include filter /transfer\_this/. In this case, DataSync transfers only the directory /
my\_source/transfer\_this/ and its contents.

- To specify a folder directly under the source location, include a forward slash (/) in front of the folder name. In the example preceding, the pattern uses /transfer\_this, not transfer\_this.
- DataSync interprets the following patterns the same way and matches both the folder and its content.

/dir /dir/

 When you are transferring data from or to an Amazon S3 bucket, DataSync treats the / character in the object key as the equivalent of a folder on a file system.

# **Special characters**

Following are special characters for use with filtering.

Special character	Description
* (wildcard)	A character used to match zero or more characters. For example, /movies_folder* matches both /movies_f older and /movies_folder1 .
(pipe delimiter)	A character used as a delimiter between patterns. It enables specifying multiple patterns, any of which can match the filter. For example, *.tmp *.temp matches files ending with either tmp or temp.
	(3) Note  This delimiter isn't needed when you add patterns on the console because you add each pattern on a separate line.

Special character	Description
\ (backslash)	A character used for escaping special characters (*,  , $\setminus$ ) in a file or object name.
	A double backslash (\\) is required when a backslash is part of a file name. Similarly, \\\\ represents two consecutive backslashes in a file name.
	A backslash followed by a pipe (\ ) is required when a pipe is part of a file name.
	A backslash (\) followed by any other character, or at the end of a pattern, is ignored.

### **Example filters**

The following examples show common filters you can use with DataSync.



# Note

There are limits to how many characters you can use in a filter. For more information, see DataSync quotas.

# Exclude some folders from your source location

In some cases, you want might exclude folders in your source location to not copy them to your destination location. For example, if you have temporary work-in-progress folders, you can use something like the following filter:

\*/.temp

To exclude folders with similar content (such as /reports2021 and /reports2022)), you can use an exclude filter like the following:

/reports\*

To exclude folders at any level in the file hierarchy, you can use an exclude filter like the following.

```
*/folder-to-exclude-1|*/folder-to-exclude-2
```

To exclude folders at the top level of the source location, you can use an exclude filter like the following.

/top-level-folder-to-exclude-1|/top-level-folder-to-exclude-2

### Include a subset of the folders on your source location

In some cases, your source location might be a large share and you need to transfer a subset of the folders under the root. To include specific folders, start a task execution with an include filter like the following.

/folder-to-transfer/\*

#### **Exclude specific file types**

To exclude certain file types from the transfer, you can create a task execution with an exclude filter such as \*.temp.

### Transfer individual files you specify

To transfer a list of individual files, start a task execution with an include filter like the following: "/folder/subfolder/file1.txt|/folder/subfolder/file2.txt|/folder/subfolder/ file2.txt"

### **Creating include filters**

Include filters define the files, objects, and folders that you want DataSync to transfer. You can configure include filters when you create, edit, or start a task.

DataSync scans and transfers only files and folders that match the include filters. For example, to include a subset of your source folders, you might specify /important\_folder\_1|/ important\_folder\_2.



### Note

Include filters support the wildcard (\*) character only as the rightmost character in a pattern. For example, /documents\*//code\* is supported, but \*.txt isn't.

#### Using the DataSync console

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, choose **Tasks**, and then choose **Create task**.
- 3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- 4. For Contents to scan, choose Specific files, objects, and folders, then select Using filters.
- 5. For **Includes**, enter your filter (for example, /important\_folders to include an important directory), then choose **Add pattern**.
- Add other include filters as needed.

### Using the AWS CLI

When using the AWS CLI, you must use single quotation marks (') around the filter and a | (pipe) as a delimiter if you have more than one filter.

The following example specifies two include filters /important\_folder1 and / important\_folder2 when running the create-task command.

```
aws datasync create-task
    --source-location-arn 'arn:aws:datasync:region:account-id:location/location-id' \
    --destination-location-arn 'arn:aws:datasync:region:account-id:location/location-id'
    --includes FilterType=SIMPLE_PATTERN, Value='/important_folder1|/important_folder2'
```

# Creating exclude filters

Exclude filters define the files, objects, and folders in your source location that you don't want DataSync to transfer. You can configure these filters when you create, edit, or start a task.

### **Topics**

• Data excluded by default

## Data excluded by default

DataSync automatically excludes some data from being transferred:

• .snapshot – DataSync ignores any path ending with .snapshot, which typically is used for point-in-time snapshots of a storage system's files or directories.

- /.aws-datasync and /.awssync DataSync creates these folders in your location to help facilitate your transfer.
- /.zfs You might see this folder with Amazon FSx for OpenZFS locations.

### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Tasks**, and then choose **Create task**.
- 3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- 4. For **Excludes**, enter your filter (for example, \*/temp to exclude temporary folders), then choose **Add pattern**.
- 5. Add other exclude filters as needed.
- 6. If needed, add include filters.

## **Using the AWS CLI**

When using the AWS CLI, you must use single quotation marks (') around the filter and a | (pipe) as a delimiter if you have more than one filter.

The following example specifies two exclude filters \*/temp and \*/tmp when running the create-task command.

```
aws datasync create-task \
    --source-location-arn 'arn:aws:datasync:region:account-id:location/location-id' \
    --destination-location-arn 'arn:aws:datasync:region:account-id:location/location-id'
    --excludes FilterType=SIMPLE_PATTERN, Value='*/temp|*/tmp'
```

# Understanding how DataSync handles file and object metadata

AWS DataSync can preserve your file or object metadata during a data transfer. How your metadata gets copied depends on your transfer locations and if those locations use similar types of metadata.

### System-level metadata

In general, DataSync doesn't copy system-level metadata. For example, when transferring from an SMB file server, the permissions you configured at the file system level aren't copied to the destination storage system.

There are exceptions. When transferring between Amazon S3 and other object storage, DataSync does copy some system-defined object metadata.

### Metadata copied in Amazon S3 transfers

The following tables describe what metadata DataSync can copy when a transfer involves an Amazon S3 location.

### **Topics**

- To Amazon S3
- Between Amazon S3 and other object storage
- Between Amazon S3 and HDFS

#### To Amazon S3

When copying from one of these locations	To this location	DataSync can copy
<ul> <li>NFS</li> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP (using NFS)</li> </ul>	• Amazon S3	<ul> <li>The following as Amazon S3 user metadata:</li> <li>File and folder modification timestamps</li> <li>File and folder access timestamps (DataSync can only do this on a best-effort basis)</li> <li>User ID and group ID</li> <li>POSIX permissions</li> </ul>

When copying from one of these locations	To this location	DataSync can copy
		The file metadata stored in Amazon S3 user metadata is interoperable with NFS shares on file gateways using AWS Storage Gateway. A file gateway enables low-laten cy access from on-premis es networks to data that was copied to Amazon S3 by DataSync. This metadata is also interoperable with FSx for Lustre.  When DataSync copies objects that contain this metadata back to an NFS server, the file metadata is restored. Restoring metadata requires granting elevated permissions to the NFS server. For more information, see Configuring AWS DataSync transfers with an NFS file server.

# Between Amazon S3 and other object storage

When copying between these locations	DataSync can copy
Object storage	User-defined object metadata
Amazon S3	Object tags
Microsoft Azure Blob Storage	<ul> <li>The following system-defined object metadata:</li> </ul>

When copying between these locations	DataSync can copy
Amazon S3	<ul> <li>Content-Disposition</li> </ul>
	<ul> <li>Content-Encoding</li> </ul>
	<ul> <li>Content-Language</li> </ul>
	• Content-Type
	Note: DataSync copies system-level
	metadata for all objects during an initial
	transfer. If you configure your task to
	transfer only data that has changed,
	DataSync won't copy system metadata in
	subsequent transfers unless an object's
	content or user metadata has also been
	modified.
	DataSync doesn't copy other object metadata,
	such as object access control lists (ACLs), prior
	object versions, or the Last-Modified key.

# **Between Amazon S3 and HDFS**

When copying between these locations	DataSync can copy
<ul><li>Hadoop Distributed File System (HDFS)</li><li>Amazon S3</li></ul>	<ul><li>The following as Amazon S3 user metadata:</li><li>File and folder modification timestamps</li></ul>
	<ul> <li>File and folder access timestamps (DataSync can only do this on a best-effort basis)</li> </ul>
	User ID and group ID
	<ul> <li>POSIX permissions</li> </ul>

When copying between these locations	DataSync can copy
	HDFS uses strings to store file and folder user and group ownership, rather than numeric identifiers, such as UIDs and GIDs.

# **Metadata copied in NFS transfers**

The following table describes what metadata DataSync can copy between locations that use Network File System (NFS).

When copying between these locations	DataSync can copy
• NFS	File and folder modification timestamps
Amazon EFS	• File and folder access timestamps (DataSync
<ul> <li>Amazon FSx for Lustre</li> </ul>	can only do this on a best-effort basis)
<ul> <li>Amazon FSx for OpenZFS</li> </ul>	<ul> <li>User ID (UID) and group ID (GID)</li> </ul>
• Amazon FSx for NetApp ONTAP (using NFS)	POSIX permissions

# Metadata copied in SMB transfers

The following table describes what metadata DataSync can copy between locations that use Server Message Block (SMB).

When copying between these locations	DataSync can copy
<ul> <li>SMB</li> <li>Amazon FSx for Windows File Server</li> <li>FSx for ONTAP (using SMB)</li> </ul>	<ul> <li>File timestamps: access time, modification time, and creation time</li> <li>File owner security identifier (SID)</li> <li>Standard file attributes: read-only (R), archive (A), system (S), hidden (H), compressed (C), not content indexed (I), encrypted (E), temporary (T), offline (O), and sparse (P)</li> </ul>

When copying between these locations	DataSync can copy
	<ul> <li>DataSync attempts to copy the archive (A), compressed (C), not context indexed (I), sparse (P), and temporary (T) attributes on a best-effort basis. If these attributes aren't applied on the destination, they're ignored during task verification.</li> <li>NTFS discretionary access lists (DACLs), which determine whether to grant access to an object.</li> <li>NTFS system access control lists (SACLs), which are used by administrators to log attempts to access a secured object.</li> <li>Note: SACLs are not copied if you use SMB version 1.0.</li> <li>Copying DACLs and SACLs requires granting specific permissions to the Windows user that DataSync uses to access your location using SMB. For more informati on, see creating a location for SMB, FSx for Windows File Server, or FSx for ONTAP (depending on the type of location in your transfer).</li> </ul>

# Metadata copied in other transfer scenarios

DataSync handles metadata the following ways when copying between these storage systems (most of which have different metadata structures).

When copying from one of these locations	To one of these locations	DataSync can copy
<ul> <li>SMB</li> <li>FSx for Windows File Server</li> <li>FSx for ONTAP (using SMB)</li> </ul>	<ul> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP (using NFS)</li> <li>Amazon S3</li> <li>Object storage</li> <li>Azure Blob Storage</li> <li>NFS</li> </ul>	Default POSIX metadata for all files and folders on the destination file system or objects in the destination S3 bucket. This approach includes using the default POSIX user ID and group ID values.  Windows-based metadata (such as ACLs) is not preserved.
<ul><li>Object storage</li><li>Amazon S3</li><li>Azure Blob Storage</li></ul>	<ul><li>Amazon EFS</li><li>FSx for Lustre</li><li>FSx for OpenZFS</li><li>FSx for ONTAP (using NFS)</li></ul>	Default POSIX metadata on the destination files and folders. This approach includes using the default POSIX user ID and group ID values.
<ul> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP (using NFS)</li> </ul>	Azure Blob Storage	<ul> <li>The following as user-defined metadata:</li> <li>File and folder modification timestamps</li> <li>File and folder access timestamps (DataSync can only do this on a best-effort basis)</li> <li>User ID and group ID</li> <li>POSIX permissions</li> </ul>
• HDFS	<ul><li>Amazon EFS</li><li>FSx for Lustre</li></ul>	• File and folder modification timestamps

When copying from one of these locations	To one of these locations	DataSync can copy
	FSx for OpenZFS     FSx for ONTAP (using NFS)	<ul> <li>File and folder access timestamps (DataSync can only do this on a best-effort basis)</li> <li>POSIX permissions</li> <li>HDFS stores file and folder user and group ownership as strings rather than numeric identifiers (such as UIDs and GIDs). Default values for UIDs and GIDs are applied on the destination file system. For more information, see Understanding when and how DataSync applies default POSIX metadata.</li> </ul>
<ul> <li>Amazon S3</li> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for Windows File Server</li> <li>FSx for ONTAP</li> </ul>	• HDFS	File and folder timestamp s from the source location. The file or folder owner is set based on the HDFS user or Kerberos principal you specified when creating the HDFS transfer location. The Groups Mapping configuration on the Hadoop cluster determines the group.

When copying from one of these locations	To one of these locations	DataSync can copy
<ul> <li>Amazon S3</li> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP (using NFS)</li> <li>Object storage</li> <li>NFS</li> <li>HDFS</li> </ul>	<ul> <li>SMB</li> <li>FSx for Windows File Server</li> <li>FSx for ONTAP (using SMB)</li> </ul>	File and folder timestamp s from the source location. Ownership is set based on the Windows user that was specified in DataSync to access the Amazon FSx or SMB share. Permissions are inherited from the parent
Azure Blob Storage	<ul><li>FSx for Windows File Server</li><li>FSx for ONTAP (using SMB)</li></ul>	directory.

# Understanding when and how DataSync applies default POSIX metadata

DataSync applies default POSIX metadata in the following situations:

- When your transfer's source and destination locations don't have similar metadata structures
- When metadata is missing from the source location

The following table describes how DataSync applies default POSIX metadata during these types of transfers:

Source	Destination	File permissions	Folder permissions	UID	GID
<ul> <li>Amazon</li> <li>S3<sup>1</sup></li> <li>Object</li> <li>storage<sup>1</sup></li> </ul>	<ul><li>Amazon EFS</li><li>FSx for Lustre</li><li>FSx for OpenZFS</li></ul>	0755	0755	65534	65534

Source	Destination	File permissions	Folder permissions	UID	GID
<ul> <li>Microsoft         Azure Blob         Storage<sup>1</sup> </li> </ul>	<ul><li>FSx for ONTAP (using NFS)</li><li>NFS</li></ul>				
• SMB	<ul> <li>Amazon S3</li> <li>Object storage</li> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP (using NFS)</li> <li>NFS</li> </ul>	0644	0755	65534	65534
• HDFS	<ul> <li>Amazon EFS</li> <li>FSx for Lustre</li> <li>FSx for OpenZFS</li> <li>FSx for ONTAP (using NFS)</li> <li>NFS</li> </ul>	0644	0755	65534	65534

<sup>&</sup>lt;sup>1</sup> In cases where the objects don't have metadata that was previously applied by DataSync.

# Links and directories copied by AWS DataSync

AWS DataSync handles hard links, symbolic links, and directories differently depending on the storage locations involved in your transfer.

#### Hard links

Here's how DataSync handles hard links in some common transfer scenarios:

- When transferring between an NFS file server, FSx for Lustre, FSx for OpenZFS, FSx for ONTAP (using NFS), and Amazon EFS, hard links are preserved.
- When transferring to Amazon S3, each underlying file referenced by a hard link is transferred only once. During incremental transfers, separate objects are created in your S3 bucket. If a hard link is unchanged in Amazon S3, it's correctly restored when transferred to an NFS file server, FSx for Lustre, FSx for OpenZFS, FSx for ONTAP (using NFS), or Amazon EFS file system.
- When transferring to Microsoft Azure Blob Storage, each underlying file referenced by a
  hard link is transferred only once. During incremental transfers, separate objects are created in
  your blob storage if there are new references in the source. When transferring from Azure Blob
  Storage, DataSync transfers hard links as if they are individual files.
- When transferring between an SMB file server, FSx for Windows File Server, and FSx for ONTAP (using SMB), hard links aren't supported. If DataSync encounters hard links in these situations, the transfer task completes with an error. To learn more, check your CloudWatch logs.
- When transferring to HDFS, hard links aren't supported. CloudWatch logs show these links as skipped.

### Symbolic links

Here's how DataSync handles symbolic links in some common transfer scenarios:

- When transferring between an NFS file server, FSx for Lustre, FSx for OpenZFS, FSx for ONTAP (using NFS), and Amazon EFS, symbolic links are preserved.
- When transferring to Amazon S3, the link target path is stored in the Amazon S3 object. The
  link is correctly restored when transferred to an NFS file server, FSx for Lustre, FSx for OpenZFS,
  FSx for ONTAP, or Amazon EFS file system.
- When transferring to Azure Blob Storage, symbolic links aren't supported. CloudWatch logs show these links as skipped.

When transferring between an SMB file server, FSx for Windows File Server, and FSx for
ONTAP (using SMB), symbolic links aren't supported. DataSync doesn't transfer a symbolic
link itself but instead a file referenced by the symbolic link. To recognize duplicate files and
deduplicate them with symbolic links, you must configure deduplication on your destination file
system.

• When transferring to HDFS, symbolic links aren't supported. CloudWatch logs show these links as skipped.

#### **Directories**

In general, DataSync preserves directories when transferring between storage systems. This isn't the case in the following situations:

- When transferring to Amazon S3, directories are represented as empty objects that have prefixes and end with a forward slash (/).
- When transferring to Azure Blob Storage without a hierarchical namespace, directories don't exist. What looks like a directory is just part of an object name.

# Configuring how to handle files, objects, and metadata

You can configure how AWS DataSync handles your files, objects, and their associated metadata when transferring between locations.

For example, with recurring transfers, you might want to overwrite files in your destination with changes in the source to keep the locations in sync. You can copy properties such as POSIX permissions for files and folders, tags associated with objects, and access control lists (ACLs).

### **Transfer mode options**

You can configure whether DataSync transfers only the data (including metadata) that's changed following an initial copy or all data every time you run the task. If you're planning on recurring transfers, you might only want to transfer what's changed since your previous task execution.

Option in console	Option in API	Description
Transfer only data that has	<u>TransferMode</u> set to CHANGED	After your initial full transfer,
changed		DataSync copies only the data

Option in console	Option in API	Description
		and metadata that differs between the source and destination location.
Transfer all data	<u>TransferMode</u> set to ALL	DataSync copies everything in the source to the destination without comparing differenc es between the locations.

# File and object handling options

You can control some aspects of how DataSync treats your files or objects in the destination location. For example, DataSync can delete files in the destination that aren't in the source.

Option in console	Option in API	Description
Keep deleted files	PreserveDeletedFiles	Specifies whether DataSync maintains files or objects in the destination location that don't exist in the source.  If you configure your task to delete objects from your Amazon S3 bucket, you might incur minimum storage duration charges for certain storage classes. For detailed information, see <a href="Storage class considerations with Amazon S3 transfers">Storage class considerations with Amazon S3 transfers</a> .  Marning  You can't configure your task to delete data in the destinati

Option in console	Option in API	Description
		on and also <u>transfer</u> <u>all data</u> . When you transfer all data, DataSync doesn't scan your destination location and doesn't know what to delete.
Overwrite files	OverwriteMode	Specifies whether DataSync modifies data in the destinati on location when the source data or metadata has changed. If you don't configure your task to overwrite data, the destinati on data isn't overwritten even if the source data differs.
		objects, you might incur additional charges for certain storage classes (for example, for retrieval or early deletion) . For detailed information, see Storage class considerations with Amazon S3 transfers.

# Metadata handling options

DataSync can preserve file and object metadata during a transfer. The metadata that DataSync can preserve depends on the storage systems involved and whether those systems use a similar metadata structure.

Before configuring your task, make sure that you understand how DataSync handles <u>metadata</u> and <u>special files</u> when transferring between your source and destination locations.

#### **∧** Important

DataSync supports transfers to and from certain third-party cloud storage systems, such as Google Cloud Storage and IBM Cloud Object Storage, which handle system metadata in a way that is not fully S3-compatible. For these transfers, DataSync attempts to copy metadata attributes such as ContentType, ContentEncoding, ContentLanguage, and CacheControl on a best-effort basis. If the destination storage system does not apply these attributes, they will be ignored during task verification.

Option in console	Option in API	Description
Copy ownership	Gid and <u>Uid</u>	Specifies whether DataSync copies POSIX file and folder ownership, such as the group ID of the file's owners and the user ID of the file's owner.
Copy permissions	PosixPermissions	Specifies whether DataSync copies POSIX permissions for files and folders from the source to the destination.
Copy timestamps	Atime and Mtime	Specifies whether DataSync copies the timestamp metadata from the source to the destination.
Copy object tags	<u>ObjectTags</u>	Specifies whether DataSync preserves the tags associate d with your objects when transferring between object storage systems.
Copy ownership, DACLs, and SACLs	SecurityDescriptorCopyFlags set to OWNER_DACL_SACL	DataSync copies the following:
		<ul> <li>The object owner.</li> </ul>

Option in console	Option in API	Description
		<ul> <li>NTFS discretionary access lists (DACLs), which determine whether to grant access to an object.</li> <li>NTFS system access control lists (SACLs), which are used by administrators to log attempts to access a secured object.</li> <li>Note: SACLs are not copied if you use SMB version 1.0.</li> <li>Copying DACLs and SACLs requires granting specific permissions to the Windows user that DataSync uses to access your location using SMB. For more information, see creating a location for SMB, FSx for Windows File Server, or FSx for ONTAP (depending on the type of location in your transfer).</li> </ul>

Option in console	Option in API	Description
Copy ownership and DACLs	SecurityDescriptorCopyFlags set to OWNER_DACL	<ul> <li>DataSync copies the following:</li> <li>The object owner.</li> <li>DACLs, which determine whether to grant access to an object.</li> <li>DataSync won't copy SACLs when you choose this option.</li> </ul>
Do not copy ownership or ACLs	SecurityDescriptorCopyFlags set to NONE	DataSync doesn't copy any ownership or permissio ns data. The objects that DataSync writes to your destination location are owned by the user whose credentials are provided for DataSync to access the destination. Destinati on object permissions are determined based on the permissions configured on the destination server.

### Configuring file, object, and metadata handling options

You can configure how DataSync handles files, objects, and metadata when creating, editing, or starting your transfer task.

### Using the DataSync console

The following instructions describe how to configure file, object, and metadata handling options when creating a task.

1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.

In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** 2. task.

3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- 4. For **Transfer mode**, choose one of the following options:
  - · Transfer only data that has changed
  - Transfer all data

For more information about these options, see Transfer mode options.

Select Keep deleted files if you want DataSync to maintain files or objects in the destination 5. location that don't exist in the source.

If you don't choose this option and your task deletes objects from your Amazon S3 bucket, you might incur minimum storage duration charges for certain storage classes. For detailed information, see Storage class considerations with Amazon S3 transfers.

#### Marning

You can't deselect this option and enable Transfer all data. When you transfer all data, DataSync doesn't scan your destination location and doesn't know what to delete.

Select **Overwrite files** if you want DataSync to modify data in the destination location when 6. the source data or metadata has changed.

If your task overwrites objects, you might incur additional charges for certain storage classes (for example, for retrieval or early deletion). For detailed information, see Storage class considerations with Amazon S3 transfers.

If you don't choose this option, the destination data isn't overwritten even if the source data differs.

Under Transfer options, select how you want DataSync to handle metadata. For more information about the options, see Metadata handling options.

#### Important

The options you see in the console depend on your task's source and destination locations. You might have to expand Additional settings to see some of these options.

- Copy ownership
- Copy permissions
- Copy timestamps
- Copy object tags
- Copy ownership, DACLs, and SACLs
- Copy ownership and DACLs
- Do not copy ownership or ACLs

### **Using the DataSync API**

You can configure file, object, and metadata handling options by using the Options parameter with any of the following operations:

- CreateTask
- StartTaskExecution
- UpdateTask

# **Configuring how AWS DataSync verifies data integrity**

During a transfer, AWS DataSync uses checksum verification to verify the integrity of the data that you copy between locations. You also can configure DataSync to perform additional verification at the end of your transfer.

# **Data verification options**

Use the following information to help you decide if and how you want DataSync to perform these additional checks.

Verifying data integrity 243

Console option	API option	Description
Verify only transferred data (recommended)  VerifyMode set to ONLY_FILE S_TRANSFERRED		DataSync calculates the checksum of transferred data (including metadata) at the source location. At the end of your transfer, DataSync compares this checksum to the checksum calculate d on that same data at the destination.
		We recommend this option when transferring to S3 Glacier Flexible Retrieval or S3 Glacier Deep Archive storage classes. For more information, see Storage class considerations with Amazon S3 transfers.
Verify all data	<pre>VerifyMode set to POINT_IN_ TIME_CONSISTENT</pre>	At the end of your transfer, DataSync checks the entire source and destination to verify that both locations are fully synchronized.
		Note Not supported when your task uses Enhanced mode.
		If you use a <u>manifest</u> ,  DataSync only scans and

Verifying data integrity 244

Console option	API option	Description
		verifies what's listed in the manifest.  You can't use this option when transferring to S3 Glacier Flexible Retrieval or S3 Glacier Deep Archive storage classes. For more information, see Storage class considerations with Amazon S3 transfers.
Don't verify data after transfer	<u>VerifyMode</u> set to NONE	DataSync performs data integrity checks only during your transfer. Unlike other options, there's no additional verification at the end of your transfer.

## **Configuring data verification**

You can configure data verification options when creating a task, updating a task, or starting a task execution.

### Using the DataSync console

The following instructions describe how to configure data verification options when creating a task.

# To configure data verification by using the console

- Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create task**.
- 3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

Verifying data integrity 245

- For **Verification**, choose one of the following:
  - Verify only transferred data (recommended)
  - Verify all data
  - Don't verify data after transfer

#### Using the DataSync API

You can configure how DataSync verifies data by using the VerifyMode parameter with any of the following operations:

- CreateTask
- UpdateTask
- StartTaskExecution

# Setting bandwidth limits for your AWS DataSync task

You can configure network bandwidth limits for your AWS DataSync task and each of its executions.



#### Note

Not applicable to Enhanced mode tasks.

# Limiting bandwidth for a task

Set a bandwidth limit when creating, editing, or starting a task.

### Using the DataSync console

The following instructions describe how to configure a bandwidth limit for your task when you're creating it.

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.

Setting bandwidth limits 246

3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- 4. For **Bandwidth limit**, choose one of the following:
  - Select **Use available** to use all of the available network bandwidth for each task execution.
  - Select **Set bandwidth limit (MiB/s)** and enter the maximum bandwidth that you want DataSync to use for each task execution.

#### Using the DataSync API

You can configure a task's bandwidth limit by using the BytesPerSecond parameter with any of the following operations:

- CreateTask
- UpdateTask
- StartTaskExecution

### Throttling bandwidth for a task execution

You can modify the bandwidth limit for a running or queued task execution.

### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the navigation pane, expand **Data transfer**. then choose **Tasks**.
- 3. Choose the task and then select **History** to view the task's executions.
- 4. Choose the task execution that you want to modify and then choose **Edit**.
- 5. In the dialog box, choose one of the following:
  - Select Use available to use all of the available network bandwidth for the task execution.
  - Select **Set bandwidth limit (MiB/s)** and enter the maximum bandwidth that you want DataSync to use for the task execution.
- Choose Save changes.

The new bandwidth limit takes effect within 60 seconds.

Setting bandwidth limits 247

#### Using the DataSync API

You can modify the bandwidth limit for a running or queued task execution by using the BytesPerSecond parameter with the UpdateTaskExecution operation.

# Scheduling when your AWS DataSync task runs

You can set up an AWS DataSync task schedule to periodically transfer data between storage locations.

### How DataSync task scheduling works

A scheduled DataSync task runs at a frequency that you specify, with a minimum interval of 1 hour. You can create a task schedule by using a cron or rate expressions.



#### Important

You can't schedule a task to run at an interval faster than 1 hour.

### Using cron expressions

Use cron expressions for task schedules that run on a specific time and day. For example, here's how you can configure a task schedule in the AWS CLI that runs at 12:00 PM UTC every Sunday and Wednesday.

```
cron(0 12 ? * SUN,WED *)
```

#### Using rate expressions

Use rate expressions for task schedules that run on a regular interval, such as every 12 hours. For example, here's how you can configure a task schedule in the AWS CLI that runs every 12 hours:

```
rate(12 hours)
```



#### (i) Tip

For more information about cron and rate expression syntax, see the *Amazon EventBridge* User Guide.

### Creating a DataSync task schedule

You can schedule how frequently your task runs by using the DataSync console, AWS CLI, or DataSync API.

#### Using the DataSync console

The following instructions describe how to set up a schedule when creating a task. You can modify the schedule later when editing the task.

In the console, some scheduling options let you specify the exact time that your task runs (such as daily at 10:30 PM). If you don't include a time for these options, your task runs at the time that you create (or update) the task.

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.
- Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- For schedule **Frequency**, do one of the following:
  - Choose Not scheduled if you don't want your task to run on a schedule.
  - Choose **Hourly**, then choose the minute during the hour that you want your task to run.
  - Choose Daily and enter the UTC time that you want your task to run.
  - Choose Weekly and the day of the week and enter the UTC time that you want the task to run.
  - Choose Days of the week, choose a specific day or days, and enter the UTC time that the task should run in the format HH:MM.
  - Choose Custom, and then select Cron expression or Rate expression. Enter your task schedule with a minimum interval of 1 hour.

#### Using the AWS CLI

You can create a schedule for your DataSync task by using the --schedule parameter with the create-task, update-task, or start-task-execution command.

The following instructions describe how to do this with the create-task command.

Copy the following create-task command:

```
aws datasync create-task \
    --source-location-arn arn:aws:datasync:us-east-1:123456789012:location/
loc-12345678abcdefgh \
    --destination-location-arn arn:aws:datasync:us-east-1:123456789012:location/
loc-abcdefgh12345678 \
    --schedule '{
        "ScheduleExpression": "cron(0 12 ? * SUN,WED *)"
    }'
```

- 2. For the --source-location-arn parameter, specify the Amazon Resource Name (ARN) of the location that you're transferring data from.
- 3. For the --destination-location-arn parameter, specify the ARN of the location that you're transferring data to.
- 4. For the --schedule parameter, specify a cron or rate expression for your schedule.

In the example, the cron expression  $cron(0\ 12\ ?\ *\ SUN, WED\ *)$  sets a task schedule that runs at 12:00 PM UTC every Sunday and Wednesday.

5. Run the create-task command to create your task with the schedule.

# Pausing a DataSync task schedule

There can be situations where you need to pause your DataSync task schedule. For example, you might need to temporarily disable a recurring transfer to fix an issue with your task or perform maintenance on your storage system.

DataSync might disable your task schedule automatically for the following reasons:

- Your task fails repeatedly with the same error.
- You disable an AWS Region that your task is using.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, and then choose **Tasks**.
- 3. Choose the task that you want to pause the schedule for, and then choose **Edit**.
- 4. For Schedule, turn off Enable schedule. Choose Save changes.

#### **Using the AWS CLI**

1. Copy the following update-task command:

```
aws datasync update-task \
    --task-arn arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh \
    --schedule '{
        "ScheduleExpression": "cron(0 12 ? * SUN, WED *)",
        "Status": "DISABLED"
    }'
```

- 2. For the --task-arn parameter, specify the ARN of the task that you want to pause the schedule for.
- 3. For the --schedule parameter, do the following:
  - For ScheduleExpression, specify a cron or rate expression for your schedule.

In the example, the expression  $cron(0\ 12\ ?\ *\ SUN, WED\ *)$  sets a task schedule that runs at 12:00 PM UTC every Sunday and Wednesday.

- For Status, specify DISABLED to pause the task schedule.
- 4. Run the update-task command.
- 5. To resume the schedule, run the same update-task command with Status set to ENABLED.

# Checking the status of a DataSync task schedule

You can see whether your DataSync task schedule is enabled.

### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand Data transfer, and then choose Tasks.

3. In the **Schedule** column, check whether the task's schedule is enabled or disabled.

#### **Using the AWS CLI**

1. Copy the following describe-task command:

```
aws datasync describe-task \
--task-arn arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh
```

- 2. For the --task-arn parameter, specify the ARN of the task that you want information about.
- 3. Run the describe-task command.

You get a response that provides details about your task, including its schedule. (The following example focuses primarily on the task schedule configuration and doesn't show a full describetask response.)

The example shows that the task's schedule is manually disabled. If the schedule is disabled by the DataSync SERVICE, you see an error message for DisabledReason to help you understand why the task keeps failing. For more information, see ???.

```
{
   "TaskArn": "arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh",
   "Status": "AVAILABLE",
   "Schedule": {
        "ScheduleExpression": "cron(0 12 ? * SUN,WED *)",
        "Status": "DISABLED",
        "StatusUpdateTime": 1697736000,
        "DisabledBy": "USER",
        "DisabledReason": "Manually disabled by user."
},
   ...
}
```

# Tagging your AWS DataSync tasks

*Tags* are key-value pairs that help you manage, filter, and search for your AWS DataSync resources. You can add up to 50 tags to each DataSync task and task execution.

For example, you might create a task for a large data migration and tag the task with the key **Project** and value **Large Migration**. To further organize the migration, you could tag one run

Tagging your tasks 252

of the task with the key **Transfer Date** and value **May 2021** (subsequent task executions might be tagged **June 2021**, **July 2021**, and so on).

### **Tagging your DataSync task**

You can tag your DataSync task only when creating the task.

#### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.
- 3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

4. On the **Configure settings** page, choose **Add new tag** to tag your task.

#### Using the AWS CLI

Copy the following create-task command:

```
aws datasync create-task \
     --source-location-arn 'arn:aws:datasync:region:account-id:location/source-
location-id' \
     --destination-location-arn 'arn:aws:datasync:region:account-
id:location/destination-location-id' \
     --tags Key=tag-key,Value=tag-value
```

- 2. Specify the following parameters in the command:
  - --source-location-arn Specify the Amazon Resource Name (ARN) of the source location in your transfer.
  - --destination-location-arn Specify the ARN of the destination location in your transfer.
  - --tags Specify the tags that you want to apply to the task.

For more than one tag, separate each key-value pair with a space.

3. (Optional) Specify other parameters that make sense for your transfer scenario.

Tagging your tasks 253

For a list of --options, see the create-task command.

Run the create-task command.

You get a response that shows the task that you just created.

```
{
    "TaskArn": "arn:aws:datasync:us-east-2:123456789012:task/task-abcdef01234567890"
}
```

To view the tags you added to this task, you can use the list-tags-for-resource command.

### Tagging your DataSync task execution

You can tag each run of your DataSync task.

If your task already has tags, remember the following about using tags with task executions:

- If you start your task with the console, its user-created tags are applied automatically to the task execution. However, system-created tags that begin with aws: are not applied.
- If you start your task with the DataSync API or AWS CLI, its tags are not applied automatically to the task execution.

#### Using the DataSync console

To add, edit, or remove tags from a task execution, you must start the task with overriding options.

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**.
- 3. Choose the task.
- 4. Choose **Start**, then choose one of the following options:
  - **Start with defaults** Applies any tags associated with your task.
  - Start with overriding options Allows you to add, edit, or remove tags for this particular task execution.

Tagging your tasks 254

#### Using the AWS CLI

Copy the following start-task-execution command:

```
aws datasync start-task-execution \
    --task-arn 'arn:aws:datasync:region:account-id:task/task-id' \
    --tags Key=tag-key, Value=tag-value
```

- 2. Specify the following parameters in the command:
  - --task-arn Specify the ARN of the task that you want to start.
  - --tags Specify the tags that you want to apply to this specific run of the task.

For more than one tag, separate each key-value pair with a space.

3. (Optional) Specify other parameters that make sense for your situation.

For more information, see the start-task-execution command.

Run the start-task-execution command.

You get a response that shows the task execution that you just started.

```
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-2:123456789012:task/task-abcdef01234567890"
}
```

To view the tags you added to this task, you can use the <u>list-tags-for-resource</u> command.

# Starting a task to transfer your data

Once you create your AWS DataSync transfer task, you can start moving data. Each run of a task is called a *task execution*. For information about what happens during a task execution, see <u>How DataSync transfers files</u>, objects, and directories.

### Important

If you're planning to transfer data to or from an Amazon S3 location, review <u>how DataSync</u> can affect your S3 request charges and the DataSync pricing page before you begin.

# Starting your task

Once you've created your task, you can begin moving data right away.

#### Using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**.
- 3. Choose the task that you want to run.

Make sure that the task has an Available status. You also can select multiple tasks.

- 4. Choose **Actions** and then choose one of the following options:
  - **Start** Runs the task (or tasks if you selected more than one).
  - **Start with overriding options** Allows you to modify some of your task settings before you begin moving data. When you're ready, choose **Start**.
- 5. Choose **See execution details** to view details about the running task execution.

#### Using the AWS CLI

To start your DataSync task, you just need to specify the Amazon Resource Name (ARN) of the task you want to run. Here's an example start-task-execution command:

```
aws datasync start-task-execution \
    --task-arn 'arn:aws:datasync:region:account-id:task/task-id'
```

The following example starts a task with a few settings that are different than the task's default settings:

```
aws datasync start-task-execution \
    --override-options VerifyMode=NONE,OverwriteMode=NEVER,PosixPermissions=NONE
```

The command returns an ARN for your task execution similar to the following example:

```
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-1:209870788375:task/
task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f"
}
```

Starting your task 256

User Guide AWS DataSync



### Note

Each agent can run a single task at a time.

#### Using the DataSync API

You can start your task by using the StartTaskExecution operation. Use the DescribeTaskExecution operation to get details about the running task execution.

Once started, you can check the task execution's status as DataSync copies your data. You also can throttle the task execution's bandwidth if needed.

### Task execution statuses

When you start a DataSync task, you might see these statuses. (Task statuses are different than task execution statuses.)

Console status	API status	Description
Queueing	QUEUED	Another task execution is running and using the same DataSync agent. For more information, see <a href="Knowing when your task is queued">Knowing when your task is queued</a> .
Launching	LAUNCHING	DataSync is initializing the task execution. This status usually goes quickly but can take up to a few minutes.
Preparing	PREPARING	DataSync is determining what data to transfer.  Preparation can take just minutes, a few hours, or even longer depending on the number of files, objects, or directories in both locations and how you configure your task. How preparation works also depends on your task mode. For more information, see <a href="How DataSync prepares">How DataSync prepares</a> your data transfer.
Transferring	TRANSFERR ING	DataSync is performing the actual data transfer.

Task execution statuses 257

Console status	API status	Description
Verifying	VERIFYING	DataSync is verifying the integrity of your data at the end of the transfer.
Success	SUCCESS	The task execution succeeded.
Cancelling	CANCELLING	The task execution is in the process of being cancelled.
Error	ERROR	The task execution failed.

# Knowing when your task is queued

When running multiple tasks (for example, you're <u>transferring a large dataset</u>), DataSync might queue the tasks to run in a series (first in, first out). Some examples of when this happens include:

- You run different tasks that use the same DataSync agent. While you can use the same agent for multiple tasks, an agent can only run one task at a time.
- A task execution is in progress and you start additional executions of the same task using different filters or manifests.

In each example, the queued tasks don't start until the task ahead of them finishes.

# Cancelling your task execution

You can stop any running or queued DataSync task execution.

#### To cancel a task execution by using the console

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**.
- 3. Select the **Task ID** for the running task that you want to monitor.

The task status should be **Running**.

- 4. Choose **History** to view the task's executions.
- 5. Select the task execution that you want to stop, and then choose **Stop**.
- 6. In the dialog box, choose **Stop**.

To cancel a running or queued task by using the DataSync API, see <a href="CancelTaskExecution">CancelTaskExecution</a>.

# Monitoring your AWS DataSync transfers

Monitoring is important for maintaining the reliability and performance of your AWS DataSync transfer activities. We recommend that you collect monitoring data so that you can more easily debug errors if one occurs. Before you start monitoring DataSync, however, create a monitoring plan that includes answers to the following questions:

- What are your monitoring goals?
- · What resources will you monitor?
- How often will you monitor these resources?
- What monitoring tools will you use?
- Who will perform the monitoring tasks?
- Who should be notified when something goes wrong?

AWS provides various services and tools for monitoring DataSync. You can configure some of these to do the monitoring for you, but some require manual intervention. We recommend that you automate monitoring tasks as much as possible.

#### **Topics**

- Understanding data transfer performance counters
- Monitoring data transfers with Amazon CloudWatch metrics
- Monitoring your data transfers with task reports
- Monitoring data transfers with Amazon CloudWatch Logs
- Logging AWS DataSync API calls with AWS CloudTrail
- Monitoring events by using Amazon EventBridge
- Monitoring AWS DataSync with manual tools

# Understanding data transfer performance counters

When you <u>start a task</u>, AWS DataSync provides counters to help track your data transfer's performance and progress.

Use the following information to understand what each counter represents. You can view these counters in the DataSync console or a <a href="DescribeTaskExecution">DescribeTaskExecution</a> response. Some counters aren't available with every <a href="task mode">task mode</a>.

Console	DescribeT askExecution	Task mode support	Description
-	BytesWritten	Enhanced, Basic	The number of logical bytes that DataSync actually writes to the destination location.
Data throughput		Enhanced, Basic	The rate at which DataSync writes logical bytes to the destination location.  If you're using DescribeTaskExecut ion, how you calculate this counter depends on your task mode:  • Enhanced mode: Divide BytesWritten by TotalDura tion  • Basic mode: Divide BytesWritten by TransferD uration
Data transferred	BytesTran sferred	Enhanced, Basic	The number of bytes that DataSync sends to the network before compression

Console	DescribeT askExecution	Task mode support	Description
			(if compression is possible).  For the number of bytes transferred over the network, see the Network throughput (in console) or BytesCompressed (in DescribeT askExecution) counter.
Deleted from destination	FilesDeleted	Enhanced, Basic	The number of files, objects, and directori es that DataSync actually deletes in your destination location.  If you don't configure your task to delete data in the destination that isn't in the source:  • Deleted from destination doesn't display in the console.  • FilesDeleted always shows a value of 0.

Console	DescribeT askExecution	Task mode support	Description
_	Estimated BytesToTr ansfer	Enhanced, Basic	The number of logical bytes that DataSync expects to write to the destination location.
	Estimated FilesToDelete	Enhanced, Basic	The number of files, objects, and directori es that DataSync expects to delete in your destination location.  If you don't configure your task to delete data in the destination that isn't in the source, the value is always 0.

Console	DescribeT askExecution	Task mode support	Description
	Estimated FilesToTr ansfer	Enhanced, Basic	The number of files, objects, and directori es that DataSync expects to transfer over the network. This value is calculate d while DataSync prepares the transfer.  How this gets calculated depends primarily on the transfer mode you're using:  • If transfer mode is set to transfer only data that has changed: The calculation is based on comparing the content of the source and destination locations and determining the difference that needs to be transferred. The difference can include:  • Anything that's added or

Console	DescribeT askExecution	Task mode support	Description
			modified at the source location.  Anything that's in both locations and modified at the destination after an initial transfer (unless you configure your task to not overwrite data in the destination).  (Basic mode only) The number of items that DataSync expects to delete (if you configure your task to delete data in the destination).  If transfer mode is set to transfer all data: The calculati on is based only on the items that DataSync finds at the source location.

Console	DescribeT askExecution	Task mode support	Description
File throughput		Enhanced, Basic	The rate at which DataSync transfers files, objects, and directories over the network.  If you're using DescribeTaskExecut ion, how you calculate this counter depends on your task mode:  • Enhanced mode: Divide FilesTran sferred by TotalDuration • Basic mode: Divide FilesTran sferred by TransferD uration

Console	DescribeT askExecution	Task mode support	Description
	FilesFailed	Enhanced	The number of objects that DataSync fails to prepare, transfer, verify, and delete during your task execution.  If there are failures, you can see these alongside the Prepared, Transferr ed, Skipped, and Deleted from destination console counters, respectively.

Console	DescribeT askExecution	Task mode support	Description
Listed at source	FilesList ed.AtSource	Enhanced	<ul> <li>The number of objects that DataSync finds at your source location.</li> <li>With a manifest, DataSync lists only what's in your manifest (and not everything in your source location).</li> <li>With an include filter, DataSync lists only what matches the filter at your source location.</li> <li>With an exclude filter, DataSync lists everything at your source location before applying the filter.</li> </ul>

Console	DescribeT askExecution	Task mode support	Description
	FilesList ed.AtDest inationFo rDelete	Enhanced	The number of objects that DataSync finds at your destinati on location.  This counter is only applicable if you configure your task to delete data in the destination that isn't in the source.

Console	DescribeT askExecution	Task mode support	Description
Network throughpu t*	BytesCompressed	Enhanced, Basic	The number of physical bytes that DataSync transfers over the network after compression (if compression is possible).  This number is typically less than Data transferred (in console) or BytesTran sferred (in DescribeTaskExecut ion) unless the data isn't compressible.  * – For Enhanced mode, Network throughput doesn't display in the console.

Console	DescribeT askExecution	Task mode support	Description
Percent compressed		Basic	The percentage of transfer data that DataSync compressed before sending it over the network.  If you're using DescribeTaskExecut ion, you can calculate this counter with 1 - BytesComp ressed / BytesWritten .

Console	DescribeT askExecution	Task mode support	Description
Prepared	FilesPrepared	Enhanced	The number of objects that DataSync will attempt to transfer after comparing your source and destinati on locations.  In the console, this counter can also show you the number of objects that DataSync skips during preparati on. For more information, see How DataSync prepares your data transfer.  This counter isn't applicable if you configure your task to transfer all data. In that scenario, DataSync copies everything from the source to the destination without comparing differenc es between locations.

Console	DescribeT askExecution	Task mode support	Description
Processing rate		Enhanced, Basic	The rate at which DataSync reads files, objects, and directori es at your source location.  The processing rate is based on several CloudWatch metrics. The exact metrics depend on the task mode you're using.  Enhanced mode:  • FilesList edSource • FilesPrepared • FilesTran sferred • FilesVerified
			<ul> <li>FilesPrep aredSource</li> <li>FilesPrep aredDesti nation</li> <li>FilesTran sferred</li> <li>FilesVeri fiedSource</li> </ul>

Console	DescribeT askExecution	Task mode support	Description
			<ul><li>FilesVeri fiedDesti nation</li></ul>
Remaining		Basic	The remaining number of files, objects, and directori es that DataSync expects to transfer over the network.  If you're using   DescribeTaskExecut ion, you can calculate this counter by subtracting FilesTran sferred from   Estimated  FilesToTr ansfer.

Console	DescribeT askExecution	Task mode support	Description
Skipped*	FilesSkipped	Enhanced, Basic	The number of files, objects, and directori es that DataSync skips during your transfer.  * – For Enhanced mode, Skipped doesn't display in the console. Instead, skipped items are included in the Prepared counter when transferring only the data that has changed or the Transferred counter when transferring all data.

Console	DescribeT askExecution	Task mode support	Description
Transferred	FilesTran sferred	Enhanced, Basic	The number of files, objects, and directori es that DataSync transfers over the network. This value is updated periodica lly during your task execution when something is read from the source and sent over the network.  If DataSync fails to transfer something, this value can be less than Estimated FilesToTr ansfer . In some cases, this value can also be greater than Estimated FilesToTr ansfer . This counter is implement ation-specific for some location types, so don't use it as an exact indication of what's transferring or to monitor your task execution.

Console	DescribeT askExecution	Task mode support	Description
			In the console, this counter can also show you the number of objects that DataSync skips during the transfer of an Enhanced mode task. For more information, see <a href="HowDataSync transfers">HowDataSync transfers</a> your data.
Verified	FilesVerified	Enhanced, Basic	The number of files, objects, and directori es that DataSync verifies during your transfer.  When you configure your task to verify only transferred data, DataSync doesn't verify directories in some situations or files or objects that fail to transfer.

# Monitoring data transfers with Amazon CloudWatch metrics

Amazon CloudWatch provides metrics to track DataSync transfer performance and troubleshoot issues with your transfer task.

You can monitor AWS DataSync transfer performance by using Amazon CloudWatch metrics. DataSync metrics are automatically sent to CloudWatch in 5-minute intervals (regardless of how you configure logging). The metrics are retained for a period of 15 months.

To see CloudWatch metrics for DataSync, you can use the following tools:

- The CloudWatch console
- The CloudWatch CLI
- The CloudWatch API
- The DataSync console (on the task execution's details page)

For more information, see the Amazon CloudWatch User Guide.

# **CloudWatch metrics for DataSync**

DataSync metrics use the aws/datasync namespace and provide metrics for the following dimensions:

- **AgentId** The unique ID of the agent (if your task uses an agent).
- TaskId The unique ID of the task. It takes the form of task-01234567890abcdef.

The aws/datasync namespace includes the following metrics. Some metrics aren't available with every task mode.

CloudWatch metric	Task mode support	Description
BytesComp ressed	Basic	The number of physical bytes that DataSync transfers over the network after compression (if compression is possible). This number is typically less than BytesTran sferred unless the data isn't compressible.  Unit: Bytes
BytesPrep aredDesti nation	Basic	The number of logical bytes that DataSync prepares at the destination location.

CloudWatch metric	Task mode support	Description
		Unit: Bytes
BytesPrep aredSource	Basic	The number of logical bytes that DataSync prepares at the source location.
		Unit: Bytes
BytesTran sferred	Basic	The number of bytes that DataSync sends to the network before compression (if compression is possible) . For the number of bytes transferred over the network, see the BytesCompressed metric.
		Unit: Bytes
BytesVeri fiedDesti	Basic	The number of logical bytes that DataSync verifies at the destination location.
nation		Unit: Bytes
BytesVeri fiedSource	Basic	The number of logical bytes that DataSync verifies at the source location.
		Units: Bytes
BytesWritten	Enhanced, Basic	The number of logical bytes that DataSync writes to the destination location.
		Unit: Bytes
FilesDeleted	Enhanced, Basic	The number of files, objects, and directories that DataSync deletes in your destination location. If you don't configure your task to <u>delete data in the destination that isn't in the source</u> , the value is always 0.  Unit: Count

CloudWatch metric	Task mode support	Description
FilesList edSource	Enhanced	The number of objects that DataSync finds at your source location.
		Unit: Count
FilesPrep ared	Enhanced	The number of objects that DataSync will attempt to transfer after comparing your source and destination locations. For more information, see <a href="How DataSync">How DataSync</a> prepares your data transfer.  This metric isn't applicable if you configure your task to <a href="transfer all data">transfer all data</a> . In that scenario, DataSync copies everything from the source to the destination without comparing differences between the locations.
		Unit: Count
FilesPrep aredDesti nation	Basic	The number of files, objects, and directories that DataSync prepares at the destination location.  Unit: Count
FilesPrep aredSource	Basic	The number of files, objects, and directories that DataSync prepares at the source location.  Unit: Count
FilesSkipped	Basic	The number of files, objects, and directories that DataSync skips during your transfer.
		Unit: Count

CloudWatch metric	Task mode support	Description
FilesTran sferred		The number of files, objects, and directories that DataSync transfers over the network. This value is updated periodically during the <u>task execution</u> when something is read from the source and sent over the network.
		This value can be less than Estimated FilesToTransfer in a DescribeTaskExecut ion response if DataSync fails to transfer something. In some cases, this value can also be greater than EstimatedFilesToTransfer This metric is implementation-specific for some location types, so don't use it as an exact indication of what transferred or to monitor your task execution.  Unit: Count
FilesVeri fied	Enhanced	The number of objects that DataSync verifies during your transfer.  Unit: Count
FilesVeri fiedDesti nation	Basic	The number of files, objects, and directories that DataSync verifies at the destination location.  Unit: Count
FilesVeri fiedSource	Basic	The number of files, objects, and directories that DataSync verifies at the source location.  Unit: Count

# Monitoring your data transfers with task reports

*Task reports* provide detailed information about what AWS DataSync attempts to transfer, skip, verify, and delete during a task execution. For more information, see <a href="How DataSync transfers files">How DataSync transfers files</a>, objects, and directories.

Task reports are generated in JSON format. You can customize the level of detail in your reports:

- <u>Summary only task reports</u> give you the necessary details about your task execution, such as how many files transferred and whether DataSync could verify the data integrity of those files.
- <u>Standard task reports</u> include a summary plus detailed reports that list each file, object, or folder that DataSync attempts to transfer, skip, verify, and delete. With a standard task report, you can also specify the report level to show only the task execution's errors or its successes and errors.

#### **Use cases**

Here are some situations where task reports can help you monitor and audit your data transfers:

- When migrating millions of files, quickly identify files that DataSync has issues transferring.
- Verify chain-of-custody processes for your files.

# Summary only task reports

A report that's only a summary of a task execution includes the following details:

- The AWS account that ran the task execution
- The source and destination locations
- The total number of files, objects, and folders that were skipped, transferred, verified, and deleted
- The total bytes (logical and physical) that were transferred
- If the task execution was completed, canceled, or encountered an error
- The start and end times (including the total time of the transfer)
- The task's settings (such as bandwidth limits, data integrity verification, and other options for your DataSync transfer)

# Standard task reports

A standard task report includes a <u>summary</u> of your task execution plus detailed reports of what DataSync attempts to transfer, skip, verify, and delete.

#### **Topics**

- Report level
- Transferred reports
- Skipped reports
- Verified reports
- Deleted reports

# Report level

With standard task reports, you can choose one of the following report levels:

- · Errors only
- Successes and errors (essentially a list of everything that happened during your task execution)

For example, you might want to see which files DataSync skipped successfully during your transfer and which ones it didn't. Files that DataSync skipped successfully might be ones that you purposely want DataSync to exclude because they already exist in your destination location. However, a skipped error for instance might indicate that DataSync doesn't have the right permissions to read a file.

# **Transferred reports**

A list of files, objects, and directories that DataSync attempted to transfer during your task execution. A transferred report includes the following details:

- The paths for the transferred data
- What was transferred (content, metadata, or both)
- The metadata, which includes the data type, content size (objects and files only), and more
- · The time when an item was transferred
- The object version (if the destination is an Amazon S3 bucket that has versioning enabled)

Standard task reports 283

- If something was overwritten in the destination
- Whether an item transferred successfully

#### Note

When moving data between S3 buckets, the prefix that you specify in your source location can show up in your report (or in Amazon CloudWatch logs), even if that prefix doesn't exist as an object in your destination location. (In the DataSync console, you might also notice this prefix showing up as skipped or verified data.)

# **Skipped reports**

A list of files, objects, and directories that DataSync finds in your source location but didn't attempt to transfer. The reasons DataSync skips data can depend on several factors, such as how you configure your task and storage system permissions. Here are some examples:

- There's a file that exists in your source and destination locations. The file in the source hasn't been modified since the previous task execution. Since you're only transferring data that has changed, DataSync doesn't transfer that file next time you run your task.
- An object that exists in both of your locations changes in your source. When you run your task, DataSync skips this object in your destination because your task doesn't overwrite data in the destination.
- DataSync skips an object in your source that's using an archival storage class and isn't restored. You must restore an archived object for DataSync to read it.
- DataSync skips a file, object, or directory in your source location because it can't read it. If this happens and isn't expected, check your storage's access permissions and make sure that DataSync can read what was skipped.

A skipped report includes the following details:

- The paths for skipped data
- The time when an item was skipped
- The reason it was skipped
- Whether an item was skipped successfully

Standard task reports 284



#### Note

Skipped reports can be large when they include successes and errors, you configure your task to transfer only the data that has changed, and source data already exists in the destination.

### **Verified reports**

A list of files, objects, and directories that DataSync attempted to verify the integrity of during your task execution. A verified data report includes the following details:

- The paths for verified data
- The time when an item was verified
- The reason for the verification error (if any)
- The source and destination SHA256 checksums (files only)
- Whether an item was successfully verified

Note the following about verified reports:

- When you configure your task to verify only transferred data, DataSync doesn't verify directories in some situations or files or objects that fail to transfer. In either case, DataSync doesn't include unverified data in this report.
- If you're using Enhanced mode, verification might take longer than usual if you're transferring large objects.

# **Deleted reports**

A list of files, directories, and objects that were deleted during your task execution. DataSync generates this report only if you configure your task to delete data in the destination location that isn't in the source. A deleted data report includes the following details:

- The paths for deleted data
- Whether an item was successfully deleted
- · The time when an item was deleted

Standard task reports 285

# **Example task reports**

The level of detail in your task report is up to you. Here are some example transferred data reports with the following configuration:

- Report type Standard
- Report level Successes and errors



Reports use the ISO-8601 standard for the timestamp format. Times are in UTC and measured in nanoseconds. This behavior differs from how some other task report metrics are measured. For example, <u>task execution details</u>, such as TransferDuration and VerifyDuration, are measured in milliseconds.

Enhanced mode task reports use a somewhat different schema than Basic mode task reports. The following examples can help you know what to expect from your reports depending on the <u>task</u> mode you use.

#### Example transferred data reports with success status

The following reports show successful transfers for an object named object 1.txt.

Enhanced mode

```
{
    "TaskExecutionId": "exec-abcdefgh12345678",
    "Transferred": [{
        "RelativePath": "object1.txt",
        "SourceMetadata": {
            "Type": "Object",
            "ContentSize": 6,
            "LastModified": "2024-10-04T14:40:55Z",
            "SystemMetadata": {
                  "ContentType": "binary/octet-stream",
                  "ETag": "\"9b2d7e1f8054c3a2041905d0378e6f14\"",
                  "ServerSideEncryption": "AES256"
            },
            "UserMetadata": {},
```

Example task reports 286

```
"Tags": []
},
"Overwrite": "False",
"DstS3VersionId": "jtqRtX3jN4J2G8k0sFSGYK1f35KqpAVP",
"TransferTimestamp": "2024-10-04T14:48:39.748862183Z",
"TransferType": "CONTENT_AND_METADATA",
"TransferStatus": "SUCCESS"
}]
}
```

#### Basic mode

```
{
    "TaskExecutionId": "exec-abcdefgh12345678",
    "Transferred": [{
        "RelativePath": "/object1.txt",
        "SrcMetadata": {
            "Type": "Regular",
            "ContentSize": 6,
            "Mtime": "2022-01-07T16:59:26.136114671Z",
            "Atime": "2022-01-07T16:59:26.136114671Z",
            "Uid": 0,
            "Gid": 0,
            "Mode": "0644"
        },
        "Overwrite": "False",
        "DstS3VersionId": "jtqRtX3jN4J2G8k0sFSGYK1f35KqpAVP",
        "TransferTimestamp": "2022-01-07T16:59:45.747270957Z",
        "TransferType": "CONTENT_AND_METADATA",
        "TransferStatus": "SUCCESS"
    }]
}
```

#### Example transferred data reports with error status

The following reports provide examples of when DataSync can't transfer an object named object1.txt.

#### Enhanced mode

This report shows that DataSync can't access an object named object1.txt because of an AWS KMS permissions issue. (If you get an error like this, see <a href="Accessing S3">Accessing S3</a> buckets using server-side encryption.)

Example task reports 287

```
{
    "TaskExecutionId": "exec-abcdefgh12345678",
    "Transferred": [{
        "RelativePath": "object1.txt",
        "SourceMetadata": {
            "Type": "Object",
            "ContentSize": 6,
            "LastModified": "2022-10-07T20:48:32Z",
            "SystemMetadata": {
                "ContentType": "binary/octet-stream",
                "ETag": "\"3a7c0b2f1d9e5c4a6f8b2e0d1c9f7a3b2\"",
                "ServerSideEncryption": "AES256"
            },
            "UserMetadata": {},
            "Tags": []
        },
        "Overwrite": "False",
        "TransferTimestamp": "2022-10-09T16:05:11.134040717Z",
        "TransferType": "CONTENT_AND_METADATA",
        "TransferStatus": "FAILED",
        "ErrorCode": "AccessDenied",
        "ErrorDetail": "User: arn:aws:sts::111222333444:assumed-role/
AWSDataSyncS3Bucket/AwsSync-loc-0b3017fc4ba4a2d8d is not authorized
 to perform: kms:GenerateDataKey on resource: arn:aws:kms:us-
east-1:111222333444:key/1111aaaa-22bb-33cc-44d-5555eeee6666 because no identity-
based policy allows the kms:GenerateDataKey action"
    }]
}
```

#### Basic mode

This report shows that an object named object1.txt didn't transfer because of an S3 bucket permissions issue. (If you get an error like this, see <a href="Providing DataSync access to S3">Providing DataSync access to S3</a> buckets.)

```
{
   "TaskExecutionId": "exec-abcdefgh12345678",
   "Transferred": [{
        "RelativePath": "/object1.txt",
        "SrcMetadata": {
            "Type": "Regular",
            "ContentSize": 6,
            "Mtime": "2022-01-07T16:59:26.136114671Z",
```

Example task reports 288

### Limitations

- Individual task reports can't exceed 5 MB. If you're copying a large number of files, your task report might be split into multiple reports.
- There are situations when creating task reports can affect the performance of your data transfer. For example, you might notice this when your network connection has high latency and the files you're transferring are small or you're copying only metadata changes.

# Creating your DataSync task reports

AWS DataSync task reports can be only a summary of your task execution or a set of detailed reports about what DataSync attempts to transfer, skip, verify, and delete.

# **Prerequisites**

Before you can create a task report, you must do the following.

### **Topics**

- Create an S3 bucket for your task reports
- Allow DataSync to upload task reports to your S3 bucket

Limitations 289

#### Create an S3 bucket for your task reports

If you don't already have one, <u>create an S3 bucket</u> where DataSync can upload your task report. Reports are stored in the S3 Standard storage class.

We recommend the following for this bucket:

- If you're planning to transfer data to an S3 bucket, don't use the same bucket for your task report if you <u>disable the **Keep deleted files option**</u>. Otherwise, DataSync will delete any previous task reports each time you execute a task since those reports don't exist in your source location.
- To avoid a complex access permissions setup, make sure that your task report bucket is in the same AWS account and Region as your DataSync transfer task.

#### Allow DataSync to upload task reports to your S3 bucket

You must configure an AWS Identity and Access Management (IAM) role that allows DataSync to upload a task report to your S3 bucket.

In the DataSync console, you can create an IAM role that in most cases automatically includes the permissions to upload a task report to your bucket. Keep in mind that this automatically generated role might not meet your needs from a least-privilege standpoint. This role also won't work if your bucket is encrypted with a customer managed AWS Key Management Service (AWS KMS) key (SSE-KMS). In these cases, you can create the role manually as long as the role does at least the following:

• Prevents the cross-service confused deputy problem in the role's trusted entity.

The following full example shows how you can use the aws: SourceArn and aws: SourceAccount global condition context keys to prevent the confused deputy problem with DataSync.

**JSON** 

• Allows DataSync to upload a task report to your S3 bucket.

The following example does this by including the s3: PutObject action only for a specific prefix (reports/) in your bucket.

**JSON** 

• If your S3 bucket is encrypted with a customer managed SSE-KMS key, the <u>key's policy</u> must include the IAM role that DataSync uses to access the bucket.

For more information, see <u>Accessing S3 buckets using server-side encryption</u>.

# Creating a summary only task report

You can configure a task report that includes a <u>summary only</u> when creating your DataSync task, starting your task, or updating your task.

The following steps show how to configure a summary only task report when creating a task.

#### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/. 1.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.
- 3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- Scroll down to the **Task report** section. For **Report type**, choose **Summary only**. 4.
- For **S3 bucket for reports**, choose an S3 bucket where you want DataSync to upload your task 5. report.



#### (i) Tip

If you're planning to transfer data to an S3 bucket, don't use the same bucket for your task report if you disable the **Keep deleted files** option. Otherwise, DataSync will delete any previous task reports each time you execute a task since those reports don't exist in your source location.

For **Folder**, enter a prefix to use for your task report when DataSync uploads the report to your S3 bucket (for example, **reports/**).

Make sure to include the appropriate delimiter character at the end of your prefix. This character is usually a forward slash (/). For more information, see Organizing objects by using prefixes in the Amazon S3 User Guide.

- For **IAM role**, do one of the following: 7.
  - Choose Autogenerate to have DataSync automatically create an IAM role with the permissions that are required to access the S3 bucket.

If DataSync previously created an IAM role for this S3 bucket, that role is chosen by default.

Choose a custom IAM role that you created.

In some cases, you might need to create the role yourself. For more information, see Allow DataSync to upload task reports to your S3 bucket.

#### Important

If your S3 bucket is encrypted with a customer managed SSE-KMS key, the key's policy must include the IAM role that DataSync uses to access the bucket. For more information, see Accessing S3 buckets using server-side encryption.

Finish creating your task, and then start the task to begin transferring your data.

When your transfer is complete, you can view your task report.

#### Using the AWS CLI

Copy the following create-task AWS Command Line Interface (AWS CLI) command:

```
aws datasync create-task \
  --source-location-arn arn:aws:datasync:us-east-1:123456789012:location/
loc-12345678abcdefgh \
  --destination-location-arm arm:aws:datasync:us-east-1:123456789012:location/
loc-abcdefqh12345678 \
  --task-report-config '{
    "Destination":{
      "S3":{
        "Subdirectory":"reports/",
        "S3BucketArn": "arn:aws:s3:::your-task-reports-bucket",
        "BucketAccessRoleArn": "arn:aws:iam::123456789012:role/bucket-iam-role"
        }
    },
    "OutputType": "SUMMARY_ONLY"
  }'
```

- 2. For the --source-location-arn parameter, specify the Amazon Resource Name (ARN) of the source location in your transfer. Replace us-east-1 with the appropriate AWS Region, replace 123456789012 with the appropriate AWS account number, and replace 12345678abcdefgh with the appropriate source location ID.
- For the --destination-location-arn parameter, specify the ARN of the destination location in your transfer. Replace <u>us-east-1</u> with the appropriate AWS Region, replace 123456789012 with the appropriate AWS account number, and replace abcdefgh12345678 with the appropriate destination location ID.
- For the --task-report-config parameter, do the following:

• Subdirectory - Replace reports/ with the prefix in your S3 bucket where you want DataSync to upload your task reports.

Make sure to include the appropriate delimiter character at the end of your prefix. This character is usually a forward slash (/). For more information, see Organizing objects by using prefixes in the Amazon S3 User Guide.

• S3BucketArn – Specify the ARN of the S3 bucket where you want to upload your task report.



#### (i) Tip

If you're planning to transfer data to an S3 bucket, don't use the same bucket for your task report if you disable the **Keep deleted files** option. Otherwise, DataSync will delete any previous task reports each time you execute a task since those reports don't exist in your source location.

 BucketAccessRoleArn – Specify the IAM role that allows DataSync to upload a task report to your S3 bucket.

For more information, see Allow DataSync to upload task reports to your S3 bucket.



#### Important

If your S3 bucket is encrypted with a customer managed SSE-KMS key, the key's policy must include the IAM role that DataSync uses to access the bucket. For more information, see Accessing S3 buckets using server-side encryption.

OutputType – Specify SUMMARY\_ONLY.

For more information, see Summary only task reports.

5. Run the create-task command to create your task.

You get a response like the following that shows you the ARN of the task that you created. You will need this ARN to run the start-task-execution command.

```
{
    "TaskArn": "arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh"
}
```

6. Copy the following start-task-execution command.

```
aws datasync-task-report start-task-execution \
--task-arn arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh
```

 For the --task-arn parameter, specify the ARN of the task that you're starting. Use the ARN that you received from running the create-task command.

8. Run the start-task-execution command.

When your transfer is complete, you can view your task report.

### Creating a standard task report

You can configure a <u>standard task report</u> when creating your DataSync task, starting your task, or updating your task.

The following steps show how to configure a standard task report when creating a task.

#### Using the DataSync console

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create task**.
- 3. Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

- 4. Scroll down to the **Task report** section. For **Report type**, choose **Standard report**.
- 5. For **Report level**, choose one of the following:
  - **Errors only** Your task report includes only issues with what DataSync tried to transfer, skip, verify, and delete.
  - **Successes and errors** Your task report includes what DataSync successfully transferred, skipped, verified, and deleted and what it didn't.
  - **Custom** Allows you to choose whether you want to see errors only or successes and errors for specific aspects of your task report.

For example, you can choose **Successes and errors** for the transferred files list but **Errors only** for the rest of the report.

If you're transferring to an S3 bucket that uses object versioning, keep **Include Amazon** 6. **S3 object versions** selected if you want your report to include the new version for each transferred object.

For S3 bucket for reports, choose an S3 bucket where you want DataSync to upload your task report.



#### (i) Tip

If you're planning to transfer data to an S3 bucket, don't use the same bucket for your task report if you disable the **Keep deleted files** option. Otherwise, DataSync will delete any previous task reports each time you execute a task since those reports don't exist in your source location.

- For **Folder**, enter a prefix to use for your task report when DataSync uploads the report to your S3 bucket (for example, **reports**/). Make sure to include the appropriate delimiter character at the end of your prefix. This character is usually a forward slash (/). For more information, see Organizing objects by using prefixes in the Amazon S3 User Guide.
- For **IAM role**, do one of the following:
  - Choose Autogenerate to have DataSync automatically create an IAM role with the permissions that are required to access the S3 bucket.

If DataSync previously created an IAM role for this S3 bucket, that role is chosen by default.

Choose a custom IAM role that you created.

In some cases, you might need to create the role yourself. For more information, see Allow DataSync to upload task reports to your S3 bucket.



#### Important

If your S3 bucket is encrypted with a customer managed SSE-KMS key, the key's policy must include the IAM role that DataSync uses to access the bucket. For more information, see Accessing S3 buckets using server-side encryption.

10. Finish creating your task and start the task to begin transferring your data.

When your transfer is complete, you can view your task report.

#### **Using the AWS CLI**

Copy the following create-task command:

```
aws datasync create-task \
  --source-location-arn arn:aws:datasync:us-east-1:123456789012:location/
loc-12345678abcdefgh \
  --destination-location-arn arn:aws:datasync:us-east-1:123456789012:location/
loc-abcdefgh12345678 \
  --task-report-config '{
    "Destination":{
      "S3":{
        "Subdirectory":"reports/",
        "S3BucketArn": "arn:aws:s3:::your-task-reports-bucket",
        "BucketAccessRoleArn": "arn:aws:iam::123456789012:role/bucket-iam-role"
        }
    },
    "OutputType": "STANDARD",
    "ReportLevel": "level-of-detail",
    "ObjectVersionIds": "include-or-not"
  }'
```

- 2. For the --source-location-arn parameter, specify the ARN of the source location in your transfer. Replace *us-east-1* with the appropriate AWS Region, replace *123456789012* with the appropriate AWS account number, and replace *12345678abcdefgh* with the appropriate source location ID.
- 3. For the --destination-location-arn parameter, specify the ARN of the destination location in your transfer. Replace *us-east-1* with the appropriate AWS Region, replace *123456789012* with the appropriate AWS account number, and replace *abcdefgh12345678* with the appropriate destination location ID.
- 4. For the --task-report-config parameter, do the following:
  - Subdirectory Replace reports/ with the prefix in your S3 bucket where you want
    DataSync to upload your task reports. Make sure to include the appropriate delimiter
    character at the end of your prefix. This character is usually a forward slash (/). For more
    information, see Organizing objects by using prefixes in the Amazon S3 User Guide.
  - S3BucketArn Specify the ARN of the S3 bucket where you want to upload your task report.



#### (i) Tip

If you're planning to transfer data to an S3 bucket, don't use the same bucket for your task report if you disable the **Keep deleted files** option. Otherwise, DataSync will delete any previous task reports each time you execute a task since those reports don't exist in your source location.

• BucketAccessRoleArn – Specify the IAM role that allows DataSync to upload a task report to your S3 bucket.

For more information, see Allow DataSync to upload task reports to your S3 bucket.



#### Important

If your S3 bucket is encrypted with a customer managed SSE-KMS key, the key's policy must include the IAM role that DataSync uses to access the bucket. For more information, see Accessing S3 buckets using server-side encryption.

OutputType – Specify STANDARD report.

For more information, see Standard task reportsTypes of task reports.

- (Optional) ReportLevel Specify whether you want ERRORS\_ONLY (the default) or SUCCESSES\_AND\_ERRORS in your report.
- (Optional) ObjectVersionIds If you're transferring to an S3 bucket that uses object versioning, specify NONE if you don't want to include the new version for each transferred object in the report.

By default, this option is set to INCLUDE.

• (Optional) Overrides – Customize the ReportLevel of a particular aspect of your report.

For example, you might want to see SUCCESSES\_AND\_ERRORS for the list of what DataSync deletes in your destination location, but you want ERRORS\_ONLY for everything else. In this example, you would add the following Overrides option to the --task-report-config parameter:

```
"Overrides":{
  "Deleted":{
```

```
"ReportLevel":"SUCCESSES_AND_ERRORS"
}
```

If you don't use Overrides, your entire report uses the ReportLevel that you specify.

5. Run the create-task command to create your task.

You get a response like the following that shows you the ARN of the task that you created. You will need this ARN to run the start-task-execution command.

```
{
    "TaskArn": "arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh"
}
```

6. Copy the following start-task-execution command.

```
aws datasync-task-report start-task-execution \
    --task-arn arn:aws:datasync:us-east-1:123456789012:task/task-12345678abcdefgh
```

- For the --task-arn parameter, specify the ARN of the task you're running. Use the ARN that you received from running the create-task command.
- Run the start-task-execution command.

When your transfer is complete, you can view your task report.

# Viewing your DataSync task reports

DataSync creates task reports for every task execution. When your execution completes, you can find the related task reports in your S3 bucket. Task reports are organized under prefixes that include the IDs of your tasks and their executions.

To help locate task reports in your S3 bucket, use these examples:

- Summary only task report reports-prefix/Summary-Reports/task-id-folder/taskexecution-id-folder
- Standard task report reports-prefix/Detailed-Reports/task-id-folder/taskexecution-id-folder

Because task reports are in JSON format, you have several options for viewing your reports:

Viewing your task reports 299

- View a report by using Amazon S3 Select.
- Visualize reports by using AWS services such as AWS Glue, Amazon Athena, and Amazon
  QuickSight. For more information about visualizing your task reports, see the AWS Storage Blog.

# Monitoring data transfers with Amazon CloudWatch Logs

You can monitor your AWS DataSync transfer by using CloudWatch Logs. We recommend that you configure your task to at least log basic information (such as transfer errors).

# Allowing DataSync to upload logs to a CloudWatch log group

To <u>configure logging</u> for your DataSync task, you need a CloudWatch log group that DataSync has permission to send logs to. You set up this access through an AWS Identity and Access Management (IAM) role. How this specifically works depends on your task mode.

#### Enhanced mode

With Enhanced mode, DataSync automatically sends task logs to a log group named /aws/datasync. If that log group doesn't exist in your AWS Region, DataSync creates the log group on your behalf by using an IAM service-linked role when you create your task.

#### Basic mode

There are a couple ways to set up a CloudWatch log group for a DataSync task using Basic mode. In the console, you can automatically create an IAM role that in most cases includes the permissions that DataSync requires to upload logs. Keep in mind that this automatically generated role might not meet your needs from a least-privilege standpoint.

If you want to use an existing CloudWatch log group or are creating your tasks programmatically, you must create the IAM role yourself.

The following example is an IAM policy that grants these permissions.

```
"Effect": "Allow",
            "Action": [
                "logs:PutLogEvents",
                "logs:CreateLogStream"
            ],
            "Principal": {
                "Service": "datasync.amazonaws.com"
            },
            "Condition": {
                "ArnLike": {
                     "aws:SourceArn": [
                         "arn:aws:datasync:region:account-id:task/*"
                    ]
                },
                "StringEquals": {
                     "aws:SourceAccount": "account-id"
                }
            },
            "Resource": "arn:aws:logs:region:account-id:log-group:*:*"
        }
    ]
}
```

The policy uses Condition statements to help ensure that only DataSync tasks from the specified account have access to the specified CloudWatch log group. We recommend using the <a href="mailto:aws:SourceArn">aws:SourceArn</a> and <a href="mailto:aws:SourceAccount">aws:SourceAccount</a> global condition context keys in these Condition statements to protect against the confused deputy problem. For more information, see <a href="mailto:Cross-service confused deputy prevention">Cross-service confused deputy prevention</a>.

To specify the DataSync task or tasks, replace <code>region</code> with the Region code for the AWS Region where the tasks are located (for example, us-west-2), and replace <code>account-id</code> with the AWS account ID of the account that contains the tasks. To specify the CloudWatch log group, replace the same values. You can also modify the Resource statement to target specific log groups. For more information about using <code>SourceArn</code> and <code>SourceAccount</code>, see <code>Global</code> condition keys in the <code>IAM User Guide</code>.

To apply the policy, save this policy statement to a file on your local computer. Then run the following AWS CLI command to apply the resource policy. To use this example command, replace <code>full-path-to-policy-file</code> with the path to the file that contains your policy statement.

aws logs put-resource-policy --policy-name trust-datasync --policy-document file://full-path-to-policy-file



#### Note

Run this command by using the same AWS account and AWS Region where you activated your DataSync agent.

For more information, see the Amazon CloudWatch Logs User Guide.

# Configuring logging for your DataSync task

We recommend that you configure at least some level of logging for your DataSync task.

#### Before you begin

DataSync needs permission to upload logs to a CloudWatch log group. For more information, see Allowing DataSync to upload logs to a CloudWatch log group.

#### Using the DataSync console

The following instructions describe how to configure CloudWatch logging when creating a task. You also can configure logging when editing a task.

- Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**, and then choose **Create** task.
- Configure your task's source and destination locations.

For more information, see Where can I transfer my data with AWS DataSync?

On the **Configure settings** page, choose a task mode and any other options. 4.

You might be interested in some of the following options:

- Specify what data to transfer by using a manifest or filters.
- Configure how to handle file metadata and verify data integrity.

- 5. For **Log level**, choose one of the following options:
  - Log basic information such as transfer errors Publish logs with only basic information (such as transfer errors).
  - Log all transferred objects and files Publish logs for all files or objects that DataSync transfers and performs data-integrity checks on.
  - Don't generate logs
- Do one of the following depending on the task mode you're using to create or specify a CloudWatch log group:

#### Enhanced mode

When you choose **Create task**, DataSync automatically uses (or creates) a log group named /aws/datasync.

#### Basic mode

For **CloudWatch log group**, specify a log group that DataSync has permission to upload logs to by doing one of the following:

- Choose Autogenerate to automatically create a log group that allows DataSync to upload logs to it.
- Choose an existing log group in your current AWS Region.

If you choose an existing log group, make sure that <u>DataSync has permission</u> to upload logs to that log group.

7. Choose **Create task**.

You're ready to <u>start your task</u>.

#### Using the AWS CLI

Copy the following create-task command:

```
aws datasync create-task \
    --source-location-arn "arn:aws:datasync:us-east-1:account-id:location/location-
id" \
    --destination-location-arn "arn:aws:datasync:us-east-1:account-
id:location/location-id" \
    --task-mode "ENHANCED-or-BASIC" \
```

```
--name "task-name" \
--options '{"LogLevel": "log-level"}' \
--cloudwatch-log-group-arn "arn:aws:logs:us-east-1:account-id:log-group:log-group-name:*"
```

- 2. For --source-location-arn, specify the Amazon Resource Name (ARN) of your source location.
- 3. For --destination-location-arn, specify the ARN of your destination location.

If you're transferring across AWS Regions or accounts, make sure that the ARN includes the other Region or account ID.

- 4. For --task-mode, specify ENHANCED or BASIC.
- 5. (Recommended) For --name, specify a name for your task that you can remember.
- 6. For LogLevel, specify one of the following options:
  - BASIC Publish logs with only basic information (such as transfer errors).
  - TRANSFER Publish logs for all files or objects that DataSync transfers and performs dataintegrity checks on.
  - NONE Don't generate logs.
- 7. For --cloudwatch-log-group-arn, specify the ARN of a CloudWatch log group.

### Important

If your --task-mode is ENHANCED, you don't need to specify this option. For more information, see Allowing DataSync to upload logs to a CloudWatch log group.

8. Run the create-task command.

If the command is successful, you get a response that shows you the ARN of the task that you created. For example:

```
{
    "TaskArn": "arn:aws:datasync:us-east-1:111222333444:task/
    task-08de6e6697796f026"
}
```

You're ready to start your task.

#### Using the DataSync API

You can configure CloudWatch logging for your task by using the CloudWatchLogGroupArn parameter with any of the following operations:

- CreateTask
- UpdateTask

# **Viewing DataSync task logs**

When you <u>start your task</u>, you can view the task execution's logs by using the CloudWatch console or AWS CLI (among other options). For more information, see the <u>Amazon CloudWatch Logs User</u> Guide.

DataSync provides JSON-structured logs for Enhanced mode tasks. Basic mode tasks have unstructured logs. The following examples show how verification errors display in Enhanced mode logs compared to Basic mode logs.

Enhanced mode log example

```
{
    "Action": "VERIFY",
    "Source": {
        "LocationId": "loc-abcdef01234567890",
        "RelativePath": "directory1/directory2/file1.txt"
    },
    "Destination": {
        "LocationId": "loc-05ab2fdc272204a5f",
        "RelativePath": "directory1/directory2/file1.txt",
        "Metadata": {
            "Type": "Object",
            "ContentSize": 66060288,
            "LastModified": "2024-10-03T20:46:58Z",
            "S3": {
                "SystemMetadata": {
                    "ContentType": "binary/octet-stream",
                    "ETag": "\"1234abcd5678efgh9012ijkl3456mnop\"",
                     "ServerSideEncryption": "AES256"
                },
                "UserMetadata": {
                     "file-mtime": "1602647222/222919600"
```

Viewing DataSync task logs 305

```
},
    "Tags": {}
}
}

},

"ErrorCode": "FileNotAtSource",
    "ErrorDetail": "Verification failed due to file being present at the destination but not at the source"
}
```

#### Basic mode log example

# Logging AWS DataSync API calls with AWS CloudTrail

AWS DataSync is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in DataSync. CloudTrail captures all API calls for DataSync as events. The calls that are captured include calls from the DataSync console and code calls to DataSync API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for AWS DataSync. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to AWS DataSync, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

# Working with DataSync information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in AWS DataSync, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing events with CloudTrail event history.

Logging with CloudTrail 306

For an ongoing record of events in your AWS account, including events for AWS DataSync, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all AWS Regions in the same AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple Regions and Receiving CloudTrail log files from multiple accounts

All DataSync actions are logged by CloudTrail. (For more information, see the DataSync <u>API</u> reference.)

For example, calls to the CreateAgent, CreateTask, and ListLocations operations generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or AWS Identity and Access Management (IAM)
  credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see CloudTrail userIdentity element in the AWS CloudTrail User Guide.

# **Understanding DataSync log file entries**

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, the request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the CreateTask operation.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "1234567890abcdef0",
        "arn": "arn:aws:iam::123456789012:user/user1",
        "accountId": "123456789012",
        "accessKeyId": "access key",
        "userName": "user1",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-12-13T14:56:46Z"
            }
        },
        "invokedBy": "signin.amazonaws.com"
    },
    "eventTime": "2018-12-13T14:57:02Z",
    "eventSource": "datasync.amazonaws.com",
    "eventName": "CreateTask",
    "awsRegion": "ap-southeast-1",
    "sourceIPAddress": "192.0.2.1",
    "userAgent": "signin.amazonaws.com",
    "requestParameters": {
        "cloudWatchLogGroupArn": "arn:aws:logs:ap-southeast-1:123456789012:log-
group: MyLogGroup",
        "name": "MyTask-NTIzMzY1",
        "tags": [],
        "destinationLocationArn": "arn:aws:datasync:ap-
southeast-1:123456789012:location/loc-abcdef01234567890",
        "options": {
            "bytesPerSecond": -1,
            "verifyMode": "POINT_IN_TIME_CONSISTENT",
            "uid": "INT_VALUE",
            "posixPermissions": "PRESERVE",
            "mtime": "PRESERVE",
            "gid": "INT_VALUE",
            "preserveDevices": "NONE",
            "preserveDeletedFiles": "REMOVE",
            "atime": "BEST_EFFORT"
```

```
},
    "sourceLocationArn": "arn:aws:datasync:ap-southeast-1:123456789012:location/
loc-021345abcdef6789"
    },
    "responseElements": {
        "taskArn": "arn:aws:datasync:ap-southeast-1:123456789012:task/
task-1234567890abcdef0"
    },
    "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```

# Monitoring events by using Amazon EventBridge

Amazon EventBridge events describe changes in DataSync resources. You can set up rules to match these events and route them to one or more target functions or streams. Events are emitted on a best-effort basis.

# **DataSync transfer events**

The following EventBridge events are available for DataSync transfers.

#### **Agent state changes**

Event	Description
Online	The agent is configured properly and ready to use. This is the normal running status for an agent.
Offline	The agent has been out of contact with the DataSync service for five minutes or longer. This can happen for a few reasons. For more information, see <a href="What do I do if my agent is offline?">What do I do if my agent is offline?</a>

#### Location state changes

Monitoring with EventBridge 309

#### Agent state changes

**Event** Description

Adding DataSync is adding a location.

Available The location is created and is available to use.

Task state changes

**Event Description** 

Available The task was created and is ready to start.

Running The task is in progress and functioning

properly.

Unavailable The task isn't configured properly and can't be

used. You might see this event when an agent

associated with the task goes offline.

Queued Another task is running and using the same

agent. DataSync runs tasks in series (first in,

first out).

Task execution state changes

**Event Description** 

Queueing Another task execution is running and using

the same DataSync agent. For more informati

on, see Knowing when your task is queued.

Launching DataSync is initializing the task execution. This

status usually goes quickly but can take up to

a few minutes.

DataSync transfer events 310

#### **Agent state changes**

Preparing DataSync is determining what data to transfer.

> This step can take just minutes or a few hours depending on the number of files, objects, or directories in both locations and on how you configure your task. Preparation also might not be applicable to your task. For more information, see How DataSync prepares your

data transfer.

**Transferring** DataSync is performing the actual data

transfer.

Verifying DataSync is performing a data-integrity check

at the end of the transfer.

The task execution succeeded. Success

Cancelling The task execution is in the process of being

cancelled.

The task execution failed. Error

# Monitoring AWS DataSync with manual tools

You can track your AWS DataSync transfers from the console or the command line.

# Monitoring your transfer by using the DataSync console

You can monitor your DataSync transfer by using the console, which provides real-time metrics such as data transferred, data and file throughput, and data compression.

### To monitor your transfer by using the DataSync console

- After you start your DataSync task, choose **See execution details**. 1.
- 2. View metrics about your transfer.

# Monitoring your transfer by using the AWS CLI

You can monitor your DataSync transfer by using the AWS Command Line Interface (AWS CLI).

Copy the following describe-task-execution command. To use this example command, replace the *user input placeholders* with your own information.

```
aws datasync describe-task-execution \
    --task-execution-arn 'arn:aws:datasync:region:account-id:task/task-id/execution/task-
execution-id'
```

This command returns information about a task execution similar to that shown following.

```
{
    "BytesCompressed": 3500,
    "BytesTransferred": 5000,
    "BytesWritten": 5000,
    "EstimatedBytesToTransfer": 5000,
    "EstimatedFilesToDelete": 10,
    "EstimatedFilesToTransfer": 100,
    "FilesDeleted": 10,
    "FilesSkipped": 0,
    "FilesTransferred": 100,
    "FilesVerified": 100,
    "Result": {
        "ErrorCode": "??????",
        "ErrorDetail": "??????",
        "PrepareDuration": 100,
        "PrepareStatus": "SUCCESS",
        "TransferDuration": 60,
        "TransferStatus": "AVAILABLE",
        "VerifyDuration": 30,
        "VerifyStatus": "SUCCESS"
    },
    "StartTime": 1532660733.39,
    "Status": "SUCCESS",
    "OverrideOptions": {
        "Atime": "BEST_EFFORT",
        "BytesPerSecond": "1000",
        "Gid": "NONE",
        "Mtime": "PRESERVE",
        "PosixPermissions": "PRESERVE",
```

```
"PreserveDevices": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "Uid": "NONE",
        "VerifyMode": "POINT_IN_TIME_CONSISTENT"
    },
    "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-
aaaabbbbccccddddf/execution/exec-1234abcd1234abcd1",
    "TaskReportConfig": {
        "Destination": {
            "S3": {
                "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/my-datasync-
role",
                "S3BucketArn": "arn:aws:s3:::amzn-s3-demo-bucket/*",
                "Subdirectory": "reports"
            }
        },
        "ObjectVersionIds": "INCLUDE",
        "OutputType": "STANDARD",
        "Overrides": {
            "Deleted": {
                "ReportLevel": "ERRORS_ONLY"
            },
            "Skipped": {
                "ReportLevel": "SUCCESSES_AND_ERRORS"
            },
            "Transferred": {
                "ReportLevel": "ERRORS_ONLY"
            },
            "Verified": {
                "ReportLevel": "ERRORS_ONLY"
            }
        },
        "ReportLevel": "ERRORS_ONLY"
    }
}
```

- If the task execution succeeds, the value of **Status** changes to **SUCCESS**. For information about what the response elements mean, see DescribeTaskExecution.
- If the task execution fails, the result sends error codes that can help you troubleshoot issues. For information about the error codes, see TaskExecutionResultDetail.

# Monitoring your transfer by using the watch utility

To monitor the progress of your task in real time from the command line, you can use the standard Unix watch utility. Task execution duration values are measured in milliseconds.

The watch utility doesn't recognize the DataSync alias. The following example shows how to invoke the CLI directly. To use this example command, replace the *user input placeholders* with your own information.

# Managing AWS DataSync resources

Learn how to manage your AWS DataSync resources, such as agents, locations, and tasks.

# Managing your DataSync agent

Once you activate a DataSync agent, AWS manages the agent for you (including software updates). Learn more

# Testing your DataSync agent's connectivity and system resources

While AWS manages your DataSync agent once it's deployed and activated, there might be cases where you need to change your agent's settings or troubleshoot an issue. Learn more

# Replacing your DataSync agent

To replace a DataSync agent, you must create a new agent and update any locations that are using the old agent. <u>Learn more</u>

# Cleaning up DataSync resources

If you used DataSync for a test or just no longer need its resources, delete those resources so that you aren't charged for them. Learn more

# Reusing a DataSync agent's infrastructure

After you delete an agent resource from DataSync, you can still use the agent's virtual machine or Amazon EC2 instance to activate a new agent. <u>Learn more</u>

# Managing your AWS DataSync agent

Once you <u>activate an AWS DataSync agent</u>, AWS manages the virtual machine (VM) appliance for you.

# **Agent software updates**

AWS automatically updates your agent's software, including the underlying operating system and related DataSync software packages.

DataSync updates your agent only when it's idle. For example, your agent won't be updated until your transfer is complete.

The agent might go offline briefly following updates. This can happen, for instance, shortly after <u>agent activation</u> when AWS updates the agent.

### ▲ Important

- DataSync automatically and regularly patches agents to maintain their security and stability. DataSync agents use Amazon Linux 2 as their base operating system. You can view the current status of detected Common Vulnerabilities and Exposures (CVE) issues on the <u>Amazon Linux Security Center</u>. CVE patches are automatically applied within 30 days of their release date, as indicated on the Amazon Linux Security Center. Patching occurs as long as your agent is online and not actively running a task execution.
- DataSync doesn't support updating an Amazon EC2 agent manually with cloud-init directives. If you update an agent this way, you may encounter interoperability problems with DataSync where you can't activate or use the agent.

# **Agent statuses**

The following table describes the status of DataSync agents.

Agent status	Meaning
Online	The agent is configured properly and ready to use. This is the normal running status for an agent.
Offline	The agent has been out of contact with the DataSync service for five minutes or longer. This can happen for a few reasons. For more

Agent software updates 316

Agent status	Meaning
	information, see What do I do if my agent is offline?

### **Troubleshooting your agent**

While AWS manages the DataSync agent for you, there are situations when you might need to again work directly with it. For example, if your agent goes offline or loses its connection to your on-premises storage system, you can try to resolve these issues in the agent's local console.

For more information, see troubleshooting DataSync agents.

# Performing maintenance on your agent

While AWS manages your AWS DataSync agent once it's deployed and activated, there might be cases where you need to change your agent's settings or troubleshoot an issue. Here are some examples of why you'd work with your agent through its local console:

- Manually assign an IP address to the agent.
- Check your agent's system resources.

#### Important

You don't need to use the agent's local console for standard DataSync functionality.

# Accessing your agent's local console

How you access the local console depends on the type of agent you're using.

### Accessing the local console (VMware ESXi, Linux KVM, or Microsoft Hyper-V)

For security reasons, you can't remotely connect to the local console of the DataSync agent virtual machine (VM).

If this is your first time using the local console, log in with the default credentials. The default user name is **admin** and the password is **password**.

317 Troubleshooting your agent



#### Note

We recommend changing the default password. To do this, on the console main menu enter 5 (or 6 for VMware VMs), then run the passwd command to change the password.

### Accessing the local console (Amazon EC2)

To connect to an Amazon EC2 agent's local console, you must use SSH.

Before you begin: Make sure that your EC2 instance's security group allows access with SSH (TCP port 22).

Open a terminal and copy the following ssh command:

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-ip-address
```

- For /path/key-pair-name, specify the path and file name (.pem) of the private key required to connect to your instance.
- For *instance-user-name*, specify admin.
- For *instance-public-ip-address*, specify the public IP address of your instance.
- Run the ssh command to connect to the instance.

Once connected, the main menu of the agent's local console displays.

### Configuring your agent's DHCP and DNS settings

The default network configuration for the agent is Dynamic Host Configuration Protocol (DHCP). With DHCP, your agent is automatically assigned an IP address. In some cases, you might need to manually assign your agent's IP as a static IP address, as described following.

- Log in to your agent's local console.
- On the AWS DataSync Activation Configuration main menu, enter 1 to begin configuring your network.
- On the **Network Configuration** menu, choose one of the following options.

То	Do this
Get information about your network adapter	Enter 1.  A list of adapter names appears, and you are prompted to enter an adapter name—for example, eth0. If the adapter you specify is in use, the following information about the adapter is displayed:  Media access control (MAC) address  IP address  Netmask  Agent IP address  DHCP enabled status  You use the same adapter name when you configure a static IP address (option 3) as when you set your agent's default route adapter (option 5).
Configure DHCP	Enter 2.  You are prompted to configure the network interface to use DHCP.

То	Do this
Configure a static IP address for your agent	Enter <b>3</b> .  You are prompted to enter the Network adapter name.
Reset all your agent's network configuration to DHCP	Enter <b>4</b> .  All network interfaces are set to use DHCP.
Set your agent's default route adapter	Enter <b>5</b> .  The available adapters for your agent are shown, and you are prompted to choose one of the adapters—for example, <b>eth0</b> .

То	Do this
Edit your agent's Domain Name System (DNS) configuration	Enter <b>6</b> .  The available adapters of the primary and secondary DNS servers are displayed. You are prompted to provide the new IP address.
View your agent's DNS configuration	Enter 7.  The available adapters of the primary and secondary DNS servers are displayed.  (i) Note  For some versions of the VMware hypervisor, you can edit the adapter configuration in this menu.
View routing tables	Enter 8.  The default route of your agent is displayed.

### Checking your agent's system resources

When you log in to your agent console, virtual CPU cores, root volume size, and RAM are automatically checked. If there are any errors or warnings, they're flagged on the console menu display with a banner that provides details about those errors or warnings.

If there are no errors or warnings when the console starts, the menu displays white text. The **View System Resource Check** option will display (0 Errors).

If there are errors or warnings, the console menu displays the number of errors and warnings, in red and yellow respectively, in a banner across the top of the menu. For example, (1 ERROR, 1 WARNING).

#### To check your agent's system resources

- 1. Log in to your agent's local console.
- On the AWS DataSync Activation Configuration main menu, enter 4 to view the results of the system resource check.

The console displays an **[OK]**, **[WARNING]**, or **[FAIL]** message for each resource as described in the table following.

For Amazon EC2 instances, the system resource check verifies that the instance type is one of the instances recommended for use with DataSync. If the instance type matches that list, a single result is displayed in green text, as follows.

[ OK ] Instance Type Check

If the Amazon EC2 instance is not on the recommended list, the system resource check verifies the following resources.

- CPU cores check: At least four cores are required.
- Disk size check: A minimum of 80 GB of available disk space is required.
- · RAM check:
  - 32 GB of RAM assigned to the instance for task executions working with up to 20 million files, objects, or directories.
  - 64 GB of RAM assigned to the instance for task executions working with more than 20 million files, objects, or directories.
- CPU flags check: The agent VM CPU must have either SSSE3 or SSE4 instruction set flags.

If the Amazon EC2 instance is not on the list of recommended instances for DataSync, but it has sufficient resources, the result of the system resource check displays four results, all in green text.

The same resources are verified for agents deployed in Hyper-V, Linux Kernel-based Virtual Machine (KVM), and VMware VMs.

VMware agents are also checked for supported version; unsupported versions cause a red banner error. Supported versions include VMware versions 6.5 and 6.7.

# Synchronizing the time on your VMware agent

If you're using a <u>VMware agent</u>, you can view or edit your Network Time Protocol (NTP) server configuration and synchronize the agent's time with your VMware hypervisor host.

- 1. Log in to your agent's local console.
- 2. On the AWS DataSync Activation Configuration main menu, enter 5.
- 3. On the **System Time Management** menu, do one of the following:

То	Do this
View and synchronize your VM time with NTP server time	Enter 1.
	The current time of your agent displays. Your agent determines the time difference betwee n your agent and NTP server, and prompts you to synchronize the times.
	In some situations, an agent's time might drift. For example, there might be a prolon ged network outage and your hypervisor host and agent don't get time updates, so your agent's time is different from the actual time. When there's a time drift like this, a discre pancy occurs between the stated times when operations (such as snapshots occur) and the actual times that the operations occur.
Edit your NTP server configuration	Enter 2.
	You're prompted to provide an NTP server configuration.
View your NTP server configuration	Enter 3.
	Your NTP server configuration displays.

# Running maintenance-related commands for your agent

In your DataSync agent's local console, you can perform some maintenance tasks and diagnose issues with your agent.

### To run a configuration or diagnostic command in your agent's local console

- 1. Log in to your agent's local console.
- 2. On the **AWS DataSync Activation Configuration** main menu, enter **5** (or for **6** a VMware VM) for the **Command Prompt**.
- 3. Use the following commands to perform the following tasks with your agent.

Command	Description
dig	Look up DNS information about the host.
diskclean	Perform disk cleanup.
exit	Return to the console configuration menu.
h	Display a list of available commands.
ifconfig	Display or configure network interfaces.
ip	Display or configure routing, devices, and tunnels.
iptables	Set up and maintain IPv4 packet filtering and network address translation (NAT).
ncport	Test connectivity to a specific network TCP port.
nping	Get information to troubleshoot network issues.
save-iptables	Save IP table firewall rules permanently.
save-routing-table	Save a newly added routing table entry.

Command	Description
sslcheck	Verify whether an SSL certificate is valid.
tcptraceroute	Collect traceroute output on TCP traffic to a destination.

4. Follow the onscreen instructions.

# Replacing your AWS DataSync agent

To replace an AWS DataSync agent, you must create a new agent and update any transfer locations that are using the old agent.

# Creating a new agent

To create your new DataSync agent, follow the same process when you created your old agent:

- 1. Deploy an agent in your storage environment.
- 2. Choose a service endpoint that the agent uses to communicate with AWS.
- 3. Configure your network so that the agent can communicate with your storage and AWS.
- 4. Activate your agent.
- 5. Once activated, make note of the agent's Amazon Resource Name (ARN).

You need this ARN when updating your DataSync location to use the new agent.

# Updating your location with the new agent

Once you create a new agent, you can update an existing DataSync location to use this agent. In most cases, you also have to re-enter access credentials to update the location. This is because DataSync stores location credentials in a way that only your agent can use them.

### Using the DataSync console

The following instructions describe how to update locations with a new agent by using the DataSync console.

Replacing your agent 325

#### NFS

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose the location that you want to update, then choose **Edit**.
- 4. For **Agents**, choose your new agent.

You can choose more than one agent if you're replacing multiple agents for a location.

5. Choose **Save changes**.

#### **SMB**

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose the location that you want to update, then choose **Edit**.
- 4. For **Agents**, choose your new agent.

You can choose more than one agent if you're replacing multiple agents for a location.

- 5. For **Password**, enter the password of the user that can mount your SMB file server and has permission to access the files and folders involved in your transfer.
- 6. Choose **Save changes**.

#### **HDFS**

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose the location that you want to update, then choose **Edit**.
- 4. For **Agents**, choose your new agent.

You can choose more than one agent if you're replacing multiple agents for a location.

- 5. If you're using Kerberos authentication, upload your **Keytab file** and **Kerberos configuration file**.
- 6. Choose **Save changes**.

### Object storage

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose the location that you want to update, then choose **Edit**.
- 4. For **Agents**, choose your new agent.

You can choose more than one agent if you're replacing multiple agents for a location.

- 5. If your location requires credentials, enter the **Secret key** that allows DataSync to access your object storage bucket.
- 6. Choose **Save changes**.

### Azure Blob Storage

Do the following to update your Microsoft Azure Blob Storage location:

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose the location that you want to update, then choose **Edit**.
- 4. For **Agents**, choose your new agent.

You can choose more than one agent if you're replacing multiple agents for a location.

- 5. For **SAS token**, enter the <u>shared access signature (SAS) token</u> that allows DataSync to access your blob storage.
- 6. Choose **Save changes**.

### Using the AWS CLI

The following instructions describe how to update locations with a new agent by using the AWS CLI. (You can also do this by using the DataSync API.)

#### **NFS**

1. Copy the following <u>update-location-nfs</u> command:

aws datasync update-location-nfs \

```
--location-arn datasync-nfs-location-arn \
--on-prem-config AgentArns=new-datasync-agent-arn
```

- 2. For the --location-arn parameter, specify the ARN of the NFS location that you're updating.
- 3. For the --on-prem-config parameter's AgentArns option, specify the ARN of your new agent.

You can specify more than one ARN if you're replacing multiple agents for a location.

4. Run the update-location-nfs command to update the location.

#### **SMB**

1. Copy the following update-location-smb command:

```
aws datasync update-location-smb \
    --location-arn datasync-smb-location-arn \
    --agent-arns new-datasync-agent-arn \
    --password smb-file-server-password
```

- 2. For the --location-arn parameter, specify the ARN of the SMB location that you're updating.
- 3. For the --agent-arns parameter, specify the ARN of your new agent.

You can specify more than one ARN if you're replacing multiple agents for a location.

- 4. For the --password parameter, specify the password of the user that can mount your SMB file server and has permission to access the files and folders involved in your transfer.
- 5. Run the update-location-smb command to update the location.

#### **HDFS**

1. Copy the following <u>update-location-hdfs</u> command:

```
aws datasync update-location-hdfs \
    --location-arn datasync-hdfs-location-arn \
    --agent-arns new-datasync-agent-arn \
    --kerberos-keytab keytab-file \
    --kerberos-krb5-conf krb5-conf-file
```

2. For the --location-arn parameter, specify the ARN of the HDFS location that you're updating.

3. For the --agent-arns parameter, specify the ARN of your new agent.

You can specify more than one ARN if you're replacing multiple agents for a location.

- 4. If you're using Kerberos authentication, include the --kerberos-keytab and -- kerberos-krb5-conf parameters:
  - For the --kerberos-keytab parameter, specify the Kerberos key table (keytab) that contains mappings between the defined Kerberos principal and encrypted keys.

You can specify the keytab file by providing the file's address.

• For the --kerberos-krb5-conf parameter, specify the file that contains the configuration for your Kerberos realm.

You can specify the krb5. conf file by providing the file's address.

If you're using simple authentication, you don't need to include these Kerberos-related parameters in your command.

5. Run the update-location-hdfs command to update the location.

#### Object storage

1. Copy the following <u>update-location-object-storage</u> command:

```
aws datasync update-location-object-storage \
    --location-arn datasync-object-storage-location-arn \
    --agent-arns new-datasync-agent-arn \
    --secret-key bucket-secret-key
```

- For the --location-arn parameter, specify the ARN of the object storage location that you're updating.
- 3. For the --agent-arns parameter, specify the ARN of your new agent.

You can specify more than one ARN if you're replacing multiple agents for a location.

4. Do the following depending on if your object storage location requires access credentials:

• If your location requires credentials – For the --secret-key parameter, specify the secret key that allows DataSync to access your object storage bucket.

• If your location requires credentials – Specify empty strings for the --access-key and --secret-key parameters. Here's an example command:

```
aws datasync update-location-object-storage \
    --location-arn arn:aws:datasync:us-east-2:111122223333:location/
loc-abcdef01234567890 \
    --agent-arns arn:aws:datasync:us-east-2:111122223333:agent/
agent-1234567890abcdef0 \
    --access-key "" \
    --secret-key ""
```

5. Run the update-location-object-storage command to update the location.

#### Azure Blob Storage

1. Copy the following update-location-azure-blob command:

```
aws datasync update-location-azure-blob \
    --location-arn datasync-azure-blob-storage-location-arn \
    --agent-arns new-datasync-agent-arn \
    --sas-configuration '{
        "Token": "sas-token-for-azure-blob-storage"
    }'
```

- For the --location-arn parameter, specify the ARN of the Azure Blob Storage location that you're updating.
- 3. For the --agent-arns parameter, specify the ARN of your new agent.

You can specify more than one ARN if you're replacing multiple agents for a location.

- 4. For the --sas-configuration parameter's Token option, specify the <u>SAS token</u> that allows DataSync to access your blob storage.
- 5. Run the update-location-azure-blob command to update the location.

### **Next steps**

Delete your old agent. If you have any running DataSync tasks using this agent, wait until 1. those tasks finish before deleting it.

- 2. If you need to replace agents for multiple locations, repeat the previous steps.
- When you're done, you can resume running your tasks.



### Note

**Replacing agents for scheduled tasks** – If you replace an agent for a scheduled task, you must start that task manually if the new agent is using a different type of service endpoint than your old agent. If you don't run the task manually before its next scheduled run, the task fails.

For example, if your old agent used a public service endpoint, but the new agent uses a VPC endpoint, start that task manually by using the console or StartTaskExecution operation. After that, your task will resume running on its schedule.

# Filtering AWS DataSync resources

You can filter your AWS DataSync locations and tasks by using the ListLocations and ListTasks API operations in the AWS CLI. For example, retrieve a list of your most recent tasks.

# **Parameters for filtering**

You can use API filters to narrow down the list of resources returned by ListTasks and ListLocations. For example, to retrieve all of your Amazon S3 locations, you can use ListLocations with the filter name LocationType S3 and Operator Equals.

To filter API results, you must specify a filter name, operator, and value.

- Name The name of the filter that's being used. Each API call supports a list of filters that are available for it (for example, LocationType for ListLocations).
- Values The values that you want to filter for. For example, you might want to display only Amazon S3 locations.
- Operator The operator that's used to compare filter values (for example, Equals or Contains).

Next steps 331

The following table lists the available operators.

Operator	Key types
Equals	String, Number
NotEquals	String, Number
LessThan	Number
LessThanOrEqual	Number
GreaterThan	Number
GreaterThanOrEqual	Number
In	String
Contains	String
NotContains	String
BeginsWith	String

# Filtering by location

ListLocations supports the following filter names:

- LocationType Filters on the location type:
  - SMB
  - NFS
  - HDFS
  - OBJECT\_STORAGE
  - S3
  - OUTPOST\_S3
  - FSX\_WINDOWS
  - FSX\_LUSTRE

Filtering by location 332

- FSX OPENZFS NFS
- FSX\_ONTAP\_NFS
- FSX ONTAP SMB
- LocationUri Filters on the uniform resource identifier (URI) assigned to the location, as returned by the DescribeLocation\* API call (for example, s3://bucket-name/yourprefix for Amazon S3 locations).
- CreationTime Filters on the time that the location was created. The input format is yyyy-MM-dd:mm:ss in Coordinated Universal Time (UTC).

The following AWS CLI example lists all locations of type Amazon S3 that have a location URI starting with the string "s3://amzn-s3-demo-bucket" and that were created at or after 2019-12-15 17:15:20 UTC.

```
aws datasync list-locations \
    --filters [{Name=LocationType, Values=["S3"], Operator=Equals},
    {Name=LocationUri, Values=["s3://amzn-s3-demo-bucket"], Operator=BeginsWith},
    {Name=CreationTime, Values=["2019-12-15 17:15:20"], Operator=GreaterThanOrEqual}]
```

This command returns output similar to the following.

### Filtering by task

ListTasks supports the following filter names.

Filtering by task 333

 LocationId – Filters on both source and destination locations on Amazon Resource Name (ARN) values.

 CreationTime – Filters on the time that the task was created. The input format is yyyy-MMdd:mm:ss in UTC.

The following AWS CLI example shows the syntax when filtering on LocationId.

```
aws datasync list-tasks \
    --filters Name=LocationId, Values=arn:aws:datasync:us-east-1:your-account-
id:location/your-location-id,Operator=Contains
```

The output of this command looks similar to the following.

```
{
    "Tasks": [
        {
            "TaskArn": "arn:aws:datasync:us-east-1:your-account-id:task/your-task-id",
            "Status": "AVAILABLE",
            "Name": "amzn-s3-demo-bucket"
        }
    ]
}
```

# Cleaning up your AWS DataSync resources

If you used AWS DataSync for a test or don't need the AWS resources that you created, delete them so that you aren't charged for resources you don't plan to use.



### Note

If you have DataSync resources in an opt-in Region that you disable, those resources aren't automatically deleted. The resources are still there if you enable that Region again.

# **Deleting a DataSync agent**

When you delete an agent from AWS DataSync, the agent resource is no longer associated with your AWS account and can't be undone.

Keep in mind that deleting an agent from DataSync doesn't remove its virtual machine (VM) or Amazon EC2 instance from your storage environment. You can delete the VM or instance or reuse it to activate a new agent.

### **Prerequisites**

Don't delete an agent until you update or remove the DataSync resources that depend on it. If you're replacing an agent, <u>update your transfer locations</u> with the new agent. If you aren't replacing an agent, delete transfer tasks and locations using that agent first.

### **Deleting the agent**

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, choose **Agents**.
- 3. Choose the agent that you want to delete.
- 4. Choose **Delete**, enter **delete** in the text box that appears, and then choose **Delete**.
- 5. If you aren't planning to <u>reuse the agent's infrastructure</u> for other DataSync activities, delete the agent's VM or Amazon EC2 instance to remove it from your storage environment.

# Reusing a DataSync agent's infrastructure

You can delete an agent resource from DataSync and still use the agent's underlying VM or Amazon EC2 instance to activate a new agent.

### To reuse an agent's infrastructure

- 1. <u>Test the agent's connection to AWS</u>. If the network tests pass, go to the next step.
  - The network tests must pass before you can move to the next step.
- 2. <u>Delete the agent</u> resource from DataSync but don't delete the agent's VM or Amazon EC2 instance.
- 3. Repeat step 1 to test the agent's connection to AWS again. If the network tests pass, go to the next step.
- 4. About three minutes after deleting the agent resource from DataSync, check if port 80 is open on the agent VM or Amazon EC2 instance. If it is, go to the next step.
- 5. <u>Activate a new agent</u> with the existing VM or Amazon EC2 instance.

You can activate the new agent in a different AWS Region, AWS account, and with another type of <u>service endpoint</u>. If you use a different type of service endpoint, you have to adjust your <u>network configuration</u>.

### **Deleting a DataSync location**

As a best practice, remove the AWS DataSync locations that you no longer need.

### To remove a location by using the DataSync console

- 1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Locations**.
- 3. Choose the location that you want to remove.
- 4. Choose **Delete**. Confirm the deletion by entering **delete**, and then choose **Delete**.

# **Deleting a DataSync task**

If you no longer need an AWS DataSync task, you can delete it and its related AWS resources.

# **Prerequisites**

When you run a task, DataSync automatically creates and manages <u>network interfaces</u> for data transfer traffic. When you delete a task, you also delete its related network interfaces as long as you have the following permissions:

- ec2:DeleteNetworkInterface
- ec2:DescribeNetworkInterfaces
- ec2:ModifyNetworkInterfaceAttribute

These permissions are available in the AWS managed policy AWSDataSyncFullAccess. For more information, see AWS managed policies for AWS DataSync.

### **Deleting the task**

Once you delete a task, you can't restore it.

### Using the DataSync console

- 1. Open the AWS DataSync console at <a href="https://console.aws.amazon.com/datasync/">https://console.aws.amazon.com/datasync/</a>.
- 2. In the left navigation pane, expand **Data transfer**, then choose **Tasks**.
- 3. Select the task that you want to delete.
- 4. Choose **Actions**, then choose **Delete**.
- 5. In the dialog box, choose **Delete**.

### Using the AWS CLI

1. Copy the following delete-task command:

```
aws datasync delete-task \
--task-arn "task-to-delete"
```

- 2. For the --task-arn parameter, specify the Amazon Resource Name (ARN) of the task you're deleting (for example, arn:aws:datasync:us-east-2:123456789012:task/task-012345678abcd0123).
- Run the delete-task command.

Deleting a DataSync task 337

# **Security in AWS DataSync**

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The <u>shared responsibility model</u> describes this as security *of* the cloud and security *in* the cloud:

- Security of the cloud AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the <u>AWS</u> compliance programs. To learn about the compliance programs that apply to AWS DataSync, see AWS services in scope by compliance program.
- **Security in the cloud** Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using DataSync. The following topics show you how to configure DataSync to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your DataSync resources.

#### **Topics**

- Data protection in AWS DataSync
- Identity and access management in AWS DataSync
- Compliance validation for AWS DataSync
- Resilience in AWS DataSync
- Infrastructure security in AWS DataSync
- Securing storage location credentials

# **Data protection in AWS DataSync**

AWS DataSync securely transfers data between self-managed storage systems and AWS storage services and also between AWS storage services. How your storage data is encrypted in transit depends in part on the locations involved in the transfer.

After the transfer completes, data is encrypted at rest by the system or service that's storing the data (not DataSync).

### **Topics**

- AWS DataSync encryption in transit
- AWS DataSync encryption at rest
- Internetwork traffic privacy

# **AWS DataSync encryption in transit**

Your storage data (including metadata) is encrypted in transit, but how it's encrypted throughout the transfer depends on your source and destination locations.

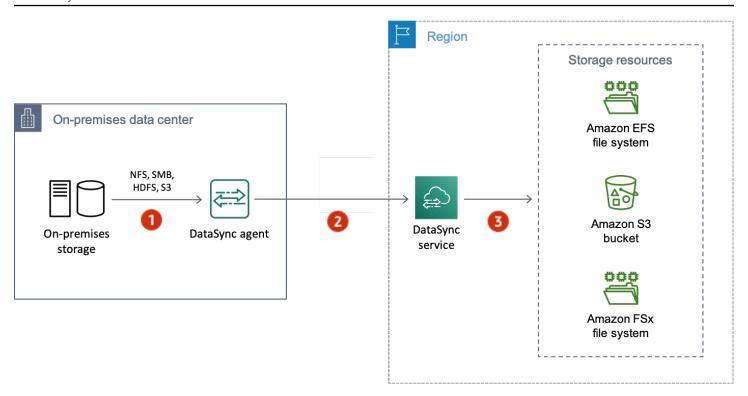
When connecting with a location, DataSync uses the most secure options provided by that location's data access protocol. For example, when connecting with a file system using Server Message Block (SMB), DataSync uses the security features provided by SMB.

#### Network connections in a transfer

DataSync requires three network connections to copy data: a connection to read data from a source location, another to transfer data between locations, and one more to write data to a destination location.

The following diagram is an example of the network connections that DataSync uses to transfer data from an on-premises storage system to an AWS storage service. To understand where the connections happen and how data is protected as it transfers through each connection, use the accompanying table.

Data protection 339



Reference	Network connection	Description
1	Reading data from the source location	DataSync connects by using the storage system's protocol for accessing data (for example, SMB or the Amazon S3 API). For this connection, data is protected by using the security features of the storage system unless DataSync doesn't support those features. For example, DataSync currently doesn't support Kerberos authentication with NFS file servers or when using TDE encryption with HDFS.
2	Transferring data between locations	For this connection, DataSync encrypts all network traffic with mutual Transport Layer Security (mTLS) 1.3.

Encryption in transit 340

Reference	Network connection	Description
3	Writing data to the destination location	As with the source location, DataSync connects by using the storage system's protocol for accessing data. Data is again protected by using the security features of the storage system unless DataSync doesn't support those features.

Learn how your data is encrypted in transit when DataSync connects to the following AWS storage services:

- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP
- Amazon S3

### **TLS ciphers**

When transferring data between locations, DataSync uses different TLS ciphers. The TLS cipher depends on the type of service endpoint that your agent uses to communicate with DataSync. (For more information, see Choosing a service endpoint for your AWS DataSync agent.)

#### **Contents**

- Public or VPC endpoints
- FIPS endpoints

### **Public or VPC endpoints**

For public and virtual private cloud (VPC) service endpoints, DataSync uses one of the following TLS ciphers:

Encryption in transit 341

- TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384 (ecdh\_x25519)
- TLS\_ECDHE\_RSA\_WITH\_CHACHA20\_POLY1305\_SHA256 (ecdh\_x25519)

TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256 (ecdh\_x25519)

#### **FIPS endpoints**

For Federal Information Processing Standard (FIPS) service endpoints, DataSync uses the following TLS cipher:

TLS AES 128 GCM SHA256 (secp256r1)

### AWS DataSync encryption at rest

Because AWS DataSync is a transfer service, it generally doesn't manage your storage data at rest. The storage services and systems that DataSync supports are responsible for protecting data in that state. However, there is some service-related data that DataSync manages at rest.

### What's encrypted?

The only data that DataSync handles at rest relates to the details it needs to complete your transfer. DataSync stores the following data with full at-rest encryption in Amazon DynamoDB:

- Task configurations (for example, details about the locations in your transfer).
- User credentials that allow your DataSync agent to authenticate with a location. These credentials are encrypted by using your agent's public keys. The agent can decrypt these keys as needed with its private keys.

For more information, see <u>DynamoDB encryption at rest</u> in the *Amazon DynamoDB Developer Guide*.

### Key management

You can't manage the encryption keys that DataSync uses to store information in DynamoDB related to running your task. This information includes your task configurations and the credentials that agents use to authenticate with a storage location.

Encryption at rest 342

### What's not encrypted?

Though DataSync doesn't control how your storage data is encrypted at rest, we still recommend configuring your locations with the highest level of security that they support. For example, you can encrypt objects with Amazon S3 managed encryption keys (SSE-S3) or AWS Key Management Service (AWS KMS) keys (SSE-KMS).

Learn more about how AWS storage services encrypt data at rest:

- Amazon S3
- Amazon EFS
- Amazon FSx for Windows File Server
- Amazon FSx for Lustre
- Amazon FSx for OpenZFS
- Amazon FSx for NetApp ONTAP

### Internetwork traffic privacy

We recommend configuring your source and destination locations with the highest level of security that each one supports. When connecting to a location, AWS DataSync works with the most secure version of the data access protocol that the storage system uses. Additionally, consider limiting subnet traffic to known protocols and services.

DataSync secures the connection between locations—including between AWS accounts, AWS Regions, and Availability Zones—by using Transport Layer Security (TLS) 1.3.

# Identity and access management in AWS DataSync

AWS uses security credentials to identify you and to grant you access to your AWS resources. You can use features of AWS Identity and Access Management (IAM) to allow other users, services, and applications to use your AWS resources fully or in a limited way, without sharing your security credentials.

By default, IAM identities (users, groups, and roles) don't have permission to create, view, or modify AWS resources. To allow users, groups, and roles to access AWS DataSync resources and interact with the DataSync console and API, we recommend that you use an IAM policy that grants them permission to use the specific resources and API actions that they will need. You then attach the

Internetwork traffic privacy 343

policy to the IAM identity that requires access. For an overview of the basic elements for a policy, see Access management for AWS DataSync.

#### **Topics**

- Access management for AWS DataSync
- AWS managed policies for AWS DataSync
- IAM customer managed policies for AWS DataSync
- Using service-linked roles for DataSync
- Permissions for tagging DataSync resources during creation
- Cross-service confused deputy prevention

### Access management for AWS DataSync

Every AWS resource is owned by an AWS account. Permissions to create or access a resource are governed by permissions policies. An account administrator can attach permissions policies to AWS Identity and Access Management (IAM) identities. Some services (such as AWS Lambda) also support attaching permissions policies to resources.



### Note

An account administrator is a user with administrator privileges in an AWS account. For more information, see IAM best practices in the IAM User Guide.

### **Topics**

- DataSync resources and operations
- Understanding resource ownership
- Managing access to resources
- Specifying policy elements: Actions, effects, resources, and principals
- Specifying conditions in a policy

### **DataSync resources and operations**

In DataSync, the primary resources are agent, location, task, and task execution.

344 Access management

These resources have unique Amazon Resource Names (ARNs) associated with them, as shown in the following table.

Resource type	ARN format
Agent ARN	arn:aws:datasync: region:account-id :agent/agent-id
Location ARN	arn:aws:datasync: region:account-id :location/ location-id
Task ARN	arn:aws:datasync: region:account-id :task/task-id
Task execution ARN	<pre>arn:aws:datasync: region:account-id :task/task-id/executio n/ exec-id</pre>

To grant permissions for specific API operations, such as creating a task, DataSync defines a set of actions that you can specify in a permissions policy. An API operation can require permissions for more than one action.

# **Understanding resource ownership**

A resource owner is the AWS account that created the resource. That is, the resource owner is the AWS account of the principal entity (for example, an IAM role) which authenticates the request that creates the resource. The following examples illustrate how this behavior works:

- If you use the root account credentials of your AWS account to create a task, your AWS account is the owner of the resource (in DataSync, the resource is the task).
- If you create an IAM roles in your AWS account and grant permissions to the CreateTask action to that user, the user can create a task. However, your AWS account, to which the user belongs, owns the task resource.
- If you create an IAM role in your AWS account with permissions to create a task, anyone who can assume the role can create a task. Your AWS account, to which the role belongs, owns the task resource.

Access management 345

### Managing access to resources

A permissions policy describes who has access to what. The following section explains the available options for creating permissions policies.



### Note

This section discusses using IAM in the context of DataSync. It doesn't provide detailed information about the IAM service. For complete IAM documentation, see What is IAM? in the IAM User Guide. For information about IAM policy syntax and descriptions, see AWS Identity and Access Management policy reference in the IAM User Guide.

Policies attached to an IAM identity are referred to as identity-based policies (IAM policies) and policies attached to a resource are referred to as resource-based policies. DataSync supports only identity-based policies (IAM policies).

### **Topics**

- Identity-based policies
- Resource-based policies

### **Identity-based policies**

You can manage DataSync resource access with IAM policies. These policies can help an AWS account administrator do the following with DataSync:

- Grant permissions to create and manage DataSync resources Create an IAM policy that allows an IAM role in your AWS account to create and manage DataSync resources, such as agents, locations, and tasks.
- Grant permissions to a role in another AWS account or an AWS service Create an IAM policy that grants permissions to an IAM role in a different AWS account or an AWS service. For example:
  - 1. The Account A administrator creates an IAM role and attaches a permissions policy to the role that grants permissions on resources in Account A.
  - 2. The Account A administrator attaches a trust policy to the role that identifies Account B as the principal who can assume the role.

346 Access management

To grant an AWS service permissions to assume the role, the Account A administrator can specify an AWS service as the principal in the trust policy.

3. The Account B administrator can then delegate permissions to assume the role to any users in Account B. This allows anyone using the role in Account B to create or access resources in Account A.

For more information about using IAM to delegate permissions, see <u>Access management</u> in the *IAM User Guide*.

The following example policy grants permissions to all List\* actions on all resources. This action is a read-only action and doesn't allow resource modification.

**JSON** 

For more information about using identity-based policies with DataSync, see <u>AWS managed</u> <u>policies</u> and <u>customer managed policies</u>. For more information about IAM identities, see the <u>IAM</u> <u>User Guide</u>.

#### **Resource-based policies**

Other services, such as Amazon S3, support resource-based permissions policies. For example, you can attach a policy to an Amazon S3 bucket to manage access permissions to that bucket. However, DataSync doesn't support resource-based policies.

Access management 347

### Specifying policy elements: Actions, effects, resources, and principals

For each DataSync resource, the service defines a set of API operations (see <u>Actions</u>). To grant permissions for these API operations, DataSync defines a set of actions that you can specify in a policy. For example, for the DataSync resource, the following actions are defined: CreateTask, DeleteTask, and DescribeTask. Performing an API operation can require permissions for more than one action.

The following are the most basic policy elements:

- Resource In a policy, you use an Amazon Resource Name (ARN) to identify the resource to
  which the policy applies. For DataSync resources, you can use the wildcard character (\*) in IAM
  policies. For more information, see DataSync resources and operations.
- Action You use action keywords to identify resource operations that you want to allow or deny. For example, depending on the specified Effect element, the datasync:CreateTask permission allows or denies the user permissions to perform the DataSync CreateTask operation.
- Effect You specify the effect when the user requests the specific action—this effect can be either Allow or Deny. If you don't explicitly grant access to (Allow) a resource, access is implicitly denied. You can also explicitly deny access to a resource, which you might do to make sure that a user cannot access it, even if a different policy grants that user access. For more information, see Authorization in the IAM User Guide.
- **Principal** In identity-based policies (IAM policies), the user that the policy is attached to is the implicit principal. For resource-based policies, you specify the user, account, service, or other entity that you want to receive permissions (applies to resource-based policies only). DataSync doesn't support resource-based policies.

To learn more about IAM policy syntax and descriptions, see <u>AWS Identity and Access Management</u> policy reference in the *IAM User Guide*.

### Specifying conditions in a policy

When you grant permissions, you can use the IAM policy language to specify the conditions when a policy should take effect when granting permissions. For example, you might want a policy to be applied only after a specific date. For more information about specifying conditions in policy language, see Condition in the IAM User Guide.

Access management 348

To express conditions, you use predefined condition keys. There are no condition keys specific to DataSync. However, there are AWS wide condition keys that you can use as appropriate. For a complete list of AWS wide keys, see Available keys in the *IAM User Guide*.

### **AWS managed policies for AWS DataSync**

To add permissions to users, groups, and roles, it's easier to use AWS managed policies than to write policies yourself. It takes time and expertise to <u>create IAM customer managed policies</u> that provide your team with only the permissions they need. To get started quickly, you can use our AWS managed policies. These policies cover common use cases and are available in your AWS account. For more information about AWS managed policies, see <u>AWS managed policies</u> in the *IAM User Guide*.

AWS services maintain and update AWS managed policies. You can't change the permissions in AWS managed policies. Services occasionally add additional permissions to an AWS managed policy to support new features. This type of update affects all identities (users, groups, and roles) where the policy is attached. Services are most likely to update an AWS managed policy when a new feature is launched or when new operations become available. Services do not remove permissions from an AWS managed policy, so policy updates won't break your existing permissions.

Additionally, AWS supports managed policies for job functions that span multiple services. For example, the ReadOnlyAccess AWS managed policy provides read-only access to all AWS services and resources. When a service launches a new feature, AWS adds read-only permissions for new operations and resources. For a list and descriptions of job function policies, see <a href="AWS managed policies for job functions">AWS managed policies for job functions</a> in the IAM User Guide.

### AWS managed policy: AWSDataSyncReadOnlyAccess

You can attach the AWSDataSyncReadOnlyAccess policy to your IAM identities.

This policy grants read-only permissions for DataSync.

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "DataSyncReadOnlyAccessPermissions",
        "Effect": "Allow",
        "Action": [
            "datasync:Describe*",
            "datasync:List*",
            "ec2:DescribeSecurityGroups",
            "ec2:DescribeSubnets",
            "elasticfilesystem:DescribeFileSystems",
            "elasticfilesystem:DescribeMountTargets",
            "fsx:DescribeFileSystems",
            "iam:GetRole",
            "iam:ListRoles",
            "logs:DescribeLogGroups",
            "logs:DescribeResourcePolicies",
            "s3:ListAllMyBuckets",
            "s3:ListBucket"
        ],
        "Resource": "*"
    }]
}
```

### AWS managed policy: AWSDataSyncFullAccess

You can attach the AWSDataSyncFullAccess policy to your IAM identities.

This policy grants administrative permissions for DataSync and is required for AWS Management Console access to the service. AWSDataSyncFullAccess provides full access to DataSync API operations and the operations that interact with related resources (such as Amazon S3 buckets, Amazon EFS file systems, AWS KMS keys, and Secrets Manager secrets). The policy also grants permissions for Amazon CloudWatch, including creating log groups and creating or updating a resource policy.

**JSON** 

```
{
```

```
"Version": "2012-10-17",
"Statement": [{
        "Sid": "DataSyncFullAccessPermissions",
        "Effect": "Allow",
        "Action": [
            "datasync:*",
            "ec2:CreateNetworkInterface",
            "ec2:CreateNetworkInterfacePermission",
            "ec2:DeleteNetworkInterface",
            "ec2:DescribeNetworkInterfaces",
            "ec2:DescribeRegions",
            "ec2:DescribeSecurityGroups",
            "ec2:DescribeSubnets",
            "ec2:DescribeVpcEndpoints",
            "ec2:ModifyNetworkInterfaceAttribute",
            "fsx:DescribeFileSystems",
            "fsx:DescribeStorageVirtualMachines",
            "elasticfilesystem:DescribeAccessPoints",
            "elasticfilesystem:DescribeFileSystems",
            "elasticfilesystem:DescribeMountTargets",
            "iam:GetRole",
            "iam:ListRoles",
            "logs:CreateLogGroup",
            "logs:DescribeLogGroups",
            "logs:DescribeResourcePolicies",
            "outposts:ListOutposts",
            "s3:GetBucketLocation",
            "s3:ListAllMyBuckets",
            "s3:ListBucket",
            "s3:ListBucketVersions",
            "s3-outposts:ListAccessPoints",
            "s3-outposts:ListRegionalBuckets",
            "secretsmanager:ListSecrets",
            "kms:ListAliases",
            "kms:DescribeKey"
        ],
        "Resource": "*"
   },
    {
        "Sid": "DataSyncPassRolePermissions",
        "Effect": "Allow",
        "Action": [
            "iam:PassRole"
        ],
```

```
"Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "datasync.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Sid": "DataSyncCreateSLRPermissions",
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::*:role/aws-service-role/
datasync.amazonaws.com/AWSServiceRoleForDataSync",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": "datasync.amazonaws.com"
                }
            }
        },
        {
            "Sid": "DataSyncSecretsManagerCreateAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:CreateSecret"
            ],
            "Resource": [
                "arn:*:secretsmanager:*:*:secret:aws-datasync!*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "${aws:PrincipalAccount}"
                }
            }
        },
        {
            "Sid": "DataSyncSecretsManagerAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:DeleteSecret",
```

```
"secretsmanager:UpdateSecret",
                "secretsmanager:PutSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:*:*:secret:aws-datasync!*"
            ],
            "Condition": {
                "StringEquals": {
                    "secretsmanager:ResourceTag/
aws:secretsmanager:owningService": "aws-datasync",
                    "aws:ResourceAccount": "${aws:PrincipalAccount}"
                }
            }
         }
    ]
}
```

### AWS managed policy: AWSDataSyncServiceRolePolicy

You can't attach the AWSDataSyncServiceRolePolicy policy to your IAM identities. This policy is attached to a service-linked role that allows DataSync to perform actions on your behalf. For more information, see Using service-linked roles for DataSync.

This policy grants administrative permissions that allow the service-linked role to create Amazon CloudWatch logs for DataSync tasks using Enhanced mode.

### **Policy updates**

Change	Description	Date
<u>AWSDataSyncFullAccess</u> – Change	DataSync modified permissio n statements for AWSDataSy ncFullAccess :	May 13, 2025
	The updated statements remove tagging condition s from the permissions	

Change	Description	Date
	DataSync uses to create Secrets Manager secrets.	
AWSDataSyncFullAccess – Change	DataSync added new permissions to AWSDataSy ncFullAccess : • secretsmanager:Cre	May 7, 2025
	<ul><li>ateSecret</li><li>secretsmanager:Put</li><li>SecretValue</li><li>secretsmanager:Del</li><li>eteSecret</li></ul>	
	<ul> <li>secretsmanager:Upd ateSecret</li> </ul>	
	These permissions let DataSync create, edit, and delete AWS Secrets Manager secrets.	

Change	Description	Date
AWSDataSyncFullAccess – Change	DataSync added new permissions to AWSDataSy ncFullAccess :  • secretsmanager:Lis tSecrets • kms:ListAliases • kms:DescribeKey  These permissions let DataSync retrieve metadata about your AWS Secrets Manager secrets and AWS KMS keys, including any aliases associated with your keys.	April 23, 2025
AWSDataSyncService RolePolicy – Change	DataSync added new permissions to the AWSDataSyncService RolePolicy policy that's used by the DataSync service- linked role AWSServic eRoleForDataSync :  • secretsmanager:Des cribeSecret • secretsmanager:Get SecretValue  These permissions let DataSync read metadata and values for secrets managed by AWS Secrets Manager.	April 15, 2025

Change	Description	Date
AWSDataSyncService RolePolicy – New policy	DataSync added a policy that's used by the DataSync service-linked role AWSServiceRoleForD ataSync . This new managed policy automatically creates Amazon CloudWatch logs for your DataSync tasks that use Enhanced mode.	October 30, 2024
AWSDataSyncFullAccess – Change	DataSync added new a permission to AWSDataSy ncFullAccess :  • iam:CreateServiceL inkedRole  This permission lets DataSync create service-linked roles for you.	October 30, 2024
AWSDataSyncFullAccess – Change	DataSync added new a permission to AWSDataSy ncFullAccess:  • ec2:DescribeRegions  This permission lets you choose opt-in Regions when creating a DataSync task for transfers between AWS Regions.	July 22, 2024

Change	Description	Date
AWSDataSyncFullAccess – Change	DataSync added new a permission to AWSDataSy ncFullAccess :  • s3:ListBucketVersi ons  This permission lets you	February 16, 2024
	choose a specific version of your <u>DataSync manifest</u> .	

Change	Description	Date
<u>AWSDataSyncFullAccess</u> – Change	DataSync added new permissions to AWSDataSy ncFullAccess :	May 2, 2023
	<ul> <li>ec2:DescribeVpcEnd points</li> </ul>	
	<ul> <li>elasticfilesystem:</li> <li>DescribeAccessPoin</li> <li>ts</li> </ul>	
	<ul> <li>fsx:DescribeStorag</li> <li>eVirtualMachines</li> </ul>	
	<ul><li>outposts:ListOutpo sts</li></ul>	
	<ul><li>s3:GetBucketLocati on</li></ul>	
	<ul> <li>s3-outposts:ListAc cessPoints</li> </ul>	
	• s3-outposts:ListRe gionalBuckets	
	These permissions help you create DataSync agents and locations for Amazon EFS, Amazon FSx for NetApp ONTAP, Amazon S3, and S3 on Outposts.	
DataSync started tracking changes	DataSync started tracking changes for its AWS managed policies.	March 1, 2021

### IAM customer managed policies for AWS DataSync

In addition to AWS managed policies, you also can create your own identity-based policies for AWS DataSync and attach them to the AWS Identity and Access Management (IAM) identities that require those permissions. These are known as customer managed policies, which are standalone policies that you administer in your own AWS account.

#### Important

Before you begin, we recommend that you learn about the basic concepts and options for managing access to your DataSync resources. For more information, see Access management for AWS DataSync.

When creating a customer managed policy, you include statements about DataSync operations that can be used on certain AWS resources. The following example policy has two statements (note the Action and Resource elements in each statement):

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowsSpecifiedActionsOnAllTasks",
            "Effect": "Allow",
            "Action": [
                "datasync:DescribeTask",
            "Resource": "arn:aws:datasync:us-east-2:111222333444:task/*"
        },
        {
            "Sid": "ListAllTasks",
            "Effect": "Allow",
            "Action": [
                "datasync:ListTasks"
            ],
            "Resource": "*"
        },
}
```

The policy's statements do the following:

359 Customer managed policies

• The first statement grants permissions to perform the datasync: DescribeTask action on certain transfer task resources by specifying an Amazon Resource Name (ARN) with a wildcard character (\*).

• The second statement grants permissions to perform the datasync:ListTasks action on all tasks by specifying just a wildcard character (\*).

### **Examples of customer managed policies**

The following example customer managed policies grant permissions for various DataSync operations. The policies work if you're using the AWS Command Line Interface (AWS CLI) or an AWS SDK. To use these policies in the console, you must also use the managed policy AWSDataSyncFullAccess.

#### **Topics**

- Example 1: Create a trust relationship that allows DataSync to access your Amazon S3 bucket
- Example 2: Allow DataSync to read and write to your Amazon S3 bucket
- Example 3: Allow DataSync to upload logs to CloudWatch log groups

#### Example 1: Create a trust relationship that allows DataSync to access your Amazon S3 bucket

The following is an example of a trust policy that allows DataSync to assume an IAM role. This role allows DataSync to access an Amazon S3 bucket. To prevent the <a href="mailto:cross-service confused deputy">cross-service confused deputy</a> <a href="mailto:problem">problem</a>, we recommend using the <a href="mailto:aws:SourceArn">aws:SourceArn</a> and <a href="mailto:aws:SourceAccount">aws:SourceAccount</a> global condition context keys in the policy.

**JSON** 

Customer managed policies 360

#### Example 2: Allow DataSync to read and write to your Amazon S3 bucket

The following example policy grants DataSync the minimum permissions to read and write data to an S3 bucket that's used as a destination location.

#### Note

The value for aws: ResourceAccount should be the account ID that owns the Amazon S3 bucket specified in the policy.

```
{
 "Version": "2012-10-17",
 "Statement": [
     {
         "Action": [
             "s3:GetBucketLocation",
             "s3:ListBucket",
             "s3:ListBucketMultipartUploads"
         ],
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
         "Condition": {
             "StringEquals": {
                 "aws:ResourceAccount": "123456789012"
             }
         }
     },
     {
         "Action": [
```

Customer managed policies 361

```
"s3:AbortMultipartUpload",
             "s3:DeleteObject",
             "s3:GetObject",
             "s3:GetObjectTagging",
             "s3:GetObjectVersion",
             "s3:GetObjectVersionTagging",
             "s3:ListMultipartUploadParts",
             "s3:PutObject",
             "s3:PutObjectTagging"
           ],
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
         "Condition": {
             "StringEquals": {
                  "aws:ResourceAccount": "123456789012"
             }
         }
     }
]
}
```

#### Example 3: Allow DataSync to upload logs to CloudWatch log groups

DataSync requires permissions to be able to upload logs to your Amazon CloudWatch log groups. You can use CloudWatch log groups to monitor and debug your tasks.

For an example of an IAM policy that grants such permissions, see <u>Allowing DataSync to upload</u> logs to a CloudWatch log group.

### Using service-linked roles for DataSync

AWS DataSync uses AWS Identity and Access Management (IAM) <u>service-linked roles</u>. A service-linked role is a unique type of IAM role that is linked directly to DataSync. Service-linked roles are predefined by DataSync and include all the permissions that the service requires to call other AWS services on your behalf.

#### **Topics**

• Using roles for DataSync

### **Using roles for DataSync**

AWS DataSync uses AWS Identity and Access Management (IAM) <u>service-linked roles</u>. A service-linked role is a unique type of IAM role that is linked directly to DataSync. Service-linked roles are predefined by DataSync and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up DataSync easier because you don't have to manually add the necessary permissions. DataSync defines the permissions of its service-linked roles, and unless defined otherwise, only DataSync can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

You can delete a service-linked role only after first deleting their related resources. This protects your DataSync resources because you can't inadvertently remove permission to access the resources.

For information about other services that support service-linked roles, see <u>AWS services that work</u> with <u>IAM</u> and look for the services that have **Yes** in the **Service-linked roles** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

#### Service-linked role permissions for DataSync

DataSync uses the service-linked role named **AWSServiceRoleForDataSync** – Allows DataSync to perform essential operations for transfer task execution, including reading secrets from AWS Secrets Manager, and creating CloudWatch log groups and events.

The AWSServiceRoleForDataSync service-linked role trusts the following services to assume the role:

• datasync.amazonaws.com

The service-linked role uses the AWS managed policy named <u>AWSDataSyncServiceRolePolicy</u>, which allows DataSync to complete the following actions on the specified resources:

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [{
```

```
"Sid": "DataSyncCloudWatchLogCreateAccess",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream"
            ],
            "Resource": [
                "arn:*:logs:*:*:log-group:/aws/datasync*"
            1
        },
            "Sid": "DataSyncCloudWatchLogStreamUpdateAccess",
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:*:logs:*:*:log-group:/aws/datasync*:log-stream:*"
            ]
        },
            "Sid": "DataSyncSecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:DescribeSecret",
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:*:secretsmanager:*:*:secret:aws-datasync!*"
            ],
            "Condition": {
                "StringEquals": {
                    "secretsmanager:ResourceTag/
aws:secretsmanager:owningService": "aws-datasync",
                    "aws:ResourceAccount": "${aws:PrincipalAccount}"
                }
            }
        }
    ]
}
```

You must configure permissions to allow your users, groups, or roles to create, edit, or delete a service-linked role. For more information, see Service-linked role permissions in the *IAM User Guide*.

#### Creating a service-linked role for DataSync

You don't need to manually create a service-linked role. When you create a DataSync task in the AWS Management Console, the AWS CLI, or the AWS API, DataSync creates the service-linked role for you.

In the AWS CLI or the AWS API, you can create a service-linked role with the datasync.amazonaws.com service name. For more information, see <a href="Creating a service-linked">Creating a service-linked</a> role in the IAM User Guide. If you delete this service-linked role, you can use this same process to create the role again.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you create a DataSync task, DataSync creates the service-linked role for you again.

If you delete this service-linked role, you can use the same IAM process to create the role again.

#### Editing a service-linked role for DataSync

DataSync does not allow you to edit the AWSServiceRoleForDataSync service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a service-linked role in the IAM User Guide.

#### Deleting a service-linked role for DataSync

If you no longer need to use a feature or service that requires a service-linked role, we recommend that you delete that role. That way you don't have an unused entity that is not actively monitored or maintained. However, you must clean up your service-linked role before you can manually delete it.

#### Cleaning up a service-linked role

Before you can use IAM to delete a service-linked role, you must first delete any resources used by the role.



#### Note

If the DataSync service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

#### To delete DataSync resources used by the AWSServiceRoleForDataSync

- 1. Delete the DataSync agents used by the task (if there are any).
- 2. Delete the task's locations.
- 3. Delete the task.

#### Manually delete the service-linked role

Use the IAM console, the AWS CLI, or the AWS API to delete the AWSServiceRoleForDataSync service-linked role. For more information, see Deleting a service-linked role in the IAM User Guide.

#### Supported Regions for DataSync service-linked roles

DataSync supports using service-linked roles in all of the Regions where the service is available. For more information, see AWS Regions and endpoints.

### Permissions for tagging DataSync resources during creation

Some resource-creating AWS DataSync API actions enable you to specify tags when you create the resource. You can use resource tags to implement attribute-based access control (ABAC). For more information, see What is ABAC for AWS? in the IAM User Guide.

To enable users to tag resources on creation, they must have permissions to use the action that creates the resource (such as datasync:CreateAgent or datasync:CreateTask). If tags are specified in the resource-creating action, users must also have explicit permissions to use the datasync: TagResource action.

The datasync: TagResource action is only evaluated if tags are applied during the resourcecreating action. Therefore, a user that has permissions to create a resource (assuming there are no tagging conditions) doesn't require permissions to use the datasync: TagResource action if no tags are specified in the request.

However, if the user attempts to create a resource with tags, the request fails if the user doesn't have permissions to use the datasync: TagResource action.

### **Example IAM policy statements**

Use the following example IAM policy statements to grant TagResource permissions to users creating DataSync resources.

The following statement allows users to tag a DataSync agent when they create the agent.

**JSON** 

```
{
  "Version": "2012-10-17",
  "Statement": [
      {
          "Effect": "Allow",
          "Action": "datasync:TagResource",
          "Resource": "arn:aws:datasync:region:account-id:agent/*"
      }
  ]
}
```

The following statement allows users to tag a DataSync location when they create the location.

**JSON** 

The following statement allows users to tag a DataSync task when they create the task.

**JSON** 

### **Cross-service confused deputy prevention**

The confused deputy problem is a security issue where an entity that doesn't have permission to perform an action can coerce a more-privileged entity to perform the action. In AWS, cross-service impersonation can result in the confused deputy problem. Cross-service impersonation can occur when one service (the *calling service*) calls another service (the *called service*). The calling service can be manipulated to use its permissions to act on another customer's resources in a way it should not otherwise have permission to access. To prevent this, AWS provides tools that help you protect your data for all services with service principals that have been given access to resources in your account.

We recommend using the <a href="mailto:aws:SourceArn">aws:SourceAccount</a> global condition context keys in resource policies to limit the permissions that AWS DataSync gives another service to the resource. If you use both global condition context keys and the <a href="mailto:aws:SourceArn">aws:SourceArn</a> value contains the account ID, the <a href="mailto:aws:SourceArn">aws:SourceArn</a> value must use the same account ID when used in the same policy statement. Use <a href="mailto:aws:SourceArn">aws:SourceArn</a> if you want only one resource to be associated with the cross-service access. Use <a href="mailto:aws:SourceAccount">aws:SourceAccount</a> if you want any resource in that account to be associated with the cross-service use.

The value of aws: SourceArn must include the DataSync location ARN with which DataSync is allowed to assume the IAM role.

The most effective way to protect against the confused deputy problem is to use the aws:SourceArn key with the full ARN of the resource. If you don't know the full ARN or if you're

specifying multiple resources, use wildcard characters (\*) for the unknown portions. Here are some examples of how to do this for DataSync:

- To limit the trust policy to an existing DataSync location, include the full location ARN in the policy. DataSync will assume the IAM role only when dealing with that particular location.
- When creating an Amazon S3 location for DataSync, you don't know the location's ARN. In these scenarios, use the following format for the aws:SourceArn key: arn:aws:datasync:us-east-2:123456789012:\*. This format validates the partition (aws), account ID, and Region.

The following full example shows how you can use the aws:SourceArn and aws:SourceAccount global condition context keys in a trust policy to prevent the confused deputy problem with DataSync.

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "datasync.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                     "aws:SourceAccount": "123456789012"
                },
                "StringLike": {
                     "aws:SourceArn": "arn:aws:datasync:us-east-2:123456789012:*"
                }
            }
        }
    ]
}
```

For more example policies that show how you can use the aws:SourceArn and aws:SourceAccount global condition context keys with DataSync, see the following topics:

- Create a trust relationship that allows DataSync to access your Amazon S3 bucket
- Configure an IAM role to access your Amazon S3 bucket

### **Compliance validation for AWS DataSync**

To learn whether an AWS service is within the scope of specific compliance programs, see <u>AWS</u> <u>services in Scope by Compliance Program</u> and choose the compliance program that you are interested in. For general information, see <u>AWS Compliance Programs</u>.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact.

Your compliance responsibility when using AWS services is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- <u>Security Compliance & Governance</u> These solution implementation guides discuss architectural considerations and provide steps for deploying security and compliance features.
- HIPAA Eligible Services Reference Lists HIPAA eligible services. Not all AWS services are HIPAA eligible.
- <u>AWS Compliance Resources</u> This collection of workbooks and guides might apply to your industry and location.
- <u>AWS Customer Compliance Guides</u> Understand the shared responsibility model through the lens of compliance. The guides summarize the best practices for securing AWS services and map the guidance to security controls across multiple frameworks (including National Institute of Standards and Technology (NIST), Payment Card Industry Security Standards Council (PCI), and International Organization for Standardization (ISO)).
- <u>Evaluating Resources with Rules</u> in the *AWS Config Developer Guide* The AWS Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- <u>AWS Security Hub</u> This AWS service provides a comprehensive view of your security state within AWS. Security Hub uses security controls to evaluate your AWS resources and to check your compliance against security industry standards and best practices. For a list of supported services and controls, see <u>Security Hub controls reference</u>.
- <u>Amazon GuardDuty</u> This AWS service detects potential threats to your AWS accounts, workloads, containers, and data by monitoring your environment for suspicious and malicious

Compliance validation 370

activities. GuardDuty can help you address various compliance requirements, like PCI DSS, by meeting intrusion detection requirements mandated by certain compliance frameworks.

• AWS Audit Manager – This AWS service helps you continuously audit your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

### Resilience in AWS DataSync

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.



#### Note

If an Availability Zone you're migrating data to or from does fail while you're running a DataSync task, the task also will fail.

For more information about AWS Regions and Availability Zones, see AWS global infrastructure.

## Infrastructure security in AWS DataSync

As a managed service, AWS DataSync is protected by AWS global network security. For information about AWS security services and how AWS protects infrastructure, see AWS Cloud Security. To design your AWS environment using the best practices for infrastructure security, see Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

You use AWS published API calls to access DataSync through the network. Clients must support the following:

- Transport Layer Security (TLS). We require TLS 1.2 and recommend TLS 1.3.
- · Cipher suites with perfect forward secrecy (PFS) such as DHE (Ephemeral Diffie-Hellman) or ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Most modern systems such as Java 7 and later support these modes.

Resilience 371

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

### Securing storage location credentials

DataSync uses locations to access your storage resources located on premises, in other clouds, or in AWS. Some location types require you to provide credentials, such as an access key and secret key or a user name and password, to authenticate with your storage system. When you create a DataSync location that requires credentials for authentication, you can choose one of the following options to control how the secret for your credentials is stored:

- Store the secret in AWS Secrets Manager using a service-managed secret encrypted with a default key.
- Store the secret in AWS Secrets Manager using a service-managed secret encrypted with an AWS KMS key that you manage.
- Store the secret in AWS Secrets Manager using a secret and key that you create and manage. DataSync accesses this secret using an IAM role that you provide.

In all cases, the Secrets Manager secret is stored in your account, allowing you to update the secret as needed, independent of the DataSync service. You are charged only for the use of secrets that you create outside of DataSync. Secrets created and managed by DataSync have the prefix awsdatasync.

### Using a service-managed secret encrypted with a default key

When you create your DataSync location, you simply provide the secret string. DataSync creates a secret resource in AWS Secrets Manager to store the secret you provide, and encrypts the secret with the default Secrets Manager KMS key for your account. You can change the secret value directly in Secrets Manager, or by updating the location using the DataSync console, AWS CLI, or SDK. When you delete the location resource or update it to use a custom secret, DataSync deletes the secret resource automatically.



#### Note

To create, modify, and delete secret resources in Secrets Manager, DataSync must have the appropriate permissions. For more information, see AWS managed policies for DataSync.

### Using a service-managed secret encrypted with a custom AWS KMS key

When you create your DataSync location, you provide the secret and the ARN of your AWS KMS key. DataSync automatically creates a secret resource in AWS Secrets Manager to store the secret you provide, and encrypts it using your AWS KMS key. You can change the secret value directly in Secrets Manager, or by updating the location using the DataSync console, AWS CLI, or SDK. When you delete the location resource or update it to use a custom secret, DataSync deletes the secret resource automatically.



#### Note

Your AWS KMS key must use symmetric encryption with the ENCRYPT\_DECRYPT key type. For more information, see Choosing a AWS Key Management Service key in the AWS Secrets Manager User Guide.

To create, modify, and delete secret resources in Secrets Manager, DataSync must have the appropriate permissions. For more information, see AWS managed policy: AWSDataSyncFullAccess.

In addition to using the correct DataSync managed policy, you also need the following permissions:

```
{
    "Sid": "DataSyncKmsPermissions",
    "Effect": "Allow",
    "Action": [
        "kms:Encrypt",
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": "your-kms-key-arn",
    "Condition": {
        "StringLike": {
            "kms:ViaService": "secretsmanager.*.amazonaws.com"
        }
    }
}
```

Replace *your-kms-key-arn* with your KMS key ARN.

To retrieve and decrypt the secret value, DataSync uses a Service Linked Role (SLR) to access your AWS KMS key. To make sure DataSync can use your KMS key, add the following to the key's policy statement:

Replace accountid with your AWS account ID.

### Using a secret that you manage

Before you create your DataSync location, <u>create a secret in AWS Secrets Manager</u>. The value for the secret must only contain the secret string itself in plain text. When you create your DataSync location, you provide the ARN of your secret and an IAM role that DataSync uses to access both your secret and the AWS KMS key used to encrypt your secret. To create an IAM role with the appropriate permissions, do the following:

- 1. Open the IAM console at <a href="https://console.aws.amazon.com/iam/">https://console.aws.amazon.com/iam/</a>.
- 2. In the left navigation pane, under **Access management**, choose **Roles**, and then choose **Create role**.
- 3. On the **Select trusted entity** page, for **Trusted entity type**, choose **AWS service**.
- 4. For **Use case**, choose **DataSync** from the drop-down list. Choose **Next**.
- 5. On the **Add permissions** page, choose **Next**. Enter a name for your role, and then choose **Create role**.
- 6. On the **Roles** page, search for the role that you just created and choose its name.
- 7. On the **Details** page for the role, choose the **Permissions** tab. Choose **Add permissions**, and then **Create inline policy**.
- 8. Choose the **JSON** tab and add the following permissions into the policy editor:

**JSON** 

Replace your-secret-arn with the ARN of your Secrets Manager secret.

- 9. Choose **Next**. Enter a name for your policy, and then choose **Create policy**.
- 10. (Recommended) To prevent the cross-service confused deputy problem, do the following:
  - a. On the **Details** page for the role, choose the **Trust relationships** tab. Choose **Edit trust policy**.
  - b. Update the trust policy by using the following example, which includes the aws:SourceArn and aws:SourceAccount global condition context keys:

**JSON** 

#### c. Choose Update policy.

You can specify this role when creating your location. If your secret uses a customer-managed AWS KMS key for encryption, then you will also need to update the key's policy to allow access from the role you created in the previous procedure. To update the policy, add the following to your AWS KMS key's policy statement:

```
"Sid": "Allow DataSync use of the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam accountid:role/your-role-name"
},
    "Action": "kms:Decrypt",
    "Resource": "*"
}
```

Replace *accountid* with your AWS account ID, and *your-role-name* with the name of the IAM role you created in the previous procedure.

### Note

When you store secrets in Secrets Manager, your AWS account incurs charges. For information about pricing, see AWS Secrets Manager Pricing.

# **AWS DataSync quotas**

Find out about resource quotas and limits when working with AWS DataSync.

# Storage system, file, and object limits

The following table describes the limits that DataSync has when working with storage systems, files, and objects.

Description	Limit
Maximum total file path length	4,096 bytes
Maximum file path component (file name, directory, or subdirectory) length	255 bytes
Maximum length of Windows domain	253 characters
Maximum length of server hostname	255 characters
Maximum Amazon S3 object name length	1,024 UTF-8 characters

# **DataSync quotas**

The following table describes the quotas for DataSync resources in a specific AWS account and Region.

Resource	Quota	Adjustable
Maximum number of tasks you can create	100	Yes
(Enhanced mode tasks) Maximum number of source and destination objects that DataSync can work with per task execution	Virtually unlimited	N/A
For more information, see <u>How DataSync</u> <u>transfers files</u> , <u>objects</u> , and <u>directories</u>		

#### Quota **Adjustable** Resource 50 million (Basic mode tasks) Maximum number of Yes source and destination files, objects, and directories that DataSync can work with per (i) Tip ▲ Important task execution between on-premises, self-Remember the Instead managed, or other cloud storage and AWS of following about this storage services quota: requestin g an For more information, see How DataSync • If you transfer transfers files, objects, and directories increase. Amazon S3 objects you with prefixes, the can prefixes are treated create as directories and tasks count towards the that quota. For example, focus DataSync would on considers3:// specific bucket/foo/ directori bar.txt as two es directories (./ and using ./foo/) and one include object (bar.txt). and · If your task is exclude working with more filters. than 20 million For files, objects, or more directories, make informati sure that you on, allocate a minimum see of 64 GB of RAM filtering to your DataSync the agent. For more data information, see transferr agent requireme

DataSync quotas 378

Resource	Quota	Adjustable
	nts for DataSync transfers.	ed by DataSync.

DataSync quotas 379

#### Quota **Adjustable** Resource 25 million (Basic mode tasks) Maximum number of Yes source and destination files, objects, and directories that DataSync can work with per (i) Tip ▲ Important task execution between AWS storage services If you transfer Instead Amazon S3 objects of For more information, see How DataSync with prefixes, the transfers files, objects, and directories requestin prefixes are treated g an as directories and increase, count towards the you quota. For example, can DataSync would create considers3://buck tasks et/foo/bar.txt that focus as two directories (./ and ./foo/) and one on object (bar.txt). specific directori es using include and exclude filters. For more informati on, see filtering the data transferr

DataSync quotas 380

Resource	Quota	Adjustable
		ed by DataSync.
Maximum throughput per task (for transfers that use a DataSync agent)	10 Gbps	No
Maximum throughput per task (for transfers that don't use a DataSync agent)	5 Gbps	No
Maximum number of characters you can include in a task filter	102,400 characters  (i) Note  If you're using the  DataSync console, this limit includes all the characters combined in your include and exclude patterns.	No
Maximum number of queued executions for a single task	50	No
Maximum number of concurrent Enhanced mode task executions	120	No
Maximum number of days a task execution's history is retained	30	No

# Request a quota increase

You can request an increase for some DataSync quotas. Increases aren't granted right away and might take a couple of days to take effect.

Request a quota increase 381

#### To request a quota increase

- 1. Open the Service Quotas console at <a href="https://console.aws.amazon.com/servicequotas/">https://console.aws.amazon.com/servicequotas/</a>.
- 2. In the navigation pane, choose AWS services and then choose AWS DataSync.
- 3. Choose the quota that you want to increase, then choose Request increase at account-level.
- 4. Enter the total amount that you want the quota to be, then choose **Request**.

If you need to increase a different quota, fill out a separate request.

Request a quota increase 382

## Troubleshooting AWS DataSync issues

Use the following information to troubleshoot AWS DataSync issues and errors.

## **Topics**

- Troubleshooting issues with DataSync agents
- Troubleshooting issues with DataSync locations
- Troubleshooting issues with DataSync tasks
- Troubleshooting data verification issues
- Troubleshooting higher than expected S3 storage costs with DataSync

## Troubleshooting issues with DataSync agents

Use the following information to help you troubleshoot issues with AWS DataSync agents. Some of these issues can include:

- Trouble connecting to an Amazon EC2 agent's local console
- Failing to retrieve an agent's activation key
- Issues activating an agent with a VPC service endpoint
- Discovering an agent is offline

## How do I connect to an Amazon EC2 agent's local console?

To connect to an Amazon EC2 agent's local console, you must use SSH. Make sure that your EC2 instance's security group allows access with SSH (TCP port 22).

In a terminal, run the following ssh command to connect to the instance:

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-ip-address
```

- For /path/key-pair-name, specify the path and file name (.pem) of the private key required to connect to your instance.
- For *instance-user-name*, specify admin.

Troubleshooting agent issues 383

• For instance-public-ip-address, specify the public IP address of your instance.

## What does the Failed to retrieve agent activation key error mean?

When activating your DataSync agent, the agent connects to the service endpoint that you specify to request an activation key. This error likely means that your network security settings are blocking the connection.

#### Action to take

If you're using a virtual private cloud (VPC) service endpoint, verify that your security group settings allow your agent to connect to the VPC endpoint. For information about required ports, see Network requirements for VPC service endpoints.

If you're using a public or Federal Information Processing Standard (FIPS) endpoint, check that your firewall and router settings allow your agent to connect to the endpoint. For information, see Network requirements for public or FIPS service endpoints.

## I still can't activate an agent by using a VPC service endpoint

If you're still having issues activating a DataSync agent with a VPC service endpoint, see <u>I don't</u> know what's going on with my agent. Can someone help me?

## What do I do if my agent is offline?

Your DataSync agent can be offline for a few reasons, but you might be able to get it back online. Before you delete the agent and create a new one, go through the following checklist to help you understand what might have happened.

- **Contact your backup team** If your agent is offline because its virtual machine (VM) was restored from a snapshot or backup, you might need to replace the agent.
- Check if the agent's VM or Amazon EC2 instance is off Depending on the type of agent that you're using, try turning the VM or EC2 instance back on if it's off. Once it's on again, test your agent's network connectivity to AWS.
- Verify your agent meets the minimum hardware requirements Your agent might be offline
  because its VM or EC2 instance configuration was accidentally changed since the agent was
  activated. For example, if your VM no longer has the minimum required memory or space, the
  agent might appear as offline. For more information, see <a href="Requirements for AWS DataSync">Requirements for AWS DataSync</a>
  agents.

• Wait for agent-related software updates to finish – Your agent might go offline briefly following software updates provided by AWS. If you believe this is why the agent is offline, wait a short period then check if the agent is back online.

• Check your VPC service endpoint settings – If your offline agent is using a public service endpoint and also in the same VPC where you created a VPC service endpoint for DataSync, you might need to disable private DNS support for that VPC endpoint.

If none of these seem to be the reason that the agent is offline, you likely need to replace the agent.

## I don't know what's going on with my agent. Can someone help me?

You can allow AWS Support to access your DataSync agent and help troubleshoot agent-related issues. You must enable this access through the agent's local console.

#### To provide Support access to your agent

- 1. Log in to your agent's local console.
- At the prompt, enter 5 to open the command prompt (for VMware VMs, use 6). 2.
- Enter **h** to open the **AVAILABLE COMMANDS** window. 3.
- In the **AVAILABLE COMMANDS** window, enter the following to connect to Support: 4.

#### open-support-channel

If you are using the agent with VPC endpoints, you must provide a VPC endpoint IP address for your support channel, as follows:

#### open-support-channel vpc-ip-address

Your firewall must allow the inbound TCP port 22 to initiate a support channel to AWS. When you connect to Support, DataSync assigns you a support number. Make a note of your support number.



#### Note

The channel number isn't a TCP/UDP port number. Instead, it makes an SSH (TCP 22) connection to servers and provides the support channel for the connection.

5. When the support channel is established, provide your support service number to Support so that they can provide troubleshooting assistance.

- 6. When the support session is finished, press **Enter** to end it.
- 7. Enter **exit** to log out of the DataSync local console.
- 8. Follow the prompts to exit the local console.

## Troubleshooting issues with DataSync locations

Use the following information to help you troubleshoot issues with AWS DataSync locations. Some of these issues can include:

- Permissions and mount errors with NFS locations
- File ownership issues
- Problems accessing SMB locations that use Kerberos authentication
- Permission and access issues with object storage, such as Amazon S3 and Microsoft Azure Blob locations

## My task failed with an NFS permissions denied error

You can get a "permissions denied" error message if you configure your NFS file server with root\_squash or all\_squash and your files don't all have read access.

#### Action to take

To fix this issue, configure your NFS export with no\_root\_squash or make sure that the permissions for all of the files that you want to transfer allow read access for all users.

For DataSync to access directories, you must also enable all-execute access. To make sure that the directory can be mounted, first connect to any computer that has the same network configuration as your agent. Then run the following CLI command:

mount -t nfs -o nfsvers=<your-nfs-server-version> <your-nfs-servername>:<nfs-export-path-you-specified> <new-test-folder-on-your-computer>

If the issue still isn't resolved, contact AWS Support Center.

## My task failed with an NFS mount error

You might see the following error when running a DataSync task that involves an NFS file server location:

Task failed to access location loc-1111222233334444a: x40016: mount.nfs: Connection timed out

#### **Actions to take**

Do the following until the error is resolved.

- 1. Make sure that the NFS file server and export that you specify in your DataSync location are valid. If they aren't, delete your location and task, then create a new location and task that use a valid NFS file server and export. For more information, see Using the DataSync console.
- Check your firewall configuration between your agent and NFS file server. For more
  information, see Network requirements for on-premises, self-managed, other cloud, and edge
  storage.
- 3. Make sure that your agent can access the NFS file server and mount the export. For more information, see Providing DataSync access to NFS file servers.
- 4. If you still see the error, open a support channel with Support. For more information, see <u>I</u> don't know what's going on with my agent. Can someone help me?.

## My task failed with an Amazon EFS mount error

You might see the following error when running a DataSync task that involves an Amazon EFS location:

Task failed to access location loc-1111222233334444a: x40016: Failed to connect to EFS mount target with IP: 10.10.1.0.

This can happen if the Amazon EFS file system's mount path that you configure with your location gets updated or deleted. DataSync isn't aware of these changes in the file system.

#### Action to take

Delete your location and task and create a new Amazon EFS location with the new mount path.

## File ownership isn't maintained with NFS transfer

After your transfer, you might notice that the files in your DataSync destination location have different user IDs (UIDs) or group IDs (GIDs) than the same files in your source location. For example, the files in your destination might have a UID of 65534, 99, or nobody.

This can happen if a file system involved in your transfer uses NFS version 4 ID mapping, a feature that DataSync doesn't support.

#### Action to take

You have a couple options to work around this issue:

- Create a new location for the file system that uses NFS version 3 instead of version 4.
- Disable NFS version 4 ID mapping on the file system.

Retry the transfer. Either option should resolve the issue.

## My task can't access an SMB location that uses Kerberos

DataSync errors with SMB locations that use <u>Kerberos authentication</u> are typically related to mismatches between your location and Kerberos configurations. There also might be a network issue.

#### Failed to access location

The following error indicates that there might be configuration issues with your SMB location or Kerberos setup:

Task failed to access location

#### Verify the following:

- The SMB file server that you specify for your location is a domain name. For Kerberos, you
  can't specify the file server's IP address.
- The Kerberos principal that you specify for your location matches the principal that you use to create the Kerberos key table (keytab) file. Principal names are case sensitive.
- The Kerberos principal's mapped user password hasn't changed since you created the keytab file. If the password changes (because of password rotation or some other reason), your task execution might fail with the following error:

Task failed to access location loc-1111222233334444a: x40015: kinit: Preauthentication failed while getting initial credentials

#### Can't contact KDC realm

The following error indicates a networking issue:

kinit: Cannot contact any KDC for realm 'MYDOMAIN.ORG' while getting initial
 credentials"

#### Verify the following:

- The Kerberos configuration file (krb5.conf) that you provided DataSync has the correct information about your Kerberos realm. For an example krb5.conf file, see Kerberos authentication prerequisites.
- The Kerberos Key Distribution Center (KDC) server port is open. The KDC port is typically TCP port 88.
- The DNS configuration on your network.

## My task failed with an input/output error

You can get an input/output error message if your storage system fails I/O requests from the DataSync agent. Common reasons for this include a server disk failure, changes to your firewall configuration, or a network router failure.

If the error involves an NFS file server or Hadoop Distributed File System (HDFS) cluster, use the following steps to resolve the error.

## Actions to take (NFS)

First, check your NFS file server's logs and metrics to determine if the problem started on the NFS server. If yes, resolve that issue.

Next, check that your network configuration hasn't changed. To check if the NFS file server is configured correctly and that DataSync can access it, do the following:

- 1. Set up another NFS client on the same network subnet as the agent.
- 2. Mount your share on that client.
- Validate that the client can read and write to the share successfully.

#### Actions to take (HDFS)

Do the following until you resolve the error:

 Make sure that your HDFS cluster allows your DataSync agent to communicate with the cluster's NameNode and DataNode ports.

In most clusters, you can find the port numbers that the cluster uses in the following configuration files:

- To find the NameNode port, look in the core-site.xml file under the fs.default or fs.default.name property (depending on the Hadoop distribution).
- To find the DataNode port, look in the hdfs-site.xml file under the dfs.datanode.address property.
- 2. In your hdfs-site.xml file, verify that your dfs.data.transfer.protection property has only one value. For example:

## Error: FsS3UnableToConnectToEndpoint

DataSync can't connect to your <u>Amazon S3 location</u>. This could mean the location's S3 bucket isn't reachable or the location isn't configured correctly.

Do the following until you resolve the issue:

- Check if DataSync can access your S3 bucket.
- Make your sure location is configured correctly by using the DataSync console or DescribeLocationS3 operation.

## Error: FsS3HeadBucketFailed

DataSync can't access the S3 bucket that you're transferring to or from. Check if DataSync has permission to access the bucket by using the Amazon S3 <u>HeadBucket</u> operation. If you need to adjust your permissions, see <u>Providing DataSync access to S3 buckets</u>.

# Task fails with an Unable to list Azure Blobs on the volume root error

If your DataSync transfer task fails with an Unable to list Azure Blobs on the volume root error, there might be an issue with your shared access signature (SAS) token or your Azure storage account's network.

#### Actions to take

Try the following and run your task again until you fix the issue:

- Make sure that your <u>SAS token</u> has the right permissions to access your Microsoft Azure Blob Storage.
- If you're running your DataSync agent in Azure, configure your storage account to allow access from the virtual network where your agent resides.
- If you're running your agent on Amazon EC2, configure your Azure storage firewall to allow access from the agent's public IP address.

For information on how to configure your Azure storage account's network, see the <u>Azure Blob</u> Storage documentation.

## Error: FsAzureBlobVolRootListBlobsFailed

The shared access signature (SAS) token that DataSync uses to access your Microsoft Azure Blob Storage doesn't have the List permission.

To resolve the issue, <u>update your location</u> with a token that has the List permission and try running your task again.

## **Error: SrcLocHitAccess**

DataSync can't access your source location. Check that DataSync has permission to access the location and try running your task again.

## Error: SyncTaskErrorLocationNotAdded

DataSync can't access your location. Check that DataSync has permission to access the location and try running your task again.

# Task with S3 source location fails with HeadObject or GetObjectTagging error

If you're transferring objects with specific version IDs from an S3 bucket, you might see an error related to HeadObject or GetObjectTagging. For example, here's an error related to GetObjectTagging:

```
[WARN] Failed to read metadata for file /picture1.png (versionId: 111111): S3 Get
  Object Tagging Failed
[ERROR] S3 Exception: op=GetObjectTagging photos/picture1.png, code=403, type=15,
  exception=AccessDenied,
msg=Access Denied req-hdrs: content-type=application/xml, x-amz-api-version=2006-03-01
  rsp-hdrs: content-type=application/xml,
date=Wed, 07 Feb 2024 20:16:14 GMT, server=AmazonS3, transfer-encoding=chunked,
x-amz-id-2=IOWQ4fDEXAMPLEQM+ey7N9WgVhSnQ6JEXAMPLEZb7hSQDASK+Jd1vEXAMPLEa3Km, x-amz-
request-id=79104EXAMPLEB723
```

If you see either of these errors, validate that the IAM role that DataSync uses to access your S3 source location has the following permissions:

- s3:GetObjectVersion
- s3:GetObjectVersionTagging

If you need to update your role with these permissions, see <u>Creating an IAM role for DataSync to access your Amazon S3 location</u>.

## Troubleshooting issues with DataSync tasks

Use the following information to help you troubleshoot issues with AWS DataSync tasks and task executions. These issues might include task setup problems, stuck task executions, and data not transferring as expected.

# Error: Invalid SyncOption value. Option: TransferMode, PreserveDeletedFiles, Value: ALL, REMOVE.

This error occurs when you're creating or editing your DataSync task and you select the **Transfer all** data option and deselect the **Keep deleted files** option.

When you transfer all data, DataSync doesn't scan your destination location and doesn't know what to delete.

## Task execution fails with an EniNotFound error

This error occurs if you delete one of your task's network interfaces in your virtual private cloud (VPC). If your task is scheduled or queued, the task will fail if it's missing a <u>network interface</u> required to transfer your data.

#### **Actions to take**

You have the following options to work around this issue:

- Manually restart the task. When you do this, DataSync will create any missing network interfaces it needs to run the task.
- If you need to clean up resources in your VPC, make sure that you don't delete network interfaces related to a DataSync task that you're still using.

To see the network interfaces allocated to your task, do one of the following:

- Use the <u>DescribeTask</u> operation. You can view the network interfaces in the SourceNetworkInterfaceArns and DestinationNetworkInterfaceArns response elements.
- In the Amazon EC2 console, search for your task ID (such as task-f012345678abcdef0) to find its network interfaces.
- Consider not running your tasks automatically. This could include disabling task queueing or scheduling (through DataSync or custom automation).

## Task execution fails with a Cannot allocate memory error

When your DataSync task fails with a Cannot allocate memory error, it can mean a few different things.

#### Action to take

Try the following until you no longer see the issue:

 If your transfer involves an agent, make sure that the agent meets the <u>virtual machine (VM)</u> or Amazon EC2 instance requirements.

• Split your transfer into multiple tasks by using <u>filters</u>. It's possible that you're trying to transfer more files or objects than what one DataSync task can handle.

• If you still see the issue, contact Support.

# Task execution has a launching status but nothing seems to be happening

Your DataSync task can get stuck with a **Launching** status typically because the agent is powered off or has lost network connectivity.

#### Action to take

Make sure that your agent's status is **ONLINE**. If the agent is **OFFLINE**, make sure it's powered on.

If the agent is powered on and the task is still **Launching**, then there's likely a network connection problem between your agent and AWS. For information about how to test network connectivity, see Verifying your agent's connection to the DataSync service.

If you're still having this issue, see I don't know what's going on with my agent. Can someone help me?.

## Task execution seems stuck in the preparing status

The time your DataSync transfer task has the **Preparing** status depends on the amount of data in your transfer source and destination and the performance of those storage systems.

When a task starts, DataSync performs a recursive directory listing to discover all files, objects, directories, and metadata in your source and destination. DataSync uses these listings to identify differences between storage systems and determine what to copy. This process can take a few minutes or even a few hours.

#### **Action to take**

You shouldn't have to do anything. Continue to wait for the task status to change to **Transferring**. If the status still doesn't change, contact AWS Support Center.

## Task execution stops before the transfer finishes

If your DataSync task execution stops early, your task configuration might include an AWS Region that's disabled in your AWS account.

#### Actions to take

Do the following to run your task again:

1. Check the opt-in status of your task's Regions and make sure they're enabled.

2. Start the task again.

## Task execution fails when transferring from a Google Cloud Storage bucket

Because DataSync communicates with Google Cloud Storage by using the Amazon S3 API, there's a limitation that might cause your DataSync transfer to fail if you try to copy object tags. The following message related to the issue appears in your CloudWatch logs:

[WARN] Failed to read metadata for file /your-bucket/your-object: S3 Get Object Tagging Failed: proceeding without tagging

To prevent this, deselect the **Copy object tags** option when configuring your transfer task settings.

## There are mismatches between task execution's timestamps

When looking at the DataSync console or Amazon CloudWatch logs, you might notice that the start and end times for your DataSync task execution don't match the timestamps you see in other monitoring tools. This is because the console and CloudWatch logs take into account the time a task execution spends in the launching or queueing states, while some other tools don't.

You might notice this discrepancy when comparing execution timestamps between the DataSync console or CloudWatch logs and the following places:

- Logs for the file system involved in your transfer
- The last modified date on an Amazon S3 object that DataSync wrote to
- Network traffic coming from the DataSync agent
- Amazon EventBridge events

## Task execution fails with NoMem error

The set of data you're trying to transfer may be too large for DataSync. If you see this error, contact AWS Support Center.

# Object fails to transfer to Azure Blob Storage with user metadata key error

When transferring from an S3 bucket to Azure Blob Storage, you might see the following error:

[ERROR] Failed to transfer file /user-metadata/file1: Azure Blob user metadata key must be a CSharp identifier

This means that /user-metadata/file1 includes user metadata that doesn't use a valid C# identifier. For more information, see the Microsoft documentation.

## There's an /.aws-datasync folder in the destination location

DataSync creates a folder called /.aws-datasync in your destination location to help facilitate your data transfer.

While DataSync typically deletes this folder following your transfer, there might be situations where this doesn't happen.

#### Action to take

Delete this folder anytime as long as you don't have a running task execution copying to that location.

## Can't transfer symbolic links between locations using SMB

When your task execution finishes, you see the following error:

Transfer and verification completed. Selected files transferred except for files skipped due to errors. If no skipped files are listed in Cloud Watch Logs, please contact AWS Support for further assistance.

When transferring between SMB storage systems (such as an SMB file server and Amazon FSx for Windows File Server file system), you might see the following warnings and errors in your CloudWatch logs:

[WARN] Failed to read metadata for file /appraiser/symlink: No data available [ERROR] Failed to read metadata for directory /appraiser/symlink: No data available

#### Action to take

DataSync doesn't support transferring symbolic links (or hard links) when transferring between these location types. For more information, see <u>Links and directories copied by AWS DataSync</u>.

## Task report errors

You might run into one of the following errors while trying to monitor your DataSync transfer with a task report.

Error message	Workaround
File path exceeds the maximum length of 4,096 characters. Cannot write to Task Report	N/A (DataSync can't transfer a file with a path that exceeds 4,096 bytes)
	For more information, see <u>Storage system, file,</u> and object limits.
Failed to upload Task Report(s) to S3 due to an invalid bucket or IAM role	Check that the <u>DataSync IAM role</u> has the right permissions to upload a task report to your S3 bucket.
Execution error occurred prior to generating any Task Reports	Check your <u>CloudWatch logs</u> to identify why your task execution failed.

## **Troubleshooting data verification issues**

By default, AWS DataSync <u>verifies the integrity</u> of your data at the end of a transfer. Use the following information to help you diagnose common verification errors and warnings, such as files being modified or deleted before DataSync finishes verifying your data.

With verification issues, many times it helps to review your <u>CloudWatch logs</u> (or <u>task reports</u>) in addition to the task execution error that you're seeing. DataSync provides JSON-structured logs for Enhanced mode tasks, while Basic mode tasks have unstructured logs.

## There are mismatches between a file's contents

When your task execution finishes, you see the following error:

Task report errors 397

Transfer and verification completed. Verification detected mismatches. Files with mismatches are listed in Cloud Watch Logs

In your CloudWatch logs, you might notice failed verifications for contents that differ between the source and destination locations. This can happen if files are modified during your transfer.

For example, the following logs shows that file1.txt has different mtime, srcHash, and dstHash values:

#### Basic mode log example

```
[NOTICE] Verification failed <> /directory1/directory2/file1.txt
[NOTICE] /directory1/directory2/file1.txt srcMeta: type=R mode=0755 uid=65534
gid=65534 size=534528 atime=1633100003/684349800 mtime=1602647222/222919600
extAttrsHash=0
[NOTICE] srcHash: 0c506c26bd1e43bd3ac346734f1a9c16c4ad100d1b43c2903772ca894fd24e44
[NOTICE] /directory1/directory2/file1.txt dstMeta: type=R mode=0755 uid=65534
gid=65534 size=511001 atime=1633100003/684349800 mtime=1633106855/859227500
extAttrsHash=0
[NOTICE] dstHash: dbd798929f11a7c0201e97f7a61191a83b4e010a449dfc79fbb8233801067c46
```

In DataSync, mtime represents the last time a file was written to before <u>preparation</u>. When verifying transfers, DataSync compares mtime values between source and destination locations. A verification failure like this occurs if the mtime for a file isn't the same for both locations. The differences between srcHash and dstHash indicate the file's contents don't match at both locations.

#### Actions to take

Do the following:

- 1. Use an epoch time converter to determine whether the source or destination file or object was modified more recently. This can help identify which version is current.
- 2. To avoid this error again, <u>schedule your task</u> to run during a maintenance window when there's no activity at your source and destination.

## There's a mismatch between a file's SMB metadata

When your task execution finishes, you see the following error:

Transfer and verification completed. Verification detected mismatches. Files with mismatches are listed in Cloud Watch Logs

When transferring between storage systems that support the Server Message Block (SMB) protocol, you might see this error when a file's extended SMB attributes don't match between source and destination.

For example, the following logs show that file1.txt has a different extAttrsHash value between locations, indicating the file contents are identical but extended attributes weren't set at the destination:

## Basic mode log example

```
[NOTICE] Verification failed <> /directory1/directory2/file1.txt
[NOTICE] /directory1/directory2/file1.txt srcMeta: type=R mode=0755 uid=65534
gid=65534 size=1469752 atime=1631354985/174924200 mtime=1536995541/986211400
extAttrsHash=2272191894
[NOTICE] srcHash: 38571d42b646ac8f4034b7518636b37dd0899c6fc03cdaa8369be6e81a1a2bb5
[NOTICE] /directory1/directory2/file1.txt dstMeta: type=R mode=0755 uid=65534
gid=65534 size=1469752 atime=1631354985/174924200 mtime=1536995541/986211400
extAttrsHash=3051150340
[NOTICE] dstHash: 38571d42b646ac8f4034b7518636b37dd0899c6fc03cdaa8369be6e81a1a2bb5
```

You might also see a related error message about extended attributes:

```
[ERROR] Deferred error: WriteFileExtAttr2 failed to setextattrlist(filename="/
directory1/directory2/file1.txt"): Input/output error
```

#### **Action to take**

This error typically occurs when there are insufficient permissions to copy access control lists (ACLs) to the destination. To resolve this issue, review the following configuration guides based on your destination type:

- Required permissions with FSx for Windows File Server file systems
- Required permissions with FSx for ONTAP file systems that use SMB

## Files to transfer are no longer at source location

When your task execution finishes, you see the following error:

Transfer and verification completed. Selected files transferred except for files skipped due to errors. If no skipped files are listed in Cloud Watch Logs, please contact AWS Support for further assistance.

In your logs, you might see errors indicating that files aren't at the source location. This can happen if files (such as file1.dll and file2.dll) are deleted after <u>preparation</u> but before DataSync transfers them:

## Basic mode log example

```
[ERROR] Failed to open source file /file1.dll: No such file or directory [ERROR] Failed to open source file /file2.dll: No such file or directory
```

#### Action to take

To avoid these situations, schedule your task to run when there's no activity at the source location.

For example, you can run your task during a maintenance window when users and applications aren't actively working with that location.

In some cases, you might not see logs associated with this error. If that happens, contact <u>AWS</u> <u>Support Center</u>.

## DataSync can't verify destination data

When your task execution finishes, you see the following error:

Transfer and verification completed. Verification detected mismatches. Files with mismatches are listed in Cloud Watch Logs

In your logs, you might notice that DataSync can't verify certain folders or files in the destination location. These errors can look like this:

#### Basic mode log example

[ERROR] Failed to read metadata for destination file /directory1/directory2/ file1.txt: No such file or directory

For files, you might see verification failures like this:

#### Basic mode log example

#### Action to take

These logs indicate that destination data was deleted after the transfer but before verification. (Logs look similar when data is uploaded to a source location during the same time frame.)

To avoid these situations, <u>schedule your task</u> to run when there's no activity at the destination location.

For example, you can run your task during a maintenance window when users and applications aren't actively working with that location.

## DataSync can't read object metadata

When your task execution finishes, you see the following error:

Transfer and verification completed. Selected files transferred except for files skipped due to errors. If no skipped files are listed in Cloud Watch Logs, please contact AWS Support for further assistance.

In your logs, you might notice that DataSync can't read file1.png because of a failed Amazon S3 HeadObject request. <u>DataSync makes HeadObject requests</u> with S3 locations during task preparation and verification.

#### Basic mode log example

```
[WARN] Failed to read metadata for file /file1.png: S3 Head Object Failed
```

#### Actions to take

To fix this issue, verify whether DataSync has the right level of permissions to work with your S3 bucket:

- Make sure that the IAM role that DataSync uses to access your Amazon S3 locations allows the s3:GetObject permission. For more information, see <a href="Required permissions">Required permissions</a>.
- If your S3 bucket uses server-side encryption, make sure that DataSync is allowed to access
  the objects in that bucket. For more information, see <u>Accessing S3 buckets using server-side</u>
  encryption.

## There's a mismatch in an object's system-defined metadata

When your Enhanced mode task execution between S3 buckets finishes, you see the following error:

```
Verification failed due to a difference in metadata
```

You might notice in your logs a mismatch in an object's Amazon S3 <a href="system-defined metadata">system-defined metadata</a>. In this particular example, the source object doesn't have Content-Type metadata but the destination object does. This happened because the destination S3 bucket automatically applied "ContentType": "application/octet-stream" metadata to the object when DataSync transferred it there.

## Enhanced mode log example

```
"Action": "VERIFY",
"Source": {
    "LocationId": "loc-0b3017fc4ba4a2d8d",
    "RelativePath": "encoding/content-null",
    "Metadata": {
        "Type": "Object",
        "ContentSize": 24,
        "LastModified": "2024-12-23T15:48:15Z",
```

```
"S3": {
                "SystemMetadata": {
                    "ETag": "\"68b9c323bb846841ee491481f576ed4a\""
                },
                "UserMetadata": {},
                "Tags": {}
            }
        }
    },
    "Destination": {
        "LocationId": "loc-abcdef01234567890",
        "RelativePath": "encoding/content-null",
        "Metadata": {
            "Type": "Object",
            "ContentSize": 24,
            "LastModified": "2024-12-23T16:00:03Z",
            "S3": {
                "SystemMetadata": {
                    "ContentType": "application/octet-stream",
                    "ETag": "\"68b9c323bb846841ee491481f576ed4a\""
                },
                "UserMetadata": {
                    "file-mtime": "1734968895000"
                },
                "Tags": {}
            }
        }
    "TransferType": "CONTENT_AND_METADATA",
    "ErrorCode": "MetadataDiffers",
    "ErrorDetail": "Verification failed due to a difference in metadata"
}
```

#### Action to take

To avoid this error, update your source location objects to include the Content-Type metadata property.

## **Understanding data verification duration**

DataSync verification includes an SHA256 checksum on file content and an exact comparison of file metadata between locations. How long verification takes depends on several factors,

including the number of files or objects involved, the size of the data in the storage systems, and the performance of these systems.

#### Action to take

Given the factors that can affect verification time, you shouldn't have to do anything. However, if your task execution seems stuck with a verifying status, contact AWS Support Center.

# Troubleshooting higher than expected S3 storage costs with DataSync

If your Amazon S3 storage costs are higher than you thought they would be following an AWS DataSync transfer, it might be due to one or more of the following reasons:

- When transferring to or from S3 buckets, you incur costs related to S3 API requests made by DataSync.
- DataSync uses the Amazon S3 multipart upload feature to upload objects to S3 buckets. This
  approach can result in unexpected storage charges for uploads that don't complete successfully.
- Object versioning might be enabled on your S3 bucket. Object versioning results in Amazon S3 storing multiple copies of objects that have the same name.

#### **Actions to take**

In these cases, you can take the following steps:

- Make sure you understand how DataSync uses S3 requests and how they might be affecting your storage costs. For more information, see <u>Evaluating S3 request costs when using DataSync</u>.
- If the issue's related to multipart uploads, configure a policy for multipart uploads for your S3 bucket to clean up incomplete multipart uploads to reduce storage cost. For more information, see the AWS blog post <u>S3 Lifecycle Management Update - Support for Multipart Uploads and</u> Delete Markers.
- If the issue's related to object versioning, disable object versioning on your S3 bucket.
- If you need more help, contact AWS Support Center.

## **AWS DataSync tutorials**

These tutorials walk you through some real-world scenarios with AWS DataSync.

## **Topics**

- Tutorial: Transferring data from on-premises storage to Amazon S3 across AWS accounts
- Tutorial: Transferring data between Amazon S3 buckets across AWS accounts

## Tutorial: Transferring data from on-premises storage to Amazon S3 across AWS accounts

When using AWS DataSync with on-premises storage, you typically transfer data to an AWS storage service that belongs to the same AWS account as your DataSync agent. There are situations, however, where you might need to transfer data to an Amazon S3 bucket that's associated with a different account.



#### Important

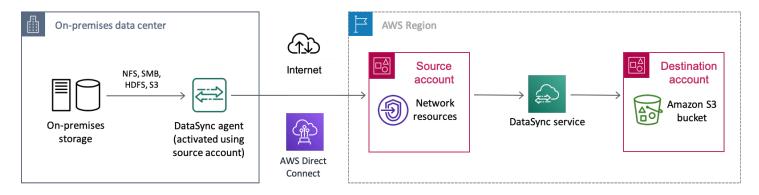
Transferring data across AWS accounts by using the methods in this tutorial works only when Amazon S3 is one of the DataSync transfer locations.

## **Overview**

It's not uncommon to need to transfer data between different AWS accounts, especially if you have separate teams managing your organization's resources. Here's what a cross-account transfer using DataSync can look like:

- Source account: The AWS account for managing network resources. This is the account that you activate your DataSync agent with.
- **Destination account**: The AWS account for managing the S3 bucket that you need to transfer data to.

The following diagram illustrates this kind of scenario.



## Prerequisite: Required source account permissions

For your source AWS account, there are two sets of permissions to consider with this kind of cross-account transfer:

- *User permissions* that allow a user to work with DataSync (this might be you or your storage administrator). These permissions let you create DataSync locations and tasks.
- DataSync service permissions that allow DataSync to transfer data to your destination account bucket.

## User permissions

In your source account, add at least the following permissions to an IAM role for creating your DataSync locations and task. For information on how to add permissions to a role, see <u>creating</u> or <u>modifying</u> an IAM role.

```
"datasync:CancelTaskExecution",
                 "datasync:ListTasks",
                 "datasync:StartTaskExecution",
                 "iam:CreateRole",
                 "iam:CreatePolicy",
                 "iam:AttachRolePolicy",
                 "iam:ListRoles",
                 "s3:GetBucketLocation",
                 "s3:ListAllMyBuckets"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                 "iam:PassRole"
            ],
            "Resource": "*",
            "Condition": {
                 "StringEquals": {
                     "iam:PassedToService": [
                         "datasync.amazonaws.com"
                     ]
                }
            }
        }
    ]
}
```

## Tip

To set up your *user permissions*, consider using <u>AWSDataSyncFullAccess</u>. This is an AWS managed policy that provides a user full access to DataSync and minimal access to its dependencies.

## DataSync service permissions

The DataSync service needs the following permissions in your source account to transfer data to your destination account bucket.

Later in this tutorial, you add these permissions when <u>creating an IAM role</u> for DataSync. You also specify this role (<u>source-datasync-role</u>) in your <u>destination bucket policy</u> and when <u>creating your DataSync destination location</u>.

```
{
  "Version": "2012-10-17",
  "Statement": [
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket"
    },
    {
      "Action": [
        "s3:AbortMultipartUpload",
        "s3:DeleteObject",
        "s3:GetObject",
        "s3:ListMultipartUploadParts",
        "s3:PutObject",
        "s3:GetObjectTagging",
        "s3:PutObjectTagging"
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
    }
 ]
}
```

## Prerequisite: Required destination account permissions

In your destination account, your *user permissions* must allow you to update your destination bucket's policy and disable its access control lists (ACLs). For more information on these specific permissions, see the *Amazon S3 User Guide*.

## Step 1: In your source account, create a DataSync agent

To get started, you must create a DataSync agent that can read from your on-premises storage system and communicate with the DataSync service. This process includes deploying an agent in your on-premises storage environment and activating the agent in your source AWS account.



#### Note

The steps in this tutorial apply to any type of agent and service endpoint that you use.

#### To create a DataSync agent

- 1. Deploy a DataSync agent in your on-premises storage environment.
- 2. Choose a service endpoint that the agent will use to communicate with AWS.
- 3. Activate your agent in your source account.

## Step 2: In your source account, create a DataSync IAM role for destination bucket access

In your source account, you need an IAM role that gives DataSync the permissions to transfer data to your destination account bucket.

Since you're transferring across accounts, you must create the role manually. (DataSync can create this role for you in the console when transferring in the same account.)

## Create the DataSync IAM role

Create an IAM role with DataSync as the trusted entity.

#### To create the IAM role

- 1. Log in to the AWS Management Console with your source account.
- 2. Open the IAM console at https://console.aws.amazon.com/iam/.
- 3. In the left navigation pane, under Access management, choose Roles, and then choose Create role.
- On the Select trusted entity page, for Trusted entity type, choose AWS service. 4.

- For **Use case**, choose **DataSync** in the dropdown list and select **DataSync**. Choose **Next**. 5.
- 6. On the **Add permissions** page, choose **Next**.
- Give your role a name and choose **Create role**. 7.

For more information, see Creating a role for an AWS service (console) in the IAM User Guide.

## Add permissions to the DataSync IAM role

The IAM role that you just created needs the permissions that allow DataSync to transfer data to the S3 bucket in your destination account.

#### To add permissions to your IAM role

- On the Roles page of the IAM console, search for the role that you just created and choose its name.
- On the role's details page, choose the **Permissions** tab. Choose **Add permissions** then **Create** inline policy.
- Choose the **JSON** tab and do the following:
  - Paste the following JSON into the policy editor:

#### Note

The value for aws: ResourceAccount should be the account ID that owns the Amazon S3 bucket specified in the policy.

```
{
 "Version": "2012-10-17",
 "Statement": [
     {
         "Action": [
             "s3:GetBucketLocation",
             "s3:ListBucket",
             "s3:ListBucketMultipartUploads"
         ],
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket"
         "Condition": {
```

```
"StringEquals": {
              "aws:ResourceAccount": "123456789012"
         }
     },
         "Action": [
             "s3:AbortMultipartUpload",
             "s3:DeleteObject",
             "s3:GetObject",
             "s3:GetObjectTagging",
             "s3:GetObjectVersion",
             "s3:GetObjectVersionTagging",
             "s3:ListMultipartUploadParts",
             "s3:PutObject",
             "s3:PutObjectTagging"
           ],
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
         "Condition": {
             "StringEquals": {
                 "aws:ResourceAccount": "123456789012"
             }
         }
    }
]
}
```

- b. Replace each instance of amzn-s3-demo-destination-bucket with the name of the S3 bucket in your destination account.
- 4. Choose **Next**. Give your policy a name and choose **Create policy**.

## Step 3: In your destination account, update your S3 bucket policy

In your destination account, modify the destination S3 bucket policy to include the <u>DataSync IAM</u> role that you created in your source account.

Before you begin: Make sure that you have the required permissions for your destination account.

## To update the destination S3 bucket policy

1. In the AWS Management Console, switch to your destination account.

- 2. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
- 3. In the left navigation pane, choose **Buckets**.
- 4. In the **Buckets** list, choose the S3 bucket that you're transferring data to.
- 5. On the bucket's detail page, choose the **Permissions** tab.
- 6. Under **Bucket policy**, choose **Edit** and do the following to modify your S3 bucket policy:
  - a. Update what's in the editor to include the following policy statements:

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "DataSyncCreateS3LocationAndTaskAccess",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::source-account:role/source-datasync-role"
      },
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads",
        "s3:AbortMultipartUpload",
        "s3:DeleteObject",
        "s3:GetObject",
        "s3:ListMultipartUploadParts",
        "s3:PutObject",
        "s3:GetObjectTagging",
        "s3:PutObjectTagging"
     ],
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-destination-bucket",
        "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
      ]
    }
 ]
}
```

- b. Replace each instance of *source-account* with the AWS account ID for your source account.
- c. Replace *source-datasync-role* with the <u>IAM role that you created for DataSync in</u> your source account.

d. Replace each instance of <a href="mailto:amzn-s3-demo-destination-bucket">amzn-s3-demo-destination-bucket</a> with the name of the S3 bucket in your destination account.

#### 7. Choose **Save changes**.

## Step 4: In your destination account, disable ACLs for your S3 bucket

It's important that all the data that you copy to the S3 bucket belongs to your destination account. To ensure that this account owns the data, disable the bucket's access control lists (ACLs). For more information, see <a href="Controlling ownership of objects and disabling ACLs for your bucket">Controlling ownership of objects and disabling ACLs for your bucket</a> in the Amazon S3 User Guide.

#### To disable ACLs for your destination bucket

- 1. While still logged in to the S3 console with your destination account, choose the S3 bucket that you're transferring data to.
- 2. On the bucket's detail page, choose the **Permissions** tab.
- 3. Under **Object Ownership**, choose **Edit**.
- 4. If it isn't already selected, choose the **ACLs disabled (recommended)** option.
- 5. Choose **Save changes**.

# Step 5: In your source account, create a DataSync source location for your on-premises storage

In your source account, create a <u>DataSync source location</u> for the on-premises storage system that you're transferring data from. This location uses the <u>agent that you activated</u> in your source account.

# Step 6: In your source account, create a DataSync destination location for your S3 bucket

While still in your source account, create a location for the S3 bucket that you're transferring data to.

**Before you begin**: Make sure that you have the <u>required permissions for your source account</u>.

Since you can't create cross-account locations by using the DataSync console interface, these instructions require that you run a create-location-s3 command to create your destination location. We recommend running the command by using AWS CloudShell, a browser-based, preauthenticated shell that you launch directly from the console. CloudShell allows you to run AWS CLI commands like create-location-s3 without downloading or installing command line tools.



#### Note

To complete the following steps by using a command line tool other than CloudShell, make sure that your AWS CLI profile uses the same IAM role that includes the required user permissions to use DataSync in your source account.

## To create a DataSync destination location by using CloudShell

- While still in your source account, do one of the following to launch CloudShell from the console:
  - Choose the CloudShell icon on the console navigation bar. It's located to the right of the search box.
  - Use the search box on the console navigation bar to search for CloudShell and then choose the **CloudShell** option.
- Copy the following command:

```
aws datasync create-location-s3 \
  --s3-bucket-arn arn:aws:s3:::amzn-s3-demo-destination-bucket \
  --s3-config '{
    "BucketAccessRoleArn":"arn:aws:iam::source-user-account:role/source-datasync-
role"
  }'
```

- Replace amzn-s3-demo-destination-bucket with the name of the S3 bucket in your destination account.
- Replace source-user-account with the AWS account ID for your source account. 4.
- Replace source-datasync-role with the DataSync IAM role that you created in your source account.
- Run the command in CloudShell. 6.

If the command returns a DataSync location ARN similar to this, you successfully created the location:

```
{
  "LocationArn": "arn:aws:datasync:us-east-2:123456789012:location/loc-
abcdef01234567890"
}
```

7. In the left navigation pane, expand **Data transfer**, then choose **Locations**.

From your source account, you can see the S3 location that you just created for your destination account bucket.

## Step 7: In your source account, create and start your DataSync task

Before starting a DataSync task to transfer your data, let's recap what you've done so far:

- In your source account, you created your DataSync agent. The agent can read from your onpremises storage system and communicate with the DataSync service.
- In your source account, you created an IAM role that allows DataSync to transfer data to the S3 bucket in your destination account.
- In your destination account, you configured your S3 bucket so that DataSync can transfer data to it.
- In your source account, you created the DataSync source and destination locations for your transfer.

## To create and start the DataSync task

- While still using the DataSync console in your source account, expand Data transfer in the left navigation pane, then choose Tasks and Create task.
- 2. On the **Configure source location** page, choose **Choose an existing location**. Choose the source location that you're copying data from (your on-premises storage) then **Next**.
- On the Configure destination location page, choose Choose an existing location. Choose the
  destination location that you're copying data to (the S3 bucket in your destination account)
  then Next.

On the **Configure settings** page, give the task a name. As needed, configure additional settings, such as specifying an Amazon CloudWatch log group. Choose **Next**.

- 5. On the **Review** page, review your settings and choose **Create task**.
- On the task's details page, choose **Start**, and then choose one of the following:
  - To run the task without modification, choose Start with defaults.
  - To modify the task before running it, choose Start with overriding options.

When your task finishes, check the S3 bucket in your destination account. You should see the data that moved from your source location.

## **Related resources**

For more information about what you did in this tutorial, see the following topics:

- Creating a role for an AWS service (console)
- Modifying a role trust policy (console)
- Adding a bucket policy by using the Amazon S3 console
- Create an S3 location with the AWS CLI

## Tutorial: Transferring data between Amazon S3 buckets across **AWS** accounts

With AWS DataSync, you can transfer data between Amazon S3 buckets that belong to different AWS accounts.



#### Important

Transferring data across AWS accounts using the methods in this tutorial works only with Amazon S3. Additionally, this tutorial can help you transfer data between S3 buckets that are also in different AWS Regions.

Related resources 416

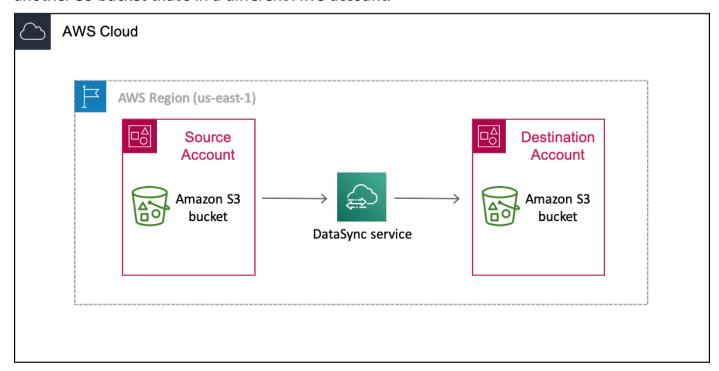
## **Overview**

It's not uncommon to transfer data between AWS accounts, especially if you have separate teams managing your organization's resources. Here's what a cross-account transfer using DataSync can look like:

- **Source account**: The AWS account for managing the S3 bucket that you need to transfer data from.
- **Destination account**: The AWS account for managing the S3 bucket that you need to transfer data to.

#### Transfers across accounts

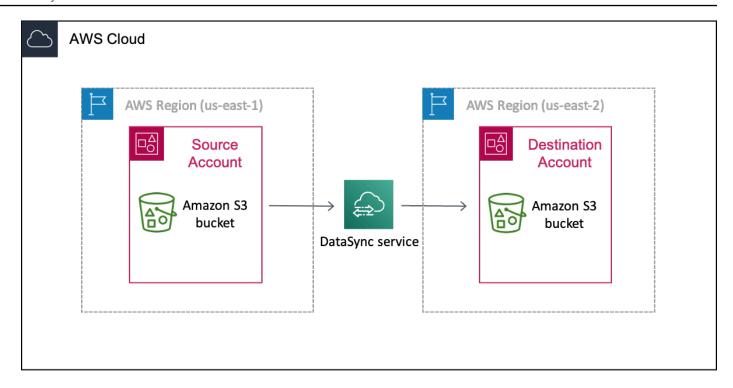
The following diagram illustrates a scenario where you transfer data from an S3 bucket to another S3 bucket that's in a different AWS account.



## Transfers across accounts and Regions

The following diagram illustrates a scenario where you transfer data from an S3 bucket to another S3 bucket that's in a different AWS account and Region.

Overview 417



## **Prerequisite: Required source account permissions**

For your source AWS account, there are two sets of permissions to consider with this kind of cross-account transfer:

- *User permissions* that allow a user to work with DataSync (this might be you or your storage administrator). These permissions let you create DataSync locations and tasks.
- DataSync service permissions that allow DataSync to transfer data to your destination account bucket.

#### User permissions for your source account

In your source account, add at least the following permissions to an IAM role for creating your DataSync locations and task. For information on how to add permissions to a role, see <u>creating</u> or <u>modifying</u> an IAM role.

**JSON** 

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
{
            "Sid": "SourceUserRolePermissions",
            "Effect": "Allow",
            "Action": [
                "datasync:CreateLocationS3",
                "datasync:CreateTask",
                "datasync:DescribeLocation*",
                "datasync:DescribeTaskExecution",
                "datasync:ListLocations",
                "datasync:ListTaskExecutions",
                "datasync:DescribeTask",
                "datasync:CancelTaskExecution",
                "datasync:ListTasks",
                "datasync:StartTaskExecution",
                "iam:CreateRole",
                "iam:CreatePolicy",
                "iam:AttachRolePolicy",
                "iam:ListRoles",
                "s3:GetBucketLocation",
                "s3:ListAllMyBuckets"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                         "datasync.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```



### (i) Tip

To set up your user permissions, consider using AWSDataSyncFullAccess. This is an AWS managed policy that provides a user full access to DataSync and minimal access to its dependencies.

### DataSync service permissions for your source account

The DataSync service needs the following permissions in your source account to transfer data to your destination account bucket.

Later in this tutorial, you add these permissions when creating an IAM role for DataSync. You also specify this role (source-datasync-role) in your destination bucket policy and when creating your DataSync destination location.

**JSON** 

```
"Version": "2012-10-17",
"Statement": [
 {
    "Action": [
      "s3:GetBucketLocation",
      "s3:ListBucket",
      "s3:ListBucketMultipartUploads"
   ],
    "Effect": "Allow",
    "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket"
 },
  {
    "Action": [
      "s3:AbortMultipartUpload",
      "s3:DeleteObject",
      "s3:GetObject",
      "s3:ListMultipartUploadParts",
      "s3:PutObject",
      "s3:GetObjectTagging",
      "s3:PutObjectTagging"
    ],
    "Effect": "Allow",
```

```
"Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
}
]
```

# Prerequisite: Required destination account permissions

In your destination account, your *user permissions* must allow you to update your destination bucket's policy and disable its access control lists (ACLs). For more information on these specific permissions, see the *Amazon S3 User Guide*.

# Step 1: In your source account, create a DataSync IAM role for destination bucket access

In your source AWS account, you need an IAM role that gives DataSync the permissions to transfer data to your destination account bucket.

Since you're transferring across accounts, you must create the role manually. (DataSync can create this role for you in the console when transferring in the same account.)

# Create the DataSync IAM role

Create an IAM role with DataSync as the trusted entity.

- 1. Log in to the AWS Management Console with your source account.
- 2. Open the IAM console at https://console.aws.amazon.com/iam/.
- 3. In the left navigation pane, under **Access management**, choose **Roles**, and then choose **Create role**.
- 4. On the Select trusted entity page, for Trusted entity type, choose AWS service.
- 5. For **Use case**, choose **DataSync** in the dropdown list and select **DataSync**. Choose **Next**.
- 6. On the **Add permissions** page, choose **Next**.
- 7. Give your role a name and choose **Create role**.

For more information, see Creating a role for an AWS service (console) in the IAM User Guide.

### Add permissions to the DataSync IAM role

The IAM role that you just created needs the permissions that allow DataSync to transfer data to the S3 bucket in your destination account.

- On the **Roles** page of the IAM console, search for the role that you just created and choose its 1. name.
- On the role's details page, choose the **Permissions** tab. Choose **Add permissions** then **Create** inline policy.
- 3. Choose the **JSON** tab and do the following:
  - Paste the following JSON into the policy editor:



### Note

The value for aws: ResourceAccount should be the account ID that owns the Amazon S3 bucket specified in the policy.

**JSON** 

```
"Version": "2012-10-17",
"Statement": [
    {
        "Action": [
            "s3:GetBucketLocation",
            "s3:ListBucket",
            "s3:ListBucketMultipartUploads"
        ],
        "Effect": "Allow",
        "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket",
        "Condition": {
            "StringEquals": {
             "aws:ResourceAccount": "123456789012"
            }
        }
    },
    {
        "Action": [
```

```
"s3:AbortMultipartUpload",
              "s3:DeleteObject",
             "s3:GetObject",
             "s3:GetObjectTagging",
             "s3:GetObjectVersion",
             "s3:GetObjectVersionTagging",
             "s3:ListMultipartUploadParts",
             "s3:PutObject",
              "s3:PutObjectTagging"
           ],
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*",
         "Condition": {
              "StringEquals": {
                  "aws:ResourceAccount": "123456789012"
             }
         }
     }
 ]
}
```

- b. Replace each instance of *amzn-s3-demo-destination-bucket* with the name of the S3 bucket in your destination account.
- 4. Choose **Next**. Give your policy a name and choose **Create policy**.

# Step 2: In your destination account, update your S3 bucket policy

In your destination account, modify the destination S3 bucket policy to include the <u>DataSync IAM</u> role that you created in your source account.

**Before you begin**: Make sure that you have the <u>required permissions for your destination account</u>.

# Update your destination S3 bucket policy

- 1. In the AWS Management Console, switch to your destination account.
- 2. Open the Amazon S3 console at <a href="https://console.aws.amazon.com/s3/">https://console.aws.amazon.com/s3/</a>.
- 3. In the left navigation pane, choose **Buckets**.
- 4. In the **Buckets** list, choose the S3 bucket that you're transferring data to.
- 5. On the bucket's detail page, choose the **Permissions** tab.
- 6. Under Bucket policy, choose Edit and do the following to modify your S3 bucket policy:

a. Update what's in the editor to include the following policy statements:

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "DataSyncCreateS3LocationAndTaskAccess",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::source-account:role/source-datasync-role"
     },
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads",
        "s3:AbortMultipartUpload",
        "s3:DeleteObject",
        "s3:GetObject",
        "s3:ListMultipartUploadParts",
        "s3:PutObject",
        "s3:GetObjectTagging",
        "s3:PutObjectTagging"
     ],
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-destination-bucket",
        "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
    }
  ]
}
```

- b. Replace each instance of *source-account* with the AWS account ID for your source account.
- c. Replace source-datasync-role with the IAM role that you created for DataSync in your source account.
- d. Replace each instance of amzn-s3-demo-destination-bucket with the name of the S3 bucket in your destination account.
- 7. Choose **Save changes**.

# Step 3: In your destination account, disable ACLs for your S3 bucket

It's important that all the data that you transfer to the S3 bucket belongs to your destination account. To ensure that this account owns the data, disable the bucket's access control lists (ACLs). For more information, see <a href="Controlling ownership of objects and disabling ACLs for your bucket">Controlling ownership of objects and disabling ACLs for your bucket</a> in the Amazon S3 User Guide.

Before you begin: Make sure that you have the required permissions for your destination account.

### Disable your destination S3 bucket ACLs

- 1. While still logged in to the S3 console with your destination account, choose the S3 bucket that you're transferring data to.
- 2. On the bucket's detail page, choose the **Permissions** tab.
- 3. Under **Object Ownership**, choose **Edit**.
- 4. If it isn't already selected, choose the **ACLs disabled (recommended)** option.
- 5. Choose **Save changes**.

# Step 4: In your source account, create your DataSync locations

In your source account, create the DataSync locations for your source and destination S3 buckets.

Before you begin: Make sure that you have the required permissions for your source account.

### Create your DataSync source location

• In your source account, create a location for the S3 bucket that you're transferring data from.

### **Create your DataSync destination location**

While still in your source account, create a location for the S3 bucket that you're transferring data to.

Since you can't create cross-account locations by using the DataSync console interface, these instructions require that you run a create-location-s3 command to create your destination location. We recommend running the command by using AWS CloudShell, a browser-based, preauthenticated shell that you launch directly from the console. CloudShell allows you to run AWS CLI commands like create-location-s3 without downloading or installing command line tools.



### Note

To complete the following steps by using a command line tool other than CloudShell, make sure that your AWS CLI profile uses the same IAM role that includes the required user permissions to use DataSync in your source account.

### To create a DataSync destination location by using CloudShell

- While still in your source account, do one of the following to launch CloudShell from the console:
  - Choose the CloudShell icon on the console navigation bar. It's located to the right of the search box.
  - Use the search box on the console navigation bar to search for **CloudShell** and then choose the **CloudShell** option.
- Copy the following create-location-s3 command:

```
aws datasync create-location-s3 \
  --s3-bucket-arn arn:aws:s3:::amzn-s3-demo-destination-bucket \
  --region amzn-s3-demo-destination-bucket-region \
  --s3-config '{
    "BucketAccessRoleArn": "arn:aws:iam::source-account-id:role/source-datasync-
role"
  }'
```

- Replace amzn-s3-demo-destination-bucket with the name of the S3 bucket in your destination account.
- 4. If your destination bucket is in a different Region than your source bucket, replace amzn-s3demo-destination-bucket-region with the Region where the destination bucket resides (for example, us-east-2). Remove this option if your buckets are in the same Region.
- 5. Replace *source-account-id* with the source AWS account ID.
- Replace source-datasync-role with the DataSync IAM role that you created in your source account.
- Run the command in CloudShell.

If the command returns a DataSync location ARN similar to this, you successfully created the location:

```
{
  "LocationArn": "arn:aws:datasync:us-east-2:123456789012:location/loc-
abcdef01234567890"
}
```

- In the left navigation pane, expand **Data transfer**, then choose **Locations**. 8.
- If you created the location in a different Region, choose that Region in the navigation pane. 9.

From your source account, you can see the S3 location that you just created for your destination account bucket.

# Step 5: In your source account, create and start your DataSync task

Before starting a DataSync task to transfer your data, let's recap what you've done so far:

- In your source account, you created an IAM role that allows DataSync to transfer data to the S3 bucket in your destination account.
- In your destination account, you configured your S3 bucket so that DataSync can transfer data to it.
- In your source account, you created the DataSync source and destination locations for your transfer.

# Create and start your DataSync task

- While still using the DataSync console in your source account, expand Data transfer in the left navigation pane, then choose **Tasks** and **Create task**.
- If the bucket in your destination account is in a different Region than the bucket in your source account, choose the destination bucket's Region in the top navigation pane.

# Important

To avoid a network connection error, you must create your DataSync task in the same Region as the destination location.

- On the **Configure source location** page, do the following: 3.
  - Select **Choose an existing location**.

(For transfers across Regions) In the **Region** dropdown, choose the Region where the source bucket resides.

- For **Existing locations**, choose the source location for the S3 bucket that you're transferring data from, then choose Next.
- On the **Configure destination location** page, do the following: 4.
  - Select **Choose an existing location**.
  - For **Existing locations**, choose the destination location for the S3 bucket that you're transferring data to, then choose **Next**.
- On the **Configure settings** page, choose a **Task mode**.



## (i) Tip

We recommend using **Enhanced** mode. For more information, see Choosing a task mode for your data transfer.

- Give the task a name and configure additional settings, such as specifying an Amazon CloudWatch log group. Choose Next.
- On the **Review** page, review your settings and choose **Create task**. 7.
- On the task's details page, choose **Start**, and then choose one of the following:
  - To run the task without modification, choose Start with defaults.
  - To modify the task before running it, choose Start with overriding options.

When your task finishes, check the S3 bucket in your destination account. You should see the data that moved from your source account bucket.

# **Troubleshooting**

Refer to the following information if you run into issues trying to complete your cross-account transfer.

### **Connection errors**

When transferring between S3 buckets in different AWS accounts and Regions with Basic mode tasks, you might get a network connection error when starting your DataSync task. To resolve

Troubleshooting 428

this, use an Enhanced mode task. Alternatively, create a Basic mode task in the same Region as your destination location and try running that task.

# Related: Cross-account transfers with S3 buckets using server-side encryption

If you're trying to do this transfer with S3 buckets using server-side encryption, see the <u>AWS</u> Storage Blog for instructions.

# Performing a large data migration with AWS DataSync

Large-scale data migrations can involve transferring significant volumes of data that encompass millions of files or objects in various formats. AWS DataSync simplifies these complex transfers by managing scheduling, monitoring, encryption, and data verification.

# What is a large data migration?

A large data migration typically involves transferring terabytes or more of data spread across various sources to a new destination storage environment (in this case, AWS). These migrations require careful planning and coordination within your organization to move data successfully while minimizing business disruption.

DataSync can simplify these migrations, which are usually complex in nature. Some benefits of using DataSync for your migration include:

- Automated management of data-transfer processes and the infrastructure required for high performance and secure data transfers.
- End-to-end security, including encryption and data integrity validation, to help ensure that your data arrives securely, intact, and ready to use.
- A purpose-built network protocol and a parallel, multi-threaded architecture to speed up migrations.

# Key stages of a large data migration

You can usually break down a large migration into the following stages:

- (Stage 1) Planning your data migration At this stage, you're trying to understand why you're migrating and what sort of data you're working with. Planning activities include:
  - Understanding why you want to migrate
  - Assembling a team to help you with all aspects of the migration.
  - Identifying data locations, formats, and usage patterns
  - Assessing available hardware resources and network requirements (if you're migrating from an on-premises data center)

 Running proof of concept (POC) tests with DataSync to estimate migration timelines, plan cutover windows, and get a sense of how you need to configure DataSync

- (Stage 2) Implementing your large data migration At this point, you're validating your plan and starting the migration. Implementation activities include:
  - · Validating the migration plan
  - Executing phased cutovers that include monitoring and verifying your data transfers as expected
  - Optimizing and adjusting as needed in between each cutover
  - Cleaning up unused resources once you're done

# **Additional resources**

AWS Prescriptive Guidance has the following resources that can help you plan and implement a large migration. Use this guide to understand how DataSync can work in the context of common migration processes and activities.

- Large migrations to the AWS cloud
- Strategy and best practices for AWS large migrations
- Migrate shared file systems in an AWS large migration This resource includes an SFS Discovery-Workbook that you can download and use to plan a migration at the file share level.

# Stage 1: Planning your large data migration

Planning is essential when migrating a large dataset. You must understand the data you're migrating, your motivations for the migration, and how AWS DataSync can help you get your data where you want it.

### **Topics**

- Gathering requirements for your migration
- Running a DataSync proof of concept
- Estimating migration timelines

Additional resources 431

# Gathering requirements for your migration

The first step in a large data migration requires collecting a variety of information across your organization.

This information helps you create a migration <u>process</u>, which for large migrations can include multiple transfers and procedures for cutting over operations (done in <u>waves</u>) from your source to your destination storage.

# Understanding why you want to migrate

Before you can start migrating to AWS, you need to clearly understand why you're migrating your data. This helps address common migration challenges such as meeting deadlines, managing resources, and coordinating across teams.

If you need help determining your motivations for the migration, answer these questions:

- Are you freeing up on-premises storage space?
- Are you meeting hardware support contract deadlines?
- Is this for a data center exit?
- What's your migration timeline?
- Are you transferring data from other cloud storage?
- Are you migrating partial or complete datasets?
- Is this for data archival?
- Do applications or users need regular access to this data?

# Figuring out logistics

Address some basic logistics about your storage environment, the migration, and your organization:

- 1. Get a basic understanding of your current data storage infrastructure.
- 2. Verify whether you need a <u>DataSync agent</u>. For example, you need an agent if you're transferring from on-premises storage.
- 3. If you need an agent, make sure that you understand the agent requirements:

 An agent can run as a virtual machine (VM) on VMware ESXi, Linux Kernel-based Virtual Machine (KVM), and Microsoft Hyper-V hypervisors. You also can deploy an agent as an Amazon EC2 instance within AWS.

- Large migrations are typically memory intensive. Make sure that your agent has enough RAM.
- 4. Identify key stakeholders from your leadership, networking, storage, and IT departments who need to be involved in the migration. This can include:
  - Find a single-threaded leader who's dedicated to the project and its results.
  - Determine who's responsible for the ownership and classification of the data that you're migrating.
  - Identify who manages your source and who eventually will manage the AWS storage service that you're migrating to.
  - Find out who will create and manage any other processes for your data once it's in AWS.
- 5. Establish cross-department communication channels.
- 6. Create a rollback plan for contingencies.
- 7. Document the complete migration process, including waves, validation, and cutover procedures. Use this as your runbook for the entire migration. You will update this process as you plan and implement the migration.

# Reviewing the data you're migrating

Work with your storage and application teams to analyze the characteristics of the data you're migrating. This information helps you determine a migration strategy that you can execute with DataSync.

### **Contents**

- Determining data usage patterns
- Identifying data structure and layout
- Documenting shares and folders
- Analyzing file sizes

### **Determining data usage patterns**

• For actively used data with frequent modifications, plan for multiple waves of incremental transfers to avoid disrupting business operations.

- For read-only data that might be considered archival, you might not need to plan for waves.
- If you have a mix of data usage patterns, plan waves that migrate these different datasets separately. For example, you might have one wave for archive data, with the rest of the waves dedicated to migrating active data.

### Identifying data structure and layout

- Determine if data is organized by time periods (year, month, day) or other patterns.
- Use this organization structure to plan your migration waves. For example, you might migrate a year's worth of archive data during one wave.

### **Documenting shares and folders**

- Create an inventory of shares and folders (including file or object counts for each).
- Identify shares and folders with active datasets. These might require incremental transfers during the migration.
- Review the <u>DataSync quotas</u>. This can help you plan how to partition your dataset when configuring DataSync.

### **Analyzing file sizes**

- Expect higher data throughput for transfers with larger files (MB or GB) compared to smaller files (KB).
- If you're working with a lot of smaller files, expect more metadata operations on your storage system and lower data throughput. DataSync performs these operations when comparing and verifying your source and destination locations.

# Identifying storage requirements

To choose a compatible AWS storage service to migrate your data, you need to evaluate your source storage system's characteristics and performance.

This information can also help you <u>schedule your transfers</u> to minimize impact on business operations during the migration.

### **Contents**

- Determining source storage support
- Reviewing metadata preservation requirements
- Collecting performance metrics from source storage
- Choosing a destination AWS storage service

### **Determining source storage support**

DataSync can work with a variety of storage systems that allow access through NFS, SMB, HDFS, and S3 compatible object storage clients.

If you're migrating from other cloud storage, verify that DataSync can work with that provider. For a list of supported source locations, see Where can I transfer my data with AWS DataSync?

### Reviewing metadata preservation requirements

DataSync can preserve your file or object metadata during a transfer. How your metadata gets preserved depends on your transfer locations and if those locations use similar types of metadata.

DataSync in some cases needs additional permissions to preserve file metadata, such as NTFS discretionary access lists (DACLs).

For more information, see <u>Understanding how DataSync handles file and object metadata</u>.

# Collecting performance metrics from source storage

Measure baseline IOPS and disk throughput during average and peak workloads for your source storage. Transferring data adds I/O overhead to both your source and destination storage systems.

Compare this performance data against your storage system's specifications to determine available performance resources.

## Choosing a destination AWS storage service

At this point, you might have an idea what AWS storage service makes sense for your data. If not, data usage patterns and storage performance are a couple areas to think about when deciding. For

example, you might consider Amazon S3 if you have archive data and Amazon FSx or Amazon EFS for active data.

To help you decide the right object or file-based storage for your data, see <u>Choosing an AWS</u> storage service.

# **Determining network requirements**

To migrate your data with DataSync, you must establish network connections between your source storage, agent, and AWS. You also need to plan for enough network bandwidth and infrastructure.

Work with your network engineers and storage administrators to gather the following network requirements.

### **Contents**

- · Assessing your available network bandwidth
- Considering options for connecting your network to AWS
- Choosing a service endpoint for agent communication
- Planning for enough network infrastructure

# Assessing your available network bandwidth

Your available network bandwidth factors into your transfer speeds and overall migration time. If you're transferring from an on-premises storage system, do the following:

- Work with your network team to determine average and peak bandwidth utilization.
- Identify windows when you can transfer data and avoid disrupting daily operations. This will inform when your migration waves and cutovers happen.

You can control how much bandwidth DataSync uses. For more information, see <u>Setting bandwidth</u> <u>limits for your AWS DataSync task</u>.

Since transfers from other cloud storage typically happen over the public internet, there usually are less bandwidth restrictions and considerations with these transfers.

### Considering options for connecting your network to AWS

Consider the following options for establishing network connectivity for your DataSync transfer:

• **AWS Direct Connect** - Review the <u>architecture and routing examples</u> for using Direct Connect with DataSync. You can monitor Direct Connect activity using Amazon CloudWatch.

- VPN AWS Site-to-Site VPN offers up to 1.25 Gbps throughput per tunnel.
- Public internet Contact with your internet service provider for network usage data.

### Choosing a service endpoint for agent communication

DataSync agents use <u>service endpoints</u> to communicate with the DataSync service. The type of endpoint you use depends on the how you're connecting for your network to AWS.

### Planning for enough network infrastructure

For every transfer task that you create, DataSync automatically generates and manages the network infrastructure for your data transfers. This infrastructure is known as *network interfaces* or *elastic network interfaces*, which are logical networking components in an Amazon virtual private cloud (VPC) that represent virtual network cards. For more information, see the <u>Amazon EC2 User Guide</u>.

Each network interface uses a single IP address in your destination VPC subnet. To make sure that you have enough network infrastructure for your migration, do the following:

- Note the number of <u>network interfaces</u> that DataSync will create for your DataSync destination location.
- Make sure that your subnet has enough IP addresses for your DataSync tasks. For example, a task
  that uses an agent requires four IP addresses. If you create four tasks for your migration, that
  means you need 16 available IP addresses in your subnet.

# Running a DataSync proof of concept

Running a proof of concept (POC) with AWS DataSync helps you validate the following aspects of your data migration planning:

- Verify network connectivity between source and destination locations.
- Validate your initial DataSync task configuration.
- Measure data transfer performance.
- Estimate migration timelines.

Running a proof of concept 437

• Define success criteria with the key stakeholders working on the migration.

# Getting started with your proof of concept

- 1. Create your DataSync agent:
  - 1. Deploy your agent.
  - 2. Choose a service endpoint for your agent.
  - 3. Activate your agent.
  - 4. Verify your agent's network connections.
- 2. Select a small subset of data that represents the data that you're migrating.

For example, if your source storage has a mix of large and small files, the subset of data you transfer in your POC should reflect that. This gives you a preliminary understanding of performance from the storage systems, your network, and DataSync.

- 3. Create a DataSync source location for your on-premises or other cloud storage system.
- 4. Create a DataSync destination location for your AWS storage service.
- 5. Create a DataSync transfer task with a filter that only transfers your data subset.
- 6. Start your DataSync task.
- 7. Collect transfer performance metrics by monitoring the following:
  - Your task execution's data and file throughput. You can do this through the DataSync console or the <a href="DescribeTaskExecution">DescribeTaskExecution</a> operation. If you use DescribeTaskExecution, here's how you calculate these metrics:
    - Data throughput: Divide BytesWritten by TransferDuration
    - File throughput: Divide FilesTransferred by TransferDuration
  - Source and destination storage utilization. Work closely with your storage administrators to get this information.
  - Network usage.
- 8. Verify the transferred data at your destination location:
  - Review your CloudWatch logs for task execution errors.
  - Verify that permissions and metadata are preserved at the destination location.

Confirm that applications and users can access destination data as expected.

Running a proof of concept 438

Address any issues that you encounter. For more information, see <u>Troubleshooting AWS</u>
 DataSync issues.

9. Run your task a few more times to get an idea how long it takes DataSync to prepare, transfer, and verify your data. (For more information, see Task execution statuses.)

If you run a task more than once, DataSync by default performs an incremental transfer and copies only the data that's changed from the previous task run.

While the transfer time will likely be shorter for incremental transfers, DataSync will always prepare your transfer the same way by scanning and comparing your locations to identify what to transfer. You can use these preparation times to <u>estimate cutover timelines</u> for your migration.

10. If needed, update your migration plan based on what you learned during the POC.

# **Estimating migration timelines**

Using the information you've collected to this point, you can estimate how long the migration will take using AWS DataSync.

# **Estimating data transfer timelines**

You can estimate how long it takes DataSync to transfer your data based on the following information you collected during migration requirements gathering and your DataSync proof of concept (POC):

- Your available network bandwidth
- Source and destination storage utilization metrics
- Performance metrics from your DataSync POC

### To estimate a data transfer timeline

- 1. Compare the data and file throughput from your POC with your available network bandwidth.
- 2. If your throughput is lower than your available bandwidth (such as 300 MiB/s for throughput with 10 Gbps of network bandwidth), consider partitioning your dataset into multiple tasks to maximize bandwidth usage.

DataSync has a few options for partitioning your dataset. For more information, see Accelerating your migration with data partitioning.

3. Calculate how many days a transfer takes by using the following formula, which provides a theoretical minimum transfer time:

```
(DATA_SIZE * 8 bits per byte)/(CIRCUIT * NETWORK_UTILIZATION percentage * 3600 seconds per hour * AVAILABLE_HOURS) = Number of days
```

When using this formula, replace the following with your own values:

- DATA\_SIZE: The amount of data that you're migrating (expressed in bytes).
- CIRCUIT: Your available network bandwidth (expressed in bits per second).
- NETWORK\_UTILIZATION: What percent of your network is being used.
- AVAILABLE\_HOURS: The number of operational hours available in each day.

For example, you would calculate a migration with 100 TB of data, a 1 Gbps internet connection, 80 percent network utilization, and 24 hours per day availability like this:

```
(100,000,000,000,000 bytes * 8) / (1,000,000,000 bps * 0.80 * 3600 * 24) = 11.57 days
```

In this case, the migration would take almost 12 days before accounting for real-world conditions.

- 4. Adjust your calculated transfer duration to account for real-world conditions:
  - Network performance fluctuations
  - Storage performance variations
  - Downtime between migration waves

# **Estimating cutover timelines**

If you're migrating active datasets, you likely need cutovers so that you don't disrupt business operations.

Don't underestimate how long cutovers take. With large migrations, it's not uncommon for cutover activities to take up to 30 percent of your overall migration time.

1. Evaluate if you need to perform cutovers in waves to reduce the amount of data scanned for incremental changes.

- One strategy for doing this is cutting over datasets that you partition based on shares, folders, or storage systems.
- 2. Review how long it generally took DataSync to prepare, transfer, and verify your data during the POC.
  - Note in particular the prepare durations of your task executions. To find this information, run the <a href="DescribeTaskExecution">DescribeTaskExecution</a> operation, then check the value of <a href="PrepareDuration">PrepareDuration</a> for the duration time (in milliseconds).
- 3. Estimate how long a cutover might take by measuring the time delta across parallel tasks.
  - For more information on parallel tasks, see Accelerating your migration with data partitioning.
- 4. Use your cutover estimation to schedule your cutovers. These essentially are maintenance windows when your source data can't be modified.

## **Next steps**

After estimating your timelines, you're ready to start implementing your migration.

# Stage 2: Implementing your large data migration

With the information you gathered during planning, you can begin using AWS DataSync to migrate to your new storage system. If you haven't already, we recommend reviewing the <u>AWS Prescriptive</u> Guidance resources for large migrations.

### **Topics**

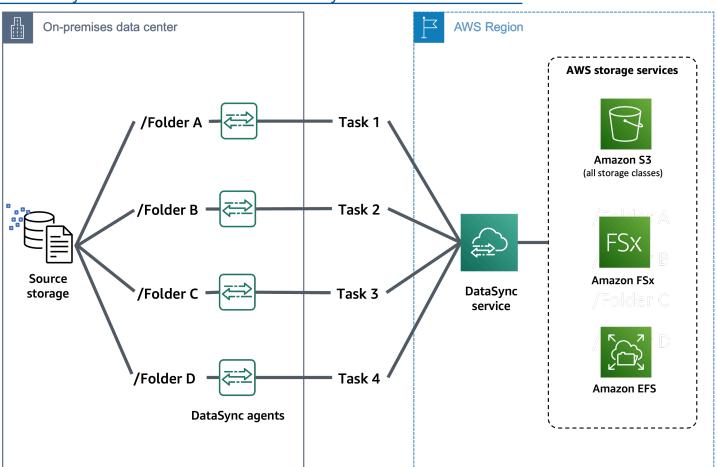
- · Accelerating your migration with data partitioning
- Running your DataSync transfer tasks
- Monitoring your transfers

# Accelerating your migration with data partitioning

With a large migration, we recommend partitioning your dataset with multiple DataSync tasks. Partitioning your source data across multiple tasks (and possibly agents) lets you parallelize your transfers and reduce the migration timeline.

Partitioning also helps you stay within DataSync <u>quotas</u> and simplifies the monitoring and debugging of your tasks.

The following diagram shows how you might use multiple DataSync tasks and agents to transfer data from the same source storage location. In this scenario, each task focuses on a specific folder in the source location. For more information and examples on these approaches, see <a href="How to accelerate your data transfers with AWS DataSync scale out architectures">How to accelerate your data transfers with AWS DataSync scale out architectures</a>.



# Partitioning your dataset by folder or prefix

When creating your DataSync source location, you can specify a folder, directory, or prefix that DataSync reads from. For example, if you're migrating a file share with top-level directories, you can create multiple locations that specify a different directory path. You can then use these locations to run multiple DataSync tasks during your migration.

# Partitioning your dataset with filters

You can apply <u>filters</u> to include or exclude data from your source location in a transfer. In the context of a large migration, filters can help you scope tasks to specific portions of your dataset.

For example, if you're migrating archive data that's organized by year, you can create an include filter to match for a specific year or multiple years. You also can modify the filter each time you run the task to match a different year.

# Partitioning your dataset with manifests

A <u>manifest</u> is a list of files or objects that you want DataSync to transfer. With a manifest, DataSync doesn't have to read everything in a source location to determine what to transfer.

You can create manifests from inventories of your source storage or through event-driven approaches (for example, see <a href="Implementing AWS DataSync with hundreds of millions of objects">Implementing AWS DataSync with hundreds of millions of objects</a>). You can also use a different manifest each time you start a task, allowing you to transfer different sets of data with the same task.

# Running your DataSync transfer tasks

During each of your migration waves, your data transfer usually follows the same general process:

- 1. Run an initial full transfer of your data.
- 2. Verify the data in the destination.
- 3. Run incremental transfers for any data that might have changed since the initial transfer.
- 4. Cut over operations to your destination location.
- 5. Review cutover results.

# **Running your tasks**

You likely will need to run your DataSync transfer tasks during business hours to minimize your overall migration time. It's common in these situations to run an initial full transfer followed by incremental transfers that account for changes to your source location from users and applications.

To avoid network-related issues during business hours, you can limit the amount of bandwidth that your tasks use. For more information, see Setting bandwidth limits for your AWS DataSync task.

- Run an initial full transfer:
  - a. Start your DataSync task (or tasks if you're running tasks in parallel).
  - b. Monitor the progress and performance of your task executions.

c. Verify that your data transferred the way you expect (for example, file metadata is preserved).

- 2. Run incremental transfers:
  - a. Schedule your tasks to run periodically.
  - b. Monitor your task executions and fix errors if encountered.

# Performing a cutover

After your initial and incremental transfers, you can start the process of cutting over operations to your destination location.

- 1. Start the scheduled maintenance window.
- 2. Update your source storage system to be read only for applications and users.
- 3. Run final incremental transfers to copy remaining deltas between your source and destination locations.
- Conduct a thorough data validation (for example, by reviewing CloudWatch logs and <u>task</u> reports).
- 5. Switch your applications and users to the new environment of your destination location.
- 6. Test application functionality and make sure that users can access data in your destination location.
- 7. Schedule a retrospective meeting to review the transfer with the migration teams. Ask the following probing sample questions:
  - Was the cutover successful? If not, what was the issue?
  - Did we use all available bandwidth?
  - Was the source and destination storage fully utilized?
  - Can we get more data throughput with additional tasks?
  - Do we need to plan for a longer maintenance window?
- 8. If needed, update your migration plan before starting the next wave.

# **Monitoring your transfers**

AWS DataSync provides several monitoring options to help you validate and debug your transfer.

Monitoring your transfers 444

# Monitoring your transfers with CloudWatch metrics

You can create custom CloudWatch dashboards with metrics from your DataSync task executions. For more information, see Monitoring data transfers with Amazon CloudWatch metrics.

# Monitoring your transfers with task reports

If you're transferring millions of files or objects, considering using task reports. Task reports provide detailed information about what DataSync attempts to transfer, skip, verify, and delete during a task execution. For more information, see <a href="Monitoring your data transfers with task">Monitoring your data transfers with task</a> reports.

You can also visualize your task reports by using AWS services such as AWS Glue, Amazon Athena, and Amazon QuickSight. For more information, see the AWS Storage Blog.

# Monitoring your transfers with CloudWatch Logs

At minimum, we recommend that you configure your task to log basic information and transfer errors. For more information, see Monitoring data transfers with Amazon CloudWatch Logs.

Monitoring your transfers 445

# **AWS DataSync API**

In addition to the AWS Management Console and AWS CLI, you can use the AWS DataSync API to configure and manage DataSync with the AWS SDKs.

# **Topics**

- Actions
- Data Types
- Common Errors
- Common Parameters

# **Actions**

The following actions are supported:

- CancelTaskExecution
- CreateAgent
- CreateLocationAzureBlob
- CreateLocationEfs
- CreateLocationFsxLustre
- CreateLocationFsxOntap
- CreateLocationFsxOpenZfs
- CreateLocationFsxWindows
- CreateLocationHdfs
- CreateLocationNfs
- CreateLocationObjectStorage
- CreateLocationS3
- CreateLocationSmb
- CreateTask
- DeleteAgent
- DeleteLocation
- DeleteTask

Actions 446

- DescribeAgent
- DescribeLocationAzureBlob
- DescribeLocationEfs
- DescribeLocationFsxLustre
- DescribeLocationFsxOntap
- DescribeLocationFsxOpenZfs
- DescribeLocationFsxWindows
- DescribeLocationHdfs
- DescribeLocationNfs
- DescribeLocationObjectStorage
- DescribeLocationS3
- DescribeLocationSmb
- DescribeTask
- DescribeTaskExecution
- ListAgents
- ListLocations
- ListTagsForResource
- ListTaskExecutions
- ListTasks
- StartTaskExecution
- TagResource
- UntagResource
- UpdateAgent
- UpdateLocationAzureBlob
- UpdateLocationEfs
- UpdateLocationFsxLustre
- UpdateLocationFsxOntap
- UpdateLocationFsxOpenZfs
- UpdateLocationFsxWindows
- UpdateLocationHdfs

Actions 447

- <u>UpdateLocationNfs</u>
- <u>UpdateLocationObjectStorage</u>
- <u>UpdateLocationS3</u>
- UpdateLocationSmb
- UpdateTask
- <u>UpdateTaskExecution</u>

Actions 448

# CancelTaskExecution

Stops an AWS DataSync task execution that's in progress. The transfer of some files are abruptly interrupted. File contents that're transferred to the destination might be incomplete or inconsistent with the source files.

However, if you start a new task execution using the same task and allow it to finish, file content on the destination will be complete and consistent. This applies to other unexpected failures that interrupt a task execution. In all of these cases, DataSync successfully completes the transfer when you start the next task execution.

# **Request Syntax**

```
{
    "<u>TaskExecutionArn</u>": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

# **TaskExecutionArn**

The Amazon Resource Name (ARN) of the task execution to stop.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync: [a-z \\-0-9]*: [0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$
```

Required: Yes

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

CancelTaskExecution 449

### **Errors**

For information about the errors that are common to all actions, see Common Errors.

# InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CancelTaskExecution 450

# **CreateAgent**

Activates an AWS DataSync agent that you deploy in your storage environment. The activation process associates the agent with your AWS account.

If you haven't deployed an agent yet, see Do I need a DataSync agent?

# **Request Syntax**

# **Request Parameters**

For information about the parameters that are common to all actions, see <u>Common Parameters</u>.

The request accepts the following data in JSON format.

# **ActivationKey**

Specifies your DataSync agent's activation key. If you don't have an activation key, see <a href="Activating your agent">Activating your agent</a>.

Type: String

Length Constraints: Maximum length of 29.

Pattern: [A-Z0-9]{5}(-[A-Z0-9]{5}){4}

Required: Yes

### **AgentName**

Specifies a name for your agent. We recommend specifying a name that you can remember.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

Required: No

### **SecurityGroupArns**

Specifies the Amazon Resource Name (ARN) of the security group that allows traffic between your agent and VPC service endpoint. You can only specify one ARN.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*:$   $[0-9]\{12\}:$  security-group/sg-[a-f0-9]+\$

Required: No

### **SubnetArns**

Specifies the ARN of the subnet where your VPC service endpoint is located. You can only specify one ARN.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]$ \*: [0-9]{12}: subnet/.\*\$

Required: No

### **Tags**

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least one tag for your agent.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

# **VpcEndpointId**

Specifies the ID of the VPC service endpoint that you're using. For example, a VPC endpoint ID looks like vpce-01234d5aff67890e1.

### Important

The VPC service endpoint you use must include the DataSync service name (for example, com.amazonaws.us-east-2.datasync).

Type: String

Pattern: ^vpce-[0-9a-f]{17}\$

Required: No

# **Response Syntax**

```
{
   "AgentArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **AgentArn**

The ARN of the agent that you just activated. Use the <u>ListAgents</u> operation to return a list of agents in your AWS account and AWS Region.

```
Type: String
```

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}: agent/agent-[0-9a-z]{17}$
```

### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

# InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

# InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

# **Examples**

## Sample Request

The following example activates a DataSync agent.

```
{
   "ActivationKey": "AAAAA-1AAAA-BB1CC-33333-EEEEE",
   "AgentName": "MyAgent",
   "Tags": [{
        "Key": "Job",
        "Value": "TransferJob-1"
   }]
}
```

#### Sample Response

The response returns the ARN of the activated agent.

```
{
    "AgentArn": "arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44baca3"
}
```

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateAgent 455

# **CreateLocationAzureBlob**

Creates a transfer *location* for a Microsoft Azure Blob Storage container. AWS DataSync can use this location as a transfer source or destination. You can make transfers with or without a <u>DataSync</u> agent that connects to your container.

Before you begin, make sure you know <u>how DataSync accesses Azure Blob Storage</u> and works with <u>access tiers</u> and <u>blob types</u>.

## Request Syntax

```
{
   "AccessTier": "string",
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "BlobType": "string",
   "CmkSecretConfig": {
      "KmsKeyArn": "string",
      "SecretArn": "string"
   },
   "ContainerUrl": "string",
   "CustomSecretConfig": {
      "SecretAccessRoleArn": "string",
      "SecretArn": "string"
   },
   "SasConfiguration": {
      "Token": "string"
   },
   "Subdirectory": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### AccessTier

Specifies the access tier that you want your objects or files transferred into. This only applies when using the location as a transfer destination. For more information, see Access tiers.

Type: String

Valid Values: HOT | COOL | ARCHIVE

Required: No

### **AgentArns**

(Optional) Specifies the Amazon Resource Name (ARN) of the DataSync agent that can connect with your Azure Blob Storage container. If you are setting up an agentless cross-cloud transfer, you do not need to specify a value for this parameter.

You can specify more than one agent. For more information, see Using multiple agents for your transfer.



#### Note

Make sure you configure this parameter correctly when you first create your storage location. You cannot add or remove agents from a storage location after you create it.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z  $-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$ \$

Required: No

# AuthenticationType

Specifies the authentication method DataSync uses to access your Azure Blob Storage. DataSync can access blob storage using a shared access signature (SAS).

Type: String

Valid Values: SAS | NONE

Required: Yes

### BlobType

Specifies the type of blob that you want your objects or files to be when transferring them into Azure Blob Storage. Currently, DataSync only supports moving data into Azure Blob Storage as block blobs. For more information on blob types, see the Azure Blob Storage documentation.

Type: String

Valid Values: BLOCK

Required: No

## CmkSecretConfig

Specifies configuration information for a DataSync-managed secret, which includes the authentication token that DataSync uses to access a specific AzureBlob storage location, with a customer-managed AWS KMS key.

When you include this paramater as part of a CreateLocationAzureBlob request, you provide only the KMS key ARN. DataSync uses this KMS key together with the authentication token you specify for SasConfiguration to create a DataSync-managed secret to store the location access credentials.

Make sure the DataSync has permission to access the KMS key that you specify.



#### Note

You can use either CmkSecretConfig (with SasConfiguration) or CustomSecretConfig (without SasConfiguration) to provide credentials for a CreateLocationAzureBlob request. Do not provide both parameters for the same request.

Type: CmkSecretConfig object

Required: No

# ContainerUrl

Specifies the URL of the Azure Blob Storage container involved in your transfer.

Type: String

Length Constraints: Maximum length of 325.

Pattern:  $^https: \/\[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}\/[a-z0-9](-?[a-z0-9])$ 

z0-9]){2,62}\$

Required: Yes

### CustomSecretConfig

Specifies configuration information for a customer-managed Secrets Manager secret where the authentication token for an AzureBlob storage location is stored in plain text. This configuration includes the secret ARN, and the ARN for an IAM role that provides access to the secret.



You can use either CmkSecretConfig (with SasConfiguration) or CustomSecretConfig (without SasConfiguration) to provide credentials for a CreateLocationAzureBlob request. Do not provide both parameters for the same request.

Type: CustomSecretConfig object

Required: No

# SasConfiguration

Specifies the SAS configuration that allows DataSync to access your Azure Blob Storage.



If you provide an authentication token using SasConfiguration, but do not provide secret configuration details using CmkSecretConfig or CustomSecretConfig, then DataSync stores the token using your AWS account's secrets manager secret.

Type: AzureBlobSasConfiguration object

Required: No

### **Subdirectory**

Specifies path segments if you want to limit your transfer to a virtual directory in your container (for example, /my/images).

Type: String

Length Constraints: Maximum length of 1024.

Pattern:  $^[\p{L}\p{M}\p{Z}\p{N}\p{P}\p{C}]*$$ 

Required: No

## **Tags**

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your transfer location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

# **Response Syntax**

```
{
    "LocationArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

# **LocationArn**

The ARN of the Azure Blob Storage transfer location that you created.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateLocationEfs

Creates a transfer *location* for an Amazon EFS file system. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync accesses Amazon EFS file systems.

## **Request Syntax**

```
{
   "AccessPointArn": "string",
   "Ec2Config": {
      "SecurityGroupArns": [ "string" ],
      "SubnetArn": "string"
   },
   "EfsFilesystemArn": "string",
   "FileSystemAccessRoleArn": "string",
   "InTransitEncryption": "string",
   "Subdirectory": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **AccessPointArn**

Specifies the Amazon Resource Name (ARN) of the access point that DataSync uses to mount your Amazon EFS file system.

For more information, see Accessing restricted file systems.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): elasticfilesystem: <math>[a-z\\-0-9]+:[0-9]\{12\}:access-point/fsap-[0-9a-f]\{8,40\}$
```

Required: No

### **Ec2Config**

Specifies the subnet and security groups DataSync uses to connect to one of your Amazon EFS file system's mount targets.

Type: Ec2Config object

Required: Yes

# **EfsFilesystemArn**

Specifies the ARN for your Amazon EFS file system.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):elasticfilesystem: [a-z\-0-9]*:[0-9]{12}:file-system/fs-.*$
```

Required: Yes

# <u>FileSystemAccessRoleArn</u>

Specifies an AWS Identity and Access Management (IAM) role that allows DataSync to access your Amazon EFS file system.

For information on creating this role, see Creating a DataSync IAM role for file system access.

Type: String

Length Constraints: Maximum length of 2048.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9] {12}:role/.*$
```

Required: No

# InTransitEncryption

Specifies whether you want DataSync to use Transport Layer Security (TLS) 1.2 encryption when it transfers data to or from your Amazon EFS file system.

If you specify an access point using AccessPointArn or an IAM role using FileSystemAccessRoleArn, you must set this parameter to TLS1\_2.

Type: String

Valid Values: NONE | TLS1\_2

Required: No

### **Subdirectory**

Specifies a mount path for your Amazon EFS file system. This is where DataSync reads or writes data on your file system (depending on if this is a source or destination location).

By default, DataSync uses the root directory (or <u>access point</u> if you provide one by using AccessPointArn). You can also include subdirectories using forward slashes (for example, / path/to/folder).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())p{Zs}]*$$ 

Required: No

## Tags

Specifies the key-value pair that represents a tag that you want to add to the resource. The value can be an empty string. This value helps you manage, filter, and search for your resources. We recommend that you create a name tag for your location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

# **Response Syntax**

```
{
    "LocationArn": "string"
```

}

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### LocationArn

The Amazon Resource Name (ARN) of the Amazon EFS file system location that you create.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z ^0-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

# Invalid Request Exception

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

# **Examples**

### **Sample Request**

The following example creates a location for an Amazon EFS file system.

```
{
```

### Sample Request: Creating a location for a restricted Amazon EFS file system

The following example creates a location for an Amazon EFS file system with restricted access. In this kind of scenario, you might have to specify values for AccessPointArn, FileSystemAccessRoleArn, and InTransitEncryption in your request.

```
{
    "AccessPointArn": "arn:aws:elasticfilesystem:us-east-2:111222333444:access-point/
fsap-1234567890abcdef0",
    "Ec2Config": {
        "SubnetArn": "arn:aws:ec2:us-east-2:111222333444:subnet/
subnet-1234567890abcdef1",
        "SecurityGroupArns": [
            "arn:aws:ec2:us-east-2:111222333444:security-group/sg-1234567890abcdef2"
        ]
    },
    "FileSystemAccessRoleArn": "arn:aws:iam::111222333444:role/
AwsDataSyncFullAccessNew",
    "InTransitEncryption": "TLS1_2",
    "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-
abcdef01234567890",
    "LocationUri": "efs://us-east-2.fs-021345abcdef6789/",
    "Subdirectory": "/mount/path",
    "Tags": [{
        "Key": "Name",
        "Value": "ElasticFileSystem-1"
    }]
```

}

### Sample Response

A response returns the location ARN of the Amazon EFS file system.

```
{
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-12abcdef012345678"
}
```

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateLocationFsxLustre

Creates a transfer *location* for an Amazon FSx for Lustre file system. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync accesses FSx for Lustre file systems.

## **Request Syntax**

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

# **FsxFilesystemArn**

Specifies the Amazon Resource Name (ARN) of the FSx for Lustre file system.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): fsx: <math>[a-z\-0-9]*: [0-9]{12}: file-system/fs-.*$
```

Required: Yes

### **SecurityGroupArns**

Specifies the Amazon Resource Names (ARNs) of up to five security groups that provide access to your FSx for Lustre file system.

The security groups must be able to access the file system's ports. The file system must also allow access from the security groups. For information about file system access, see the <u>Amazon</u> FSx for Lustre User Guide.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]{12}: security-group/sq-[a-f0-9]+$
```

Required: Yes

## **Subdirectory**

Specifies a mount path for your FSx for Lustre file system. The path can include subdirectories.

When the location is used as a source, DataSync reads data from the mount path. When the location is used as a destination, DataSync writes data to the mount path. If you don't include this parameter, DataSync uses the file system's root directory (/).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())$p{Zs}]+$$ 

Required: No

## **Tags**

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

## **Response Syntax**

```
{
    "LocationArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### LocationArn

The Amazon Resource Name (ARN) of the FSx for Lustre file system location that you created.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateLocationFsxOntap

Creates a transfer *location* for an Amazon FSx for NetApp ONTAP file system. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync <u>accesses FSx for ONTAP file</u> systems.

## **Request Syntax**

```
{
   "Protocol": {
      "NFS": {
         "MountOptions": {
            "Version": "string"
         }
      },
      "SMB": {
         "Domain": "string",
         "MountOptions": {
            "Version": "string"
         },
         "Password": "string",
         "User": "string"
      }
   },
   "SecurityGroupArns": [ "string" ],
   "StorageVirtualMachineArn": "string",
   "Subdirectory": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **Protocol**

Specifies the data transfer protocol that AWS DataSync uses to access your Amazon FSx file system.

Type: FsxProtocol object

Required: Yes

### **SecurityGroupArns**

Specifies the Amazon EC2 security groups that provide access to your file system's preferred subnet.

The security groups must allow outbound traffic on the following ports (depending on the protocol you use):

- Network File System (NFS): TCP ports 111, 635, and 2049
- Server Message Block (SMB): TCP port 445

Your file system's security groups must also allow inbound traffic on the same ports.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]\{12\}: security-group/sg-[a-f0-9]+$
```

Required: Yes

### **StorageVirtualMachineArn**

Specifies the ARN of the storage virtual machine (SVM) in your file system where you want to copy data to or from.

Type: String

Length Constraints: Maximum length of 162.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): fsx: <math>[a-z\-0-9]+: [0-9]{12}: storage-virtual-machine/fs-[0-9a-f]+/svm-[0-9a-f]{17,}$
```

Required: Yes

### **Subdirectory**

Specifies a path to the file share in the SVM where you want to transfer data to or from.

You can specify a junction path (also known as a mount point), gtree path (for NFS file shares), or share name (for SMB file shares). For example, your mount path might be /vol1, /vol1/ tree1, or /share1.



#### Note

Don't specify a junction path in the SVM's root volume. For more information, see Managing FSx for ONTAP storage virtual machines in the Amazon FSx for NetApp ONTAP User Guide.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^[^\u0000\u0085\u2028\u2029\r\n]{1,255}\$

Required: No

## **Tags**

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

# **Response Syntax**

```
{
   "LocationArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### LocationArn

Specifies the ARN of the FSx for ONTAP file system location that you create.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9]

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

# InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateLocationFsxOpenZfs

Creates a transfer location for an Amazon FSx for OpenZFS file system. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync accesses FSx for OpenZFS file systems.



#### Note

Request parameters related to SMB aren't supported with the CreateLocationFsxOpenZfs operation.

# **Request Syntax**

```
{
   "FsxFilesystemArn": "string",
   "Protocol": {
      "NFS": {
         "MountOptions": {
             "Version": "string"
         }
      },
      "SMB": {
         "Domain": "string",
         "MountOptions": {
            "Version": "string"
         },
         "Password": "string",
         "User": "string"
      }
   "SecurityGroupArns": [ "string" ],
   "Subdirectory": "string",
   "<u>Tags</u>": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
```

}

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### **FsxFilesystemArn**

The Amazon Resource Name (ARN) of the FSx for OpenZFS file system.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): fsx: <math>[a-z\-0-9]*: [0-9]{12}: file-system/fs-.*$
```

Required: Yes

#### **Protocol**

The type of protocol that AWS DataSync uses to access your file system.

Type: FsxProtocol object

Required: Yes

#### **SecurityGroupArns**

The ARNs of the security groups that are used to configure the FSx for OpenZFS file system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]\{12\}: security-group/sg-[a-f0-9]+$
```

Required: Yes

### **Subdirectory**

A subdirectory in the location's path that must begin with /fsx. DataSync uses this subdirectory to read or write data (depending on whether the file system is a source or destination location).

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[^\u0000\u0085\u2028\u2029\r\n]{1,4096}\$

Required: No

### **Tags**

The key-value pair that represents a tag that you want to add to the resource. The value can be an empty string. This value helps you manage, filter, and search for your resources. We recommend that you create a name tag for your location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

# **Response Syntax**

```
{
    "LocationArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### LocationArn

The ARN of the FSx for OpenZFS file system location that you created.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateLocationFsxWindows

Creates a transfer *location* for an Amazon FSx for Windows File Server file system. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync <u>accesses FSx for Windows File</u> Server file systems.

## **Request Syntax**

# **Request Parameters**

For information about the parameters that are common to all actions, see **Common Parameters**.

The request accepts the following data in JSON format.

# **Domain**

Specifies the name of the Windows domain that the FSx for Windows File Server file system belongs to.

If you have multiple Active Directory domains in your environment, configuring this parameter makes sure that DataSync connects to the right file system.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}$ \$

Required: No

## **FsxFilesystemArn**

Specifies the Amazon Resource Name (ARN) for the FSx for Windows File Server file system.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): fsx: <math>[a-z\-0-9]$ \*: [0-9]{12}: file-system/fs-.\*\$

Required: Yes

#### **Password**

Specifies the password of the user with the permissions to mount and access the files, folders, and file metadata in your FSx for Windows File Server file system.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^. {0,104}\$

Required: Yes

# **SecurityGroupArns**

Specifies the ARNs of the Amazon EC2 security groups that provide access to your file system's preferred subnet.

The security groups that you specify must be able to communicate with your file system's security groups. For information about configuring security groups for file system access, see the <u>Amazon FSx for Windows File Server User Guide</u>.

# Note

If you choose a security group that doesn't allow connections from within itself, do one of the following:

• Configure the security group to allow it to communicate within itself.

• Choose a different security group that can communicate with the mount target's security group.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]$ \*: [0-9]{12}: security-group/sg-[a-f0-9]+\$

Required: Yes

# **Subdirectory**

Specifies a mount path for your file system using forward slashes. This is where DataSync reads or writes data (depending on if this is a source or destination location).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-}+\./\(\)\$ p{Zs}]+\$

Required: No

# Tags

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

Type: Array of <u>TagListEntry</u> objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

# User

Specifies the user with the permissions to mount and access the files, folders, and file metadata in your FSx for Windows File Server file system.

For information about choosing a user with the right level of access for your transfer, see required permissions for FSx for Windows File Server locations.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

Required: Yes

# **Response Syntax**

```
{
    "LocationArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## LocationArn

The ARN of the FSx for Windows File Server file system location you created.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateLocationHdfs

Creates a transfer *location* for a Hadoop Distributed File System (HDFS). AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync accesses HDFS clusters.

# **Request Syntax**

```
{
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "BlockSize": number,
   "KerberosKeytab": blob,
   "KerberosKrb5Conf": blob,
   "KerberosPrincipal": "string",
   "KmsKeyProviderUri": "string",
   "NameNodes": [
      {
         "Hostname": "string",
         "Port": number
      }
   ],
   "QopConfiguration": {
      "DataTransferProtection": "string",
      "RpcProtection": "string"
   },
   "ReplicationFactor": number,
   "SimpleUser": "string",
   "Subdirectory": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### AgentArns

The Amazon Resource Names (ARNs) of the DataSync agents that can connect to your HDFS cluster.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z -0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}\$

Required: Yes

## **AuthenticationType**

The type of authentication used to determine the identity of the user.

Type: String

Valid Values: SIMPLE | KERBEROS

Required: Yes

### **BlockSize**

The size of data blocks to write into the HDFS cluster. The block size must be a multiple of 512 bytes. The default block size is 128 mebibytes (MiB).

Type: Integer

Valid Range: Minimum value of 1048576. Maximum value of 1073741824.

Required: No

# KerberosKeytab

The Kerberos key table (keytab) that contains mappings between the defined Kerberos principal and the encrypted keys. You can load the keytab from a file by providing the file's address.



### Note

If KERBEROS is specified for AuthenticationType, this parameter is required.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 65536.

Required: No

#### KerberosKrb5Conf

The krb5.conf file that contains the Kerberos configuration information. You can load the krb5. conf file by providing the file's address. If you're using the AWS CLI, it performs the base64 encoding for you. Otherwise, provide the base64-encoded text.



#### Note

If KERBEROS is specified for AuthenticationType, this parameter is required.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 131072.

Required: No

## KerberosPrincipal

The Kerberos principal with access to the files and folders on the HDFS cluster.



#### Note

If KERBEROS is specified for AuthenticationType, this parameter is required.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^.+\$

Required: No

# KmsKeyProviderUri

The URI of the HDFS cluster's Key Management Server (KMS).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^kms:\/\/http[s]?@(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-]\*[A-Za-z0-9])(;(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-]\*[A-Za-z0-9]))\*:[0-9]{1,5}\/kms\$

Required: No

#### **NameNodes**

The NameNode that manages the HDFS namespace. The NameNode performs operations such as opening, closing, and renaming files and directories. The NameNode contains the information to map blocks of data to the DataNodes. You can use only one NameNode.

Type: Array of HdfsNameNode objects

Array Members: Minimum number of 1 item.

Required: Yes

## QopConfiguration

The Quality of Protection (QOP) configuration specifies the Remote Procedure Call (RPC) and data transfer protection settings configured on the Hadoop Distributed File System (HDFS) cluster. If QopConfiguration isn't specified, RpcProtection and DataTransferProtection default to PRIVACY. If you set RpcProtection or DataTransferProtection, the other parameter assumes the same value.

Type: **QopConfiguration** object

Required: No

# ReplicationFactor

The number of DataNodes to replicate the data to when writing to the HDFS cluster. By default, data is replicated to three DataNodes.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 512.

Required: No

#### SimpleUser

The user name used to identify the client on the host operating system.



#### Note

If SIMPLE is specified for AuthenticationType, this parameter is required.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[\_.A-Za-z0-9][-\_.A-Za-z0-9]\*\$

Required: No

## **Subdirectory**

A subdirectory in the HDFS cluster. This subdirectory is used to read data from or write data to the HDFS cluster. If the subdirectory isn't specified, it will default to /.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())$p{Zs}]+$$ 

Required: No

## **Tags**

The key-value pair that represents the tag that you want to add to the location. The value can be an empty string. We recommend using tags to name your resources.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

## **Response Syntax**

```
""LocationArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### LocationArn

The ARN of the source HDFS cluster location that you create.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9]
```

#### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## CreateLocationNfs

Creates a transfer *location* for a Network File System (NFS) file server. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync accesses NFS file servers.

## **Request Syntax**

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **MountOptions**

Specifies the options that DataSync can use to mount your NFS file server.

Type: NfsMountOptions object

Required: No

## **OnPremConfig**

Specifies the Amazon Resource Name (ARN) of the DataSync agent that can connect to your NFS file server.

You can specify more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

Type: OnPremConfig object

Required: Yes

## ServerHostname

Specifies the DNS name or IP version 4 address of the NFS file server that your DataSync agent connects to.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-:]\*[A-Za-z0-9])\$

Required: Yes

## **Subdirectory**

Specifies the export path in your NFS file server that you want DataSync to mount.

This path (or a subdirectory of the path) is where DataSync transfers data to or from. For information on configuring an export for DataSync, see Accessing NFS file servers.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9]-+\./\(\)p{Zs}]+$ \$

Required: Yes

## Tags

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

Type: Array of <u>TagListEntry</u> objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

## **Response Syntax**

```
{
    "LocationArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### LocationArn

The ARN of the transfer location that you created for your NFS file server.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]^{-0-9}+:[0-9]^{12}: location/loc-[0-9a-z]^{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **Examples**

## **Example**

The following example creates a DataSync transfer location for an NFS file server.

## **Sample Request**

```
{
  "MountOptions": {
     "Version": : "NFS4_0"
     },
  "OnPremConfig": {
    "AgentArn": [ "arn:aws:datasync:us-east-2:111222333444:agent/
agent-0b0addbeef44b3nfs" ]
          },
           "ServerHostname": "MyServer@amazon.com",
           "Subdirectory": "/MyFolder",
           "Tags": [
              {
                "Key": "Name",
                "Value": "FileSystem-1"
           ]
}
```

## **Example**

The response returns the ARN of the NFS location.

## **Sample Response**

```
{
   "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-07db7abfc326c50aa"
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## CreateLocationObjectStorage

Creates a transfer *location* for an object storage system. AWS DataSync can use this location as a source or destination for transferring data. You can make transfers with or without a <u>DataSync</u> agent.

Before you begin, make sure that you understand the <u>prerequisites</u> for DataSync to work with object storage systems.

## **Request Syntax**

```
{
   "AccessKey": "string",
   "AgentArns": [ "string" ],
   "BucketName": "string",
   "CmkSecretConfig": {
      "KmsKeyArn": "string",
      "SecretArn": "string"
   },
   "CustomSecretConfig": {
      "SecretAccessRoleArn": "string",
      "SecretArn": "string"
   },
   "SecretKey": "string",
   "ServerCertificate": blob,
   "ServerHostname": "string",
   "ServerPort": number,
   "ServerProtocol": "string",
   "Subdirectory": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
   ]
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### AccessKey

Specifies the access key (for example, a user name) if credentials are required to authenticate with the object storage server.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 200.

Pattern: ^.\*\$

Required: No

## **AgentArns**

(Optional) Specifies the Amazon Resource Names (ARNs) of the DataSync agents that can connect with your object storage system. If you are setting up an agentless cross-cloud transfer, you do not need to specify a value for this parameter.



#### Note

Make sure you configure this parameter correctly when you first create your storage location. You cannot add or remove agents from a storage location after you create it.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z -0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}\$

Required: No

## **BucketName**

Specifies the name of the object storage bucket involved in the transfer.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern:  $^{a-zA-Z0-9} -++..())$ 

Required: Yes

## CmkSecretConfig

Specifies configuration information for a DataSync-managed secret, which includes the SecretKey that DataSync uses to access a specific object storage location, with a customermanaged AWS KMS key.

When you include this paramater as part of a CreateLocationObjectStorage request, you provide only the KMS key ARN. DataSync uses this KMS key together with the value you specify for the SecretKey parameter to create a DataSync-managed secret to store the location access credentials.

Make sure the DataSync has permission to access the KMS key that you specify.



#### Note

You can use either CmkSecretConfig (with SecretKey) or CustomSecretConfig (without SecretKey) to provide credentials for a CreateLocationObjectStorage request. Do not provide both parameters for the same request.

Type: CmkSecretConfig object

Required: No

## CustomSecretConfig

Specifies configuration information for a customer-managed Secrets Manager secret where the secret key for a specific object storage location is stored in plain text. This configuration includes the secret ARN, and the ARN for an IAM role that provides access to the secret.



#### Note

You can use either CmkSecretConfig (with SecretKey) or CustomSecretConfig (without SecretKey) to provide credentials for a CreateLocationObjectStorage request. Do not provide both parameters for the same request.

Type: CustomSecretConfig object

Required: No

#### SecretKey

Specifies the secret key (for example, a password) if credentials are required to authenticate with the object storage server.



#### Note

If you provide a secret using SecretKey, but do not provide secret configuration details using CmkSecretConfig or CustomSecretConfig, then DataSync stores the token using your AWS account's Secrets Manager secret.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 200.

Pattern: ^.\*\$

Required: No

#### ServerCertificate

Specifies a certificate chain for DataSync to authenticate with your object storage system if the system uses a private or self-signed certificate authority (CA). You must specify a single .pem file with a full certificate chain (for example, file:///home/user/.ssh/ object\_storage\_certificates.pem).

The certificate chain might include:

- The object storage system's certificate
- All intermediate certificates (if there are any)
- The root certificate of the signing CA

You can concatenate your certificates into a .pem file (which can be up to 32768 bytes before base64 encoding). The following example cat command creates an object\_storage\_certificates.pem file that includes three certificates:

cat object\_server\_certificate.pem intermediate\_certificate.pem ca\_root\_certificate.pem > object\_storage\_certificates.pem

To use this parameter, configure ServerProtocol to HTTPS.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 32768.

Required: No

#### ServerHostname

Specifies the domain name or IP version 4 (IPv4) address of the object storage server that your DataSync agent connects to.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-:]\*[A-Za-z0-9])\$

Required: Yes

## ServerPort

Specifies the port that your object storage server accepts inbound network traffic on (for example, port 443).

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 65536.

Required: No

## ServerProtocol

Specifies the protocol that your object storage server uses to communicate. If not specified, the default value is HTTPS.

Type: String

Valid Values: HTTPS | HTTP

Required: No

## **Subdirectory**

Specifies the object prefix for your object storage server. If this is a source location, DataSync only copies objects with this prefix. If this is a destination location, DataSync writes all objects with this prefix.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-}+\./\(\)p{Zs}]*$$ 

Required: No

## **Tags**

Specifies the key-value pair that represents a tag that you want to add to the resource. Tags can help you manage, filter, and search for your resources. We recommend creating a name tag for your location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

## **Response Syntax**

```
{
    "LocationArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## LocationArn

Specifies the ARN of the object storage system location that you create.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z  $\\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$$ 

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## CreateLocationS3

Creates a transfer *location* for an Amazon S3 bucket. AWS DataSync can use this location as a source or destination for transferring data.

## ▲ Important

Before you begin, make sure that you read the following topics:

- Storage class considerations with Amazon S3 locations
- Evaluating S3 request costs when using DataSync

For more information, see Configuring transfers with Amazon S3.

## **Request Syntax**

```
{
    "AgentArns": [ "string" ],
    "S3BucketArn": "string",
    "S3Config": {
        "BucketAccessRoleArn": "string"
    },
    "S3StorageClass": "string",
    "Subdirectory": "string",
    "Tags": [
        {
            "Key": "string",
            "value": "string"
        }
    ]
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **AgentArns**

(Amazon S3 on Outposts only) Specifies the Amazon Resource Name (ARN) of the DataSync agent on your Outpost.

For more information, see Deploy your DataSync agent on AWS Outposts.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

Required: No

#### S3BucketArn

Specifies the ARN of the S3 bucket that you want to use as a location. (When creating your DataSync task later, you specify whether this location is a transfer source or destination.)

If your S3 bucket is located on an AWS Outposts resource, you must specify an Amazon S3 access point. For more information, see <u>Managing data access with Amazon S3 access points</u> in the *Amazon S3 User Guide*.

Type: String

Length Constraints: Maximum length of 268.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3: [a-z\-0-9]*: [0-9]{12}:accesspoint[/:][a-zA-Z0-9\-.]{1,63}$|^arn: <math>(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3-outposts: [a-z\-0-9]+:[0-9]{12}:outpost[/:][a-zA-Z0-9\-]{1,63}[/:]accesspoint[/:][a-zA-Z0-9\-]{1,63}$|^arn: <math>(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3:::[a-zA-Z0-9\cdot\-_]{1,255}$
```

Required: Yes

## S3Config

Specifies the Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that DataSync uses to access your S3 bucket.

For more information, see Providing DataSync access to S3 buckets.

Type: S3Config object

Required: Yes

## S3StorageClass

Specifies the storage class that you want your objects to use when Amazon S3 is a transfer destination.

For buckets in AWS Regions, the storage class defaults to STANDARD. For buckets on AWS Outposts, the storage class defaults to OUTPOSTS.

For more information, see Storage class considerations with Amazon S3 transfers.

Type: String

Valid Values: STANDARD | STANDARD\_IA | ONEZONE\_IA | INTELLIGENT\_TIERING | GLACIER | DEEP\_ARCHIVE | OUTPOSTS | GLACIER\_INSTANT\_RETRIEVAL

Required: No

## **Subdirectory**

Specifies a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

## Note

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February
- photos/../2006/March

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9]-+./(())p{Zs}]*$$ 

Required: No

#### **Tags**

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your transfer location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

## **Response Syntax**

```
{
    "LocationArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## LocationArn

The ARN of the S3 location that you created.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **Examples**

#### Step 1. Allow to assume the IAM role required to write to the bucket

The following example shows the simplest policy that grants the required permissions for AWS DataSync to access a destination Amazon S3 bucket, followed by an IAM role to which the create-location-s3-iam-role policy has been attached.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
            "Effect": "Allow",
            "Principal": {
                 "Service": "datasync.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

```
"Role": {
    "Path": "/",
    "RoleName": "amzn-s3-demo-bucket-access-role",
    "RoleId": "role-id",
    "Arn": "arn:aws:iam::account-id:role/amzn-s3-demo-bucket-access-role",
    "CreateDate": "2018-07-27T02:49:23.117Z",
    "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
```

## Step 2. Allow the created IAM role to write to the bucket

Attach a policy that has sufficient permissions to access the bucket to the role. An example of such policy is the AWSDataSyncFullAccess managed policy.

For more information, see AWSDataSyncFullAccess in the IAM console.

You don't need to create this policy. It's managed by AWS, so all that you need to do is specify its ARN in the attach-role-policy command.

```
IAM_POLICY_ARN='arn:aws:iam::aws:policy/AWSDataSyncFullAccess'
```

#### Step 3. Create an endpoint for an Amazon S3 bucket

The following example creates an endpoint for an Amazon S3 bucket.

When the S3 endpoint is created, a response similar to the second example following returns the Amazon Resource Name (ARN) for the new Amazon S3 location.

## **Sample Request**

```
{
    "S3BucketArn": "arn:aws:s3:::amzn-s3-demo-bucket",
    "S3Config": {
        "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/amzn-s3-demo-bucket-access-role",
      },
      "S3StorageClass": "STANDARD",
      "Subdirectory": "/MyFolder",
      "Tags": [
```

```
{
    "Key": "Name",
    "Value": "s3Bucket-1"
    }
]
```

## Sample Response

```
{
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-07db7abfc326c50s3"
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## CreateLocationSmb

Creates a transfer *location* for a Server Message Block (SMB) file server. AWS DataSync can use this location as a source or destination for transferring data.

Before you begin, make sure that you understand how DataSync accesses SMB file servers. For more information, see Providing DataSync access to SMB file servers.

## **Request Syntax**

```
{
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "DnsIpAddresses": [ "string" ],
   "Domain": "string",
   "KerberosKeytab": blob,
   "KerberosKrb5Conf": blob,
   "KerberosPrincipal": "string",
   "MountOptions": {
      "Version": "string"
   },
   "Password": "string",
   "ServerHostname": "string",
   "Subdirectory": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "User": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see <u>Common Parameters</u>.

The request accepts the following data in JSON format.

## **AgentArns**

Specifies the DataSync agent (or agents) that can connect to your SMB file server. You specify an agent by using its Amazon Resource Name (ARN).

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9] agent/agent-[0-9a-z] \{17\}
```

Required: Yes

## **AuthenticationType**

Specifies the authentication protocol that DataSync uses to connect to your SMB file server. DataSync supports NTLM (default) and KERBEROS authentication.

For more information, see Providing DataSync access to SMB file servers.

Type: String

Valid Values: NTLM | KERBEROS

Required: No

## **DnsIpAddresses**

Specifies the IPv4 addresses for the DNS servers that your SMB file server belongs to. This parameter applies only if AuthenticationType is set to KERBEROS.

If you have multiple domains in your environment, configuring this parameter makes sure that DataSync connects to the right SMB file server.

Type: Array of strings

Array Members: Maximum number of 2 items.

Length Constraints: Minimum length of 7. Maximum length of 39.

```
Pattern: \A((25[0-5]|2[0-4]\d|[0-1]?\d?\d)(\.(25[0-5]|2[0-4]\d|[0-1]?\d?\d)){3}|([0-9a-fA-F]{1,4}:){7,7}[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,7}:|([0-9a-fA-F]{1,4}:){1,6}:[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,5}(:[0-9a-fA-F]{1,4}){1,2}|([0-9a-fA-F]{1,4}:){1,4}(:[0-9a-fA-F]{1,4}){1,3}|([0-9a-fA-F]{1,4}:){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA-F]{1,4})|([0-9a-fA
```

$$F]{1,4}:){1,2}(:[0-9a-fA-F]{1,4}){1,5}|[0-9a-fA-F]{1,4}:((:[0-9a-fA-F]{1,4}){1,6}))$$

Required: No

#### **Domain**

Specifies the Windows domain name that your SMB file server belongs to. This parameter applies only if AuthenticationType is set to NTLM.

If you have multiple domains in your environment, configuring this parameter makes sure that DataSync connects to the right file server.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}$ \$

Required: No

## KerberosKeytab

Specifies your Kerberos key table (keytab) file, which includes mappings between your Kerberos principal and encryption keys.

To avoid task execution errors, make sure that the Kerberos principal that you use to create the keytab file matches exactly what you specify for KerberosPrincipal.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 65536.

Required: No

## KerberosKrb5Conf

Specifies a Kerberos configuration file (krb5.conf) that defines your Kerberos realm configuration.

The file must be base64 encoded. If you're using the AWS CLI, the encoding is done for you.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 131072.

Required: No

## **KerberosPrincipal**

Specifies a Kerberos prinicpal, which is an identity in your Kerberos realm that has permission to access the files, folders, and file metadata in your SMB file server.

A Kerberos principal might look like HOST/kerberosuser@MYDOMAIN.ORG.

Principal names are case sensitive. Your DataSync task execution will fail if the principal that you specify for this parameter doesn't exactly match the principal that you use to create the keytab file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^.+\$

Required: No

## **MountOptions**

Specifies the version of the SMB protocol that DataSync uses to access your SMB file server.

Type: SmbMountOptions object

Required: No

#### **Password**

Specifies the password of the user who can mount your SMB file server and has permission to access the files and folders involved in your transfer. This parameter applies only if AuthenticationType is set to NTLM.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^. {0, 104}\$

Required: No

#### ServerHostname

Specifies the domain name or IP address of the SMB file server that your DataSync agent connects to.

Remember the following when configuring this parameter:

- You can't specify an IP version 6 (IPv6) address.
- If you're using Kerberos authentication, you must specify a domain name.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-:]\*[A-Za-z0-9])\$

Required: Yes

## **Subdirectory**

Specifies the name of the share exported by your SMB file server where DataSync will read or write data. You can include a subdirectory in the share path (for example, /path/to/subdirectory). Make sure that other SMB clients in your network can also mount this path.

To copy all data in the subdirectory, DataSync must be able to mount the SMB share and access all of its data. For more information, see Providing DataSync access to SMB file servers.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())$p{Zs}]+$$ 

Required: Yes

## <u>Tags</u>

Specifies labels that help you categorize, filter, and search for your AWS resources. We recommend creating at least a name tag for your location.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

#### User

Specifies the user that can mount and access the files, folders, and file metadata in your SMB file server. This parameter applies only if AuthenticationType is set to NTLM.

For information about choosing a user with the right level of access for your transfer, see Providing DataSync access to SMB file servers.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

Required: No

## **Response Syntax**

```
{
    "LocationArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## **LocationArn**

The ARN of the SMB location that you created.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9]{12}: location/loc-[0-9a-z]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **Examples**

#### Sample Request

The following example creates a location for an SMB file server.

```
{
   "AgentArns":[
      "arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44b3nfs",
      "arn:aws:datasync:us-east-2:111222333444:agent/agent-2345noo35nnee1123ovo3"
   ],
   "Domain": "AMAZON",
   "MountOptions":{
      "Version": "SMB3"
   },
   "Password": "string",
   "ServerHostname": "MyServer.amazon.com",
   "Subdirectory": "share",
   "Tags":[
      {
         "Key": "department",
         "Value": "finance"
      }
   ],
   "User": "user-1"
}
```

## Sample Response

A response returns the location ARN of your SMB file server.

```
{
    "LocationArn": "arn:aws:datasync:us-east-1:111222333444:location/
loc-0f01451b140b2af49"
}
```

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## CreateTask

Configures a *task*, which defines where and how AWS DataSync transfers your data.

A task includes a source location, destination location, and transfer options (such as bandwidth limits, scheduling, and more).

### 

If you're planning to transfer data to or from an Amazon S3 location, review how DataSync can affect your S3 request charges and the DataSync pricing page before you begin.

## **Request Syntax**

```
{
   "CloudWatchLogGroupArn": "string",
   "DestinationLocationArn": "string",
   "Excludes": [
      {
         "FilterType": "string",
         "Value": "string"
      }
   ],
   "Includes": [
         "FilterType": "string",
         "Value": "string"
   ],
   "ManifestConfig": {
      "Action": "string",
      "Format": "string",
      "Source": {
         "S3": {
            "BucketAccessRoleArn": "string",
            "ManifestObjectPath": "string",
            "ManifestObjectVersionId": "string",
            "S3BucketArn": "string"
         }
      }
   },
```

```
"Name": "string",
"Options": {
   "Atime": "string",
   "BytesPerSecond": number,
   "Gid": "string",
   "LogLevel": "string",
   "Mtime": "string",
   "ObjectTags": "string",
   "OverwriteMode": "string",
   "PosixPermissions": "string",
   "PreserveDeletedFiles": "string",
   "PreserveDevices": "string",
   "SecurityDescriptorCopyFlags": "string",
   "TaskQueueing": "string",
   "TransferMode": "string",
   "Uid": "string",
   "VerifyMode": "string"
},
"Schedule": {
   "ScheduleExpression": "string",
   "Status": "string"
},
"SourceLocationArn": "string",
"Tags": [
   {
      "Key": "string",
      "Value": "string"
],
"TaskMode": "string",
"TaskReportConfig": {
   "Destination": {
      "S3": {
         "BucketAccessRoleArn": "string",
         "S3BucketArn": "string",
         "Subdirectory": "string"
      }
   },
   "ObjectVersionIds": "string",
   "OutputType": "string",
   "Overrides": {
      "Deleted": {
         "ReportLevel": "string"
      },
```

```
"Skipped": {
          "ReportLevel": "string"
          },
          "Transferred": {
                "ReportLevel": "string"
          },
          "Verified": {
                "ReportLevel": "string"
          }
     },
     "ReportLevel": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## CloudWatchLogGroupArn

Specifies the Amazon Resource Name (ARN) of an Amazon CloudWatch log group for monitoring your task.

For Enhanced mode tasks, you don't need to specify anything. DataSync automatically sends logs to a CloudWatch log group named /aws/datasync.

For more information, see Monitoring data transfers with CloudWatch Logs.

Type: String

Length Constraints: Maximum length of 562.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):logs: <math>[a-z\-0-9]+: [0-9]{12}:log-group: ([^:\*]*)(:\*)?
```

Required: No

## **DestinationLocationArn**

Specifies the ARN of your transfer's destination location.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

## **Excludes**

Specifies exclude filters that define the files, objects, and folders in your source location that you don't want DataSync to transfer. For more information and examples, see <a href="Specifying what">Specifying what</a> DataSync transfers by using filters.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

## <u>Includes</u>

Specifies include filters that define the files, objects, and folders in your source location that you want DataSync to transfer. For more information and examples, see <a href="Specifying what">Specifying what</a> DataSync transfers by using filters.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

#### ManifestConfig

Configures a manifest, which is a list of files or objects that you want DataSync to transfer. For more information and configuration examples, see <a href="Specifying what DataSync transfers by using">Specifying what DataSync transfers by using a manifest</a>.

When using this parameter, your caller identity (the role that you're using DataSync with) must have the iam: PassRole permission. The <u>AWSDataSyncFullAccess</u> policy includes this permission.

Type: ManifestConfig object

Required: No

#### Name

Specifies the name of your task.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern:  $^[a-zA-Z0-9\s+=._:@/-]+$$ 

Required: No

## **Options**

Specifies your task's settings, such as preserving file metadata, verifying data integrity, among other options.

Type: Options object

Required: No

#### **Schedule**

Specifies a schedule for when you want your task to run. For more information, see <u>Scheduling</u> your task.

Type: <u>TaskSchedule</u> object

Required: No

## **SourceLocationArn**

Specifies the ARN of your transfer's source location.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9]

( 0 3] . [ 0 3] [12] . 100 d [1011/ 100 [ 0 3 4 2] [ 17]

# Required: Yes

## **Tags**

Specifies the tags that you want to apply to your task.

Tags are key-value pairs that help you manage, filter, and search for your DataSync resources.

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

#### **TaskMode**

Specifies one of the following task modes for your data transfer:

• ENHANCED - Transfer virtually unlimited numbers of objects with higher performance than Basic mode. Enhanced mode tasks optimize the data transfer process by listing, preparing, transferring, and verifying data in parallel. Enhanced mode is currently available for transfers between Amazon S3 locations, transfers between Azure Blob and Amazon S3 without an agent, and transfers between other clouds and Amazon S3 without an agent.



#### Note

To create an Enhanced mode task, the IAM role that you use to call the CreateTask operation must have the iam:CreateServiceLinkedRole permission.

 BASIC (default) - Transfer files or objects between AWS storage and all other supported DataSync locations. Basic mode tasks are subject to quotas on the number of files, objects, and directories in a dataset. Basic mode sequentially prepares, transfers, and verifies data, making it slower than Enhanced mode for most workloads.

For more information, see Understanding task mode differences.

Type: String

Valid Values: BASIC | ENHANCED

Required: No

## **TaskReportConfig**

Specifies how you want to configure a task report, which provides detailed information about your DataSync transfer. For more information, see Monitoring your DataSync transfers with task reports.

When using this parameter, your caller identity (the role that you're using DataSync with) must have the iam: PassRole permission. The AWSDataSyncFullAccess policy includes this permission.

Type: TaskReportConfig object

Required: No

## **Response Syntax**

```
{
    "<u>TaskArn</u>": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### **TaskArn**

The Amazon Resource Name (ARN) of the task.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]*: [0-9]{12}: task/task-[0-9a-f]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **Examples**

### Sample Request for an Enhanced mode task

The following example creates a DataSync task that uses Enhanced mode.

Unlike when creating Basic mode tasks, you don't have to specify an Amazon CloudWatch log group. With Enhanced mode tasks, DataSync automatically sends task logs to a log group named / aws/datasync. If that log group doesn't exist in your AWS Region, DataSync creates the log group on your behalf when you create the task.

```
{
    "SourceLocationArn": "arn:aws:datasync:us-
east-1:111222333444:location/1111aaaa2222bbbb3",
    "DestinationLocationArn": "arn:aws:datasync:us-
east-1:111222333444:location/0000zzzz1111yyyy2",
    "Name": "My Enhanced mode task",
    "TaskMode": "ENHANCED",
    "Options": {
        "TransferMode": "CHANGED",
        "VerifyMode": "ONLY_FILES_TRANSFERRED",
        "ObjectTags": "PRESERVE",
        "LogLevel": "TRANSFER"
    }
}
```

## Sample Request for a Basic mode task

The following example creates a DataSync task that uses Basic mode.

```
{
    "SourceLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-1111aaaa2222bbbb3",
    "DestinationLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-0000zzzz1111yyyy2",
    "Name": "My Basic mode task",
    "TaskMode": "BASIC",
    "Options": {
        "Atime": "BEST_EFFORT",
        "Gid": "NONE",
        "Mtime": "PRESERVE",
        "PosixPermissions": "PRESERVE",
```

```
"PreserveDevices": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "Uid": "NONE",
        "VerifyMode": "ONLY_FILES_TRANSFERRED"
    },
    "Schedule": {
        "ScheduleExpression": "0 12 ? * SUN, WED *"
    },
    "CloudWatchLogGroupArn": "arn:aws:logs:us-east-2:111222333444:log-group:/log-group-
name:*",
    "Tags": [
        {
            "Key": "Name",
            "Value": "Migration-wave-1"
        }
    ]
}
```

### Sample Response

The following response includes the ARN of a created task.

```
{
    "TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026"
}
```

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

# **DeleteAgent**

Removes an AWS DataSync agent resource from your AWS account.

Keep in mind that this operation (which can't be undone) doesn't remove the agent's virtual machine (VM) or Amazon EC2 instance from your storage environment. For next steps, you can delete the VM or instance from your storage environment or reuse it to activate a new agent.

## **Request Syntax**

```
{
    "AgentArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **AgentArn**

The Amazon Resource Name (ARN) of the agent to delete. Use the ListAgents operation to return a list of agents for your account and AWS Region.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

Required: Yes

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

DeleteAgent 531

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteAgent 532

## **DeleteLocation**

Deletes a transfer location resource from AWS DataSync.

## **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

The Amazon Resource Name (ARN) of the location to delete.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

DeleteLocation 533

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteLocation 534

## DeleteTask

Deletes a transfer task resource from AWS DataSync.

## **Request Syntax**

```
{
    "<u>TaskArn</u>": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **TaskArn**

Specifies the Amazon Resource Name (ARN) of the task that you want to delete.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]*:[0-9]{12}: task/task-[0-9a-f]{17}$
```

Required: Yes

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

DeleteTask 535

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteTask 536

# **DescribeAgent**

Returns information about an AWS DataSync agent, such as its name, service endpoint type, and status.

## **Request Syntax**

```
{
    "AgentArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **AgentArn**

Specifies the Amazon Resource Name (ARN) of the DataSync agent that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

Required: Yes

# **Response Syntax**

```
{
   "AgentArn": "string",
   "CreationTime": number,
   "EndpointType": "string",
   "LastConnectionTime": number,
   "Name": "string",
   "Platform": {
        "Version": "string"
        "string"
```

```
},
"PrivateLinkConfig": {
    "PrivateLinkEndpoint": "string",
    "SecurityGroupArns": [ "string" ],
    "SubnetArns": [ "string" ],
    "VpcEndpointId": "string"
},
"Status": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **AgentArn**

The ARN of the agent.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

#### CreationTime

The time that the agent was activated.

Type: Timestamp

## **EndpointType**

The type of service endpoint that your agent is connected to.

Type: String

Valid Values: PUBLIC | PRIVATE\_LINK | FIPS

### LastConnectionTime

The last time that the agent was communicating with the DataSync service.

Type: Timestamp

#### Name

The name of the agent.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

### **Platform**

The platform-related details about the agent, such as the version number.

Type: Platform object

## **PrivateLinkConfig**

The network configuration that the agent uses when connecting to a VPC service endpoint.

Type: PrivateLinkConfig object

#### **Status**

The status of the agent.

• If the status is ONLINE, the agent is configured properly and ready to use.

 If the status is OFFLINE, the agent has been out of contact with DataSync for five minutes or longer. This can happen for a few reasons. For more information, see <u>What do I do if my</u> agent is offline?

Type: String

Valid Values: ONLINE | OFFLINE

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **Examples**

### **Sample Request**

The following example returns information about an agent specified in a request.

```
{
   "AgentArn": "arn:aws:datasync:us-east-2:111122223333:agent/agent-1234567890abcdef0"
}
```

### Sample Response

The following example response describes an agent that uses a public service endpoint.

```
{
    "AgentArn": "arn:aws:datasync:us-east-2:111122223333:agent/
agent-1234567890abcdef0",
    "Name": "Data center migration agent",
    "Status": "ONLINE",
    "LastConnectionTime": "2022-10-17T17:21:35.540000+00:00",
    "CreationTime": "2022-10-05T20:52:29.499000+00:00",
    "EndpointType": "PUBLIC",
    "Platform": {
        "Version": "2"
    }
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationAzureBlob**

Provides details about how an AWS DataSync transfer location for Microsoft Azure Blob Storage is configured.

## **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

Specifies the Amazon Resource Name (ARN) of your Azure Blob Storage transfer location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

# **Response Syntax**

```
"AccessTier": "string",
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "BlobType": "string",
   "CmkSecretConfig": {
        "KmsKeyArn": "string",
        "SecretArn": "string"
},
```

```
"CreationTime": number,
"CustomSecretConfig": {
    "SecretAccessRoleArn": "string",
    "SecretArn": "string"
},
"LocationArn": "string",
"LocationUri": "string",
"ManagedSecretConfig": {
    "SecretArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### AccessTier

The access tier that you want your objects or files transferred into. This only applies when using the location as a transfer destination. For more information, see Access tiers.

Type: String

Valid Values: HOT | COOL | ARCHIVE

## **AgentArns**

The ARNs of the DataSync agents that can connect with your Azure Blob Storage container.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

# **AuthenticationType**

The authentication method DataSync uses to access your Azure Blob Storage. DataSync can access blob storage using a shared access signature (SAS).

Type: String

Valid Values: SAS | NONE

## **BlobType**

The type of blob that you want your objects or files to be when transferring them into Azure Blob Storage. Currently, DataSync only supports moving data into Azure Blob Storage as block blobs. For more information on blob types, see the Azure Blob Storage documentation.

Type: String

Valid Values: BLOCK

## CmkSecretConfig

Describes configuration information for a DataSync-managed secret, such as an authentication token that DataSync uses to access a specific storage location, with a customer-managed AWS KMS key.

Type: CmkSecretConfig object

### CreationTime

The time that your Azure Blob Storage transfer location was created.

Type: Timestamp

## CustomSecretConfig

Describes configuration information for a customer-managed secret, such as an authentication token that DataSync uses to access a specific storage location, with a customer-managed AWS KMS key.

Type: CustomSecretConfig object

## LocationArn

The ARN of your Azure Blob Storage transfer location.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]^{-0-9}+:[0-9]^{12}: location/loc-[0-9a-z]^{17}$ \$

#### LocationUri

The URL of the Azure Blob Storage container involved in your transfer.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$$ 

## ManagedSecretConfig

Describes configuration information for a DataSync-managed secret, such as an authentication token that DataSync uses to access a specific storage location. DataSync uses the default AWS-managed KMS key to encrypt this secret in AWS Secrets Manager.

Type: ManagedSecretConfig object

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationEfs**

Provides details about how an AWS DataSync transfer location for an Amazon EFS file system is configured.

## **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

The Amazon Resource Name (ARN) of the Amazon EFS file system location that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

# **Response Syntax**

```
"AccessPointArn": "string",
"CreationTime": number,
"Ec2Config": {
    "SecurityGroupArns": [ "string" ],
    "SubnetArn": "string"
},
"FileSystemAccessRoleArn": "string",
```

```
"InTransitEncryption": "string",

"LocationArn": "string",

"LocationUri": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### **AccessPointArn**

The ARN of the access point that DataSync uses to access the Amazon EFS file system.

For more information, see Accessing restricted file systems.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): elasticfilesystem: <math>[a-z\\-0-9]+:[0-9]\{12\}:access-point/fsap-[0-9a-f]\{8,40\}$
```

### CreationTime

The time that the location was created.

Type: Timestamp

## **Ec2Config**

The subnet and security groups that AWS DataSync uses to connect to one of your Amazon EFS file system's mount targets.

Type: Ec2Config object

## <u>FileSystemAccessRoleArn</u>

The AWS Identity and Access Management (IAM) role that allows DataSync to access your Amazon EFS file system.

For more information, see Creating a DataSync IAM role for file system access.

Type: String

Length Constraints: Maximum length of 2048.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9] {12}:role/.*$
```

## **InTransitEncryption**

Indicates whether DataSync uses Transport Layer Security (TLS) encryption when transferring data to or from the Amazon EFS file system.

Type: String

Valid Values: NONE | TLS1\_2

### LocationArn

The ARN of the Amazon EFS file system location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

### LocationUri

The URL of the Amazon EFS file system location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$$ 

### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## **Examples**

### Sample Request

The following example shows how to get information about a specific Amazon EFS file system location.

```
{
   "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-12abcdef012345678"
}
```

#### Sample Response

The following example returns location details about an Amazon EFS file system.

## Sample Response: Describing a location for a restricted Amazon EFS file system

The following example returns location details about an Amazon EFS file system with restricted access, including the AccessPointArn, FileSystemAccessRoleArn, and InTransitEncryption elements.

```
{
    "CreationTime": 1653319021.353,
    "AccessPointArn": "arn:aws:elasticfilesystem:us-east-2:111222333444:access-point/
fsap-1234567890abcdef0",
    "Ec2Config": {
        "SubnetArn": "arn:aws:ec2:us-east-2:111222333444:subnet/
subnet-1234567890abcdef1",
        "SecurityGroupArns": [
            "arn:aws:ec2:us-east-2:111222333444:security-group/sg-1234567890abcdef2"
        ]
    },
    "FileSystemAccessRoleArn": "arn:aws:iam::111222333444:role/
AwsDataSyncFullAccessNew",
    "InTransitEncryption": "TLS1_2",
    "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-
abcdef01234567890",
    "LocationUri": "efs://us-east-2.fs-021345abcdef6789/",
    "Subdirectory": "/mount/path",
    "Tags": [{
        "Key": "Name",
        "Value": "ElasticFileSystem-1"
    }]
}
```

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- · AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationFsxLustre**

Provides details about how an AWS DataSync transfer location for an Amazon FSx for Lustre file system is configured.

## **Request Syntax**

```
{
    "LocationArn": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

The Amazon Resource Name (ARN) of the FSx for Lustre location to describe.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9] {12}: [0-9]-0-9] {12}: [0-9]-0-9] {12}: [0-9]-0-9] {12}: [0-9]-0-9] {12}: [0-9]-0-9] {12}: [0-9]-0-9] {13}: [0-9]-0-9] {14]: [0-9]-0-9] {15]: [0-9]-0-9] {16]: [0-9]-0-9] {17]: [0-9]-0-9] {17]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9] {18]: [0-9]-0-9
```

Required: Yes

# **Response Syntax**

```
{
   "CreationTime": number,
   "LocationArn": "string",
   "LocationUri": "string",
   "SecurityGroupArns": [ "string" ]
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

DescribeLocationFsxLustre 553

The following data is returned in JSON format by the service.

### CreationTime

The time that the FSx for Lustre location was created.

Type: Timestamp

### LocationArn

The Amazon Resource Name (ARN) of the FSx for Lustre location that was described.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z ^0-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

### LocationUri

The URI of the FSx for Lustre location that was described.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ 

# **SecurityGroupArns**

The Amazon Resource Names (ARNs) of the security groups that are configured for the FSx for Lustre file system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]{12}: security-group/sg-[a-f0-9]+$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

DescribeLocationFsxLustre 554

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeLocationFsxLustre 555

# DescribeLocationFsxOntap

Provides details about how an AWS DataSync transfer location for an Amazon FSx for NetApp ONTAP file system is configured.



### Note

If your location uses SMB, the DescribeLocationFsxOntap operation doesn't actually return a Password.

# **Request Syntax**

```
{
   "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## LocationArn

Specifies the Amazon Resource Name (ARN) of the FSx for ONTAP file system location that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z
-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

Required: Yes

## **Response Syntax**

```
{
   "CreationTime": number,
```

```
"FsxFilesystemArn": "string",
   "LocationArn": "string",
   "LocationUri": "string",
   "Protocol": {
      "NFS": {
         "MountOptions": {
             "Version": "string"
      },
      "SMB": {
         "Domain": "string",
         "MountOptions": {
             "Version": "string"
         },
         "Password": "string",
         "<u>User</u>": "string"
      }
   },
   "SecurityGroupArns": [ "string" ],
   "StorageVirtualMachineArn": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### CreationTime

The time that the location was created.

Type: Timestamp

# **FsxFilesystemArn**

The ARN of the FSx for ONTAP file system.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): fsx: <math>[a-z\-0-9]*: [0-9]{12}: file-system/fs-.*$
```

#### **LocationArn**

The ARN of the FSx for ONTAP file system location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9]{12}: location/loc-[0-9a-z]{17}$
```

### LocationUri

The uniform resource identifier (URI) of the FSx for ONTAP file system location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$$ 

### **Protocol**

Specifies the data transfer protocol that AWS DataSync uses to access your Amazon FSx file system.

Type: FsxProtocol object

## **SecurityGroupArns**

The security groups that DataSync uses to access your FSx for ONTAP file system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]{12}: security-group/sg-[a-f0-9]+$
```

## **StorageVirtualMachineArn**

The ARN of the storage virtual machine (SVM) on your FSx for ONTAP file system where you're copying data to or from.

Type: String

Length Constraints: Maximum length of 162.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): fsx: <math>[a-z\-0-9]+: [0-9]{12}: storage-virtual-machine/fs-[0-9a-f]+/svm-[0-9a-f]{17,}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeLocationFsxOpenZfs

Provides details about how an AWS DataSync transfer location for an Amazon FSx for OpenZFS file system is configured.



### Note

Response elements related to SMB aren't supported with the DescribeLocationFsxOpenZfs operation.

## **Request Syntax**

```
{
   "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **LocationArn**

The Amazon Resource Name (ARN) of the FSx for OpenZFS location to describe.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z
```

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

# **Response Syntax**

DescribeLocationFsxOpenZfs 561

```
"CreationTime": number,
   "LocationArn": "string",
   "LocationUri": "string",
   "Protocol": {
      "NFS": {
         "MountOptions": {
            "Version": "string"
      },
      "SMB": {
         "Domain": "string",
         "MountOptions": {
            "Version": "string"
         },
         "Password": "string",
         "User": "string"
      }
   },
   "SecurityGroupArns": [ "string" ]
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## **CreationTime**

The time that the FSx for OpenZFS location was created.

Type: Timestamp

### LocationArn

The ARN of the FSx for OpenZFS location that was described.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

DescribeLocationFsxOpenZfs 562

#### LocationUri

The uniform resource identifier (URI) of the FSx for OpenZFS location that was described.

Example: fsxz://us-west-2.fs-1234567890abcdef02/fsx/folderA/folder

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ 

### **Protocol**

The type of protocol that AWS DataSync uses to access your file system.

Type: FsxProtocol object

### **SecurityGroupArns**

The ARNs of the security groups that are configured for the FSx for OpenZFS file system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]{12}:$  security-group/sg-[a-f0-9]+\$

#### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

## Internal Exception

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

DescribeLocationFsxOpenZfs 563

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- · AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationFsxWindows**

Provides details about how an AWS DataSync transfer location for an Amazon FSx for Windows File Server file system is configured.

### **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### LocationArn

Specifies the Amazon Resource Name (ARN) of the FSx for Windows File Server location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

Required: Yes

## **Response Syntax**

```
{
    "CreationTime": number,
    "Domain": "string",
    "LocationArn": "string",
    "LocationUri": "string",
    "SecurityGroupArns": [ "string" ],
    "User": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### CreationTime

The time that the FSx for Windows File Server location was created.

Type: Timestamp

### **Domain**

The name of the Microsoft Active Directory domain that the FSx for Windows File Server file system belongs to.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}$ \$

## LocationArn

The ARN of the FSx for Windows File Server location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync: [a-z ^0-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

### **LocationUri**

The uniform resource identifier (URI) of the FSx for Windows File Server location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ \$

### **SecurityGroupArns**

The ARNs of the Amazon EC2 security groups that provide access to your file system's preferred subnet.

For information about configuring security groups for file system access, see the <u>Amazon FSx for</u> Windows File Server User Guide.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]*: [0-9]{12}: security-group/sg-[a-f0-9]+$
```

### User

The user with the permissions to mount and access the FSx for Windows File Server file system.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

## Invalid Request Exception

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationHdfs**

Provides details about how an AWS DataSync transfer location for a Hadoop Distributed File System (HDFS) is configured.

### **Request Syntax**

```
{
    "LocationArn": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

Specifies the Amazon Resource Name (ARN) of the HDFS location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9]
```

Required: Yes

## **Response Syntax**

```
{
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "BlockSize": number,
   "CreationTime": number,
   "KerberosPrincipal": "string",
   "KmsKeyProviderUri": "string",
   "LocationArn": "string",
   "LocationUri": "string",
   "NameNodes": [
```

```
{
    "Hostname": "string",
    "Port": number
}

],

"QopConfiguration": {
    "DataTransferProtection": "string",
    "RpcProtection": "string"
},

"ReplicationFactor": number,
"SimpleUser": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **AgentArns**

The ARNs of the DataSync agents that can connect with your HDFS cluster.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync: [a-z \\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$
```

## **AuthenticationType**

The type of authentication used to determine the identity of the user.

Type: String

Valid Values: SIMPLE | KERBEROS

### **BlockSize**

The size of the data blocks to write into the HDFS cluster.

Type: Integer

Valid Range: Minimum value of 1048576. Maximum value of 1073741824.

#### CreationTime

The time that the HDFS location was created.

Type: Timestamp

### KerberosPrincipal

The Kerberos principal with access to the files and folders on the HDFS cluster. This parameter is used if the AuthenticationType is defined as KERBEROS.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^.+\$

### **KmsKeyProviderUri**

The URI of the HDFS cluster's Key Management Server (KMS).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

```
Pattern: ^kms:\/\/http[s]?@(([a-zA-Z0-9\-]*[a-zA-Z0-9])\.)*([A-Za-z0-9\-]*[a-zA-Z0-9\-]*[a-zA-Z0-9])\.)*([A-Za-z0-9\-]*[A-Za-z0-9]))*:[0-9]{1,5}\/kms$
```

#### LocationArn

The ARN of the HDFS location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

#### LocationUri

The URI of the HDFS location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$$ 

### **NameNodes**

The NameNode that manages the HDFS namespace.

Type: Array of HdfsNameNode objects

Array Members: Minimum number of 1 item.

### QopConfiguration

The Quality of Protection (QOP) configuration, which specifies the Remote Procedure Call (RPC) and data transfer protection settings configured on the HDFS cluster.

Type: QopConfiguration object

### ReplicationFactor

The number of DataNodes to replicate the data to when writing to the HDFS cluster.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 512.

## **SimpleUser**

The user name to identify the client on the host operating system. This parameter is used if the AuthenticationType is defined as SIMPLE.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[\_.A-Za-z0-9][-\_.A-Za-z0-9]\*\$

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationNfs**

Provides details about how an AWS DataSync transfer location for a Network File System (NFS) file server is configured.

### **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

Specifies the Amazon Resource Name (ARN) of the NFS location that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

## **Response Syntax**

```
{
    "CreationTime": number,
    "LocationArn": "string",
    "LocationUri": "string",
    "MountOptions": {
        "Version": "string"
},
    "OnPremConfig": {
```

```
"<u>AgentArns</u>": [ "string" ]
}
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### CreationTime

The time when the NFS location was created.

Type: Timestamp

### LocationArn

The ARN of the NFS location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

#### LocationUri

The URI of the NFS location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ 

### **MountOptions**

The mount options that DataSync uses to mount your NFS file server.

Type: NfsMountOptions object

## **OnPremConfig**

The AWS DataSync agents that can connect to your Network File System (NFS) file server.

Type: OnPremConfig object

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **Examples**

#### **Example**

The following example returns information about the NFS location specified in the sample request.

### Sample Request

```
{
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-07db7abfc326c50aa"
}
```

### Example

This example illustrates one usage of DescribeLocationNfs.

### Sample Response

```
{
   "CreationTime": 1532660733.39,
   "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-07db7abfc326c50aa",
   "LocationUri": "hostname.amazon.com",
```

```
"OnPremConfig": {
    "AgentArns": [ "arn:aws:datasync:us-east-2:111222333444:agent/
agent-0b0addbeef44b3nfs" ]
  }
}
```

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# ${\bf Describe Location Object Storage}$

Provides details about how an AWS DataSync transfer location for an object storage system is configured.

## **Request Syntax**

```
{
    "LocationArn": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### LocationArn

Specifies the Amazon Resource Name (ARN) of the object storage system location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

# **Response Syntax**

```
"AccessKey": "string",
"AgentArns": [ "string" ],
"CmkSecretConfig": {
    "KmsKeyArn": "string",
    "SecretArn": "string"
},
"CreationTime": number,
```

```
"CustomSecretConfig": {
    "SecretAccessRoleArn": "string",
    "SecretArn": "string"
},

"LocationArn": "string",

"LocationUri": "string",

"ManagedSecretConfig": {
    "SecretArn": "string"
},

"ServerCertificate": blob,

"ServerPort": number,

"ServerProtocol": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **AccessKey**

The access key (for example, a user name) required to authenticate with the object storage system.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 200.

Pattern: ^.\*\$

### **AgentArns**

The ARNs of the DataSync agents that can connect with your object storage system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}: agent/agent-[0-9a-z]{17}$
```

### CmkSecretConfig

Describes configuration information for a DataSync-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location, and a customer-managed AWS KMS key.

Type: CmkSecretConfig object

### CreationTime

The time that the location was created.

Type: Timestamp

### CustomSecretConfig

Describes configuration information for a customer-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location, and a customer-managed AWS KMS key.

Type: CustomSecretConfig object

### LocationArn

The ARN of the object storage system location.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9]

### **LocationUri**

The URI of the object storage system location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ 

### ManagedSecretConfig

Describes configuration information for a DataSync-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location. DataSync uses the default AWS-managed KMS key to encrypt this secret in AWS Secrets Manager.

Type: ManagedSecretConfig object

### ServerCertificate

The certificate chain for DataSync to authenticate with your object storage system if the system uses a private or self-signed certificate authority (CA).

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 32768.

### **ServerPort**

The port that your object storage server accepts inbound network traffic on (for example, port 443).

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 65536.

## **ServerProtocol**

The protocol that your object storage system uses to communicate.

Type: String

Valid Values: HTTPS | HTTP

### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationS3**

Provides details about how an AWS DataSync transfer location for an S3 bucket is configured.

### **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### **LocationArn**

Specifies the Amazon Resource Name (ARN) of the Amazon S3 location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

# **Response Syntax**

```
{
    "AgentArns": [ "string" ],
    "CreationTime": number,
    "LocationArn": "string",
    "LocationUri": "string",
    "S3Config": {
        "BucketAccessRoleArn": "string"
},
    "S3StorageClass": "string"
```

}

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **AgentArns**

The ARNs of the DataSync agents deployed on your Outpost when using working with Amazon S3 on Outposts.

For more information, see Deploy your DataSync agent on AWS Outposts.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9] agent/agent-[0-9a-z] \{17\}$
```

#### CreationTime

The time that the Amazon S3 location was created.

Type: Timestamp

### LocationArn

The ARN of the Amazon S3 location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9] + [0-9]
```

#### LocationUri

The URL of the Amazon S3 location that was described.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ 

### **S3Config**

Specifies the Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that DataSync uses to access your S3 bucket.

For more information, see Providing DataSync access to S3 buckets.

Type: S3Config object

### S3StorageClass

When Amazon S3 is a destination location, this is the storage class that you chose for your objects.

Some storage classes have behaviors that can affect your Amazon S3 storage costs. For more information, see Storage class considerations with Amazon S3 transfers.

Type: String

Valid Values: STANDARD | STANDARD\_IA | ONEZONE\_IA | INTELLIGENT\_TIERING | GLACIER | DEEP\_ARCHIVE | OUTPOSTS | GLACIER\_INSTANT\_RETRIEVAL

### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **Examples**

#### Example

The following example returns information about the Amazon S3 location specified in the sample request.

### **Sample Request**

```
{
   "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-07db7abfc326c50s3"
}
```

#### Example

This example illustrates one usage of DescribeLocationS3.

#### Sample Response

```
{
    "CreationTime": 1532660733.39,
    "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-07db7abfc326c50s3",
    "LocationUri": "s3://amzn-s3-demo-bucket",
    "S3Config": {
        "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/amzn-s3-demo-bucket-access-role",
     }
     "S3StorageClass": "STANDARD"
}
```

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeLocationSmb**

Provides details about how an AWS DataSync transfer location for a Server Message Block (SMB) file server is configured.

### **Request Syntax**

```
{
    "LocationArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

Specifies the Amazon Resource Name (ARN) of the SMB location that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

## **Response Syntax**

```
"AgentArns": [ "string" ],
"AuthenticationType": "string",
"CreationTime": number,
"DnsIpAddresses": [ "string" ],
"Domain": "string",
"KerberosPrincipal": "string",
"LocationArn": "string",
```

```
"LocationUri": "string",
"MountOptions": {
     "Version": "string"
},
    "User": "string"
}
```

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **AgentArns**

The ARNs of the DataSync agents that can connect with your SMB file server.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

### **AuthenticationType**

The authentication protocol that DataSync uses to connect to your SMB file server.

Type: String

Valid Values: NTLM | KERBEROS

#### CreationTime

The time that the SMB location was created.

Type: Timestamp

### **DnsIpAddresses**

The IPv4 addresses for the DNS servers that your SMB file server belongs to. This element applies only if AuthenticationType is set to KERBEROS.

Type: Array of strings

Array Members: Maximum number of 2 items.

Length Constraints: Minimum length of 7. Maximum length of 39.

```
Pattern: \A((25[0-5]|2[0-4]\d|[0-1]?\d?\d)(\.(25[0-5]|2[0-4]\d|[0-1]?\d?\d)){3}|([0-9a-fA-F]{1,4}:){7,7}[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,7}:|([0-9a-fA-F]{1,4}:){1,6}:[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,5}(:[0-9a-fA-F]{1,4}){1,2}|([0-9a-fA-F]{1,4}:){1,4}(:[0-9a-fA-F]{1,4}){1,3}|([0-9a-fA-F]{1,4}:){1,3}(:[0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}:){1,2}(:[0-9a-fA-F]{1,4}){1,5}|[0-9a-fA-F]{1,4}:){1,6})\z
```

#### **Domain**

The name of the Windows domain that the SMB file server belongs to. This element applies only if AuthenticationType is set to NTLM.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}$ \$

## **KerberosPrincipal**

The Kerberos principal that has permission to access the files, folders, and file metadata in your SMB file server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^.+\$

### LocationArn

The ARN of the SMB location.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z -0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

#### LocationUri

The URI of the SMB location.

Type: String

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$$ 

### **MountOptions**

The SMB protocol version that DataSync uses to access your SMB file server.

Type: SmbMountOptions object

### User

The user that can mount and access the files, folders, and file metadata in your SMB file server. This element applies only if AuthenticationType is set to NTLM.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **Examples**

#### **Example**

This example illustrates one usage of DescribeLocationSmb.

### Sample Request

```
{
    "arn:aws:datasync:us-east-1:111222333444:location/loc-0f01451b140b2af49"
}
```

### **Example**

This example illustrates one usage of DescribeLocationSmb.

### Sample Response

```
{
   "AgentArns":[
        "arn:aws:datasync:us-east-2:111222333444:agent/agent-0bc3b3dc9bbc15145",
        "arn:aws:datasync:us-east-2:111222333444:agent/agent-04b3fe3d261a18c8f"
],
   "CreationTime":"1532660733.39",
   "Domain":"AMAZON",
   "LocationArn":"arn:aws:datasync:us-east-1:111222333444:location/
loc-0f01451b140b2af49",
   "LocationUri":"smb://hostname.amazon.com/share",
   "MountOptions":{
        "Version":"SMB3"
        },
        "User":"user-1"
}
```

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

## **DescribeTask**

Provides information about a *task*, which defines where and how AWS DataSync transfers your data.

### **Request Syntax**

```
{
    "<u>TaskArn</u>": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### **TaskArn**

Specifies the Amazon Resource Name (ARN) of the transfer task that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z-0-9]*: [0-9]{12}: task/task-[0-9a-f]{17}$
```

Required: Yes

## **Response Syntax**

```
"CloudWatchLogGroupArn": "string",
    "CreationTime": number,
    "CurrentTaskExecutionArn": "string",
    "DestinationLocationArn": "string",
    "DestinationNetworkInterfaceArns": [ "string"],
    "ErrorCode": "string",
    "ErrorDetail": "string",
    "Excludes": [
```

```
{
      "FilterType": "string",
      "Value": "string"
   }
],
"Includes": [
   {
      "FilterType": "string",
      "Value": "string"
   }
],
"ManifestConfig": {
   "Action": "string",
   "Format": "string",
   "Source": {
      "S3": {
         "BucketAccessRoleArn": "string",
         "ManifestObjectPath": "string",
         "ManifestObjectVersionId": "string",
         "S3BucketArn": "string"
      }
   }
},
"Name": "string",
"Options": {
   "Atime": "string",
   "BytesPerSecond": number,
   "Gid": "string",
   "LogLevel": "string",
   "Mtime": "string",
   "ObjectTags": "string",
   "OverwriteMode": "string",
   "PosixPermissions": "string",
   "PreserveDeletedFiles": "string",
   "PreserveDevices": "string",
   "SecurityDescriptorCopyFlags": "string",
   "TaskQueueing": "string",
   "TransferMode": "string",
   "Uid": "string",
   "VerifyMode": "string"
},
"Schedule": {
   "ScheduleExpression": "string",
   "Status": "string"
```

```
},
   "ScheduleDetails": {
      "DisabledBy": "string",
      "DisabledReason": "string",
      "StatusUpdateTime": number
   },
   "SourceLocationArn": "string",
   "SourceNetworkInterfaceArns": [ "string" ],
   "Status": "string",
   "TaskArn": "string",
   "TaskMode": "string",
   "TaskReportConfig": {
      "Destination": {
         "S3": {
            "BucketAccessRoleArn": "string",
            "S3BucketArn": "string",
            "Subdirectory": "string"
         }
      },
      "ObjectVersionIds": "string",
      "OutputType": "string",
      "Overrides": {
         "Deleted": {
            "ReportLevel": "string"
         },
         "Skipped": {
            "ReportLevel": "string"
         },
         "Transferred": {
            "ReportLevel": "string"
         },
         "Verified": {
            "ReportLevel": "string"
         }
      "ReportLevel": "string"
   }
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### CloudWatchLogGroupArn

The Amazon Resource Name (ARN) of an Amazon CloudWatch log group for monitoring your task.

For more information, see Monitoring data transfers with CloudWatch Logs.

Type: String

Length Constraints: Maximum length of 562.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):logs: <math>[a-z\-0-9]+:[0-9]{12}:log-group:([^:\*]*)(:\*)?
```

#### **CreationTime**

The time that the task was created.

Type: Timestamp

#### CurrentTaskExecutionArn

The ARN of the most recent task execution.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]*:[0-9]{12}: task/task-[0-9a-f]{17}$
```

### **DestinationLocationArn**

The ARN of your transfer's destination location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

### **DestinationNetworkInterfaceArns**

The ARNs of the <u>network interfaces</u> that DataSync created for your destination location.

Type: Array of strings

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: aws[ $^a$ -a-z]{0,}:ec2:[a-z $^a$ -0-9]\*:[0-9]{12}:network-interface/eni-[0-9a-f]+\$

#### **ErrorCode**

If there's an issue with your task, you can use the error code to help you troubleshoot the problem. For more information, see Troubleshooting issues with DataSync transfers.

Type: String

### **ErrorDetail**

If there's an issue with your task, you can use the error details to help you troubleshoot the problem. For more information, see Troubleshooting issues with DataSync transfers.

Type: String

### **Excludes**

The exclude filters that define the files, objects, and folders in your source location that you don't want DataSync to transfer. For more information and examples, see <a href="Specifying what">Specifying what</a> DataSync transfers by using filters.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

## **Includes**

The include filters that define the files, objects, and folders in your source location that you want DataSync to transfer. For more information and examples, see <a href="Specifying what DataSync transfers">Specifying what DataSync transfers</a> by using filters.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

## ManifestConfig

The configuration of the manifest that lists the files or objects that you want DataSync to transfer. For more information, see Specifying what DataSync transfers by using a manifest.

Type: ManifestConfig object

#### Name

The name of your task.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

### **Options**

The task's settings. For example, what file metadata gets preserved, how data integrity gets verified at the end of your transfer, bandwidth limits, among other options.

Type: Options object

### **Schedule**

The schedule for when you want your task to run. For more information, see <u>Scheduling your</u> task.

Type: TaskSchedule object

### **ScheduleDetails**

The details about your task schedule.

Type: TaskScheduleDetails object

#### **SourceLocationArn**

The ARN of your transfer's source location.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z] - 0-9]+: [0-9] + [0-9]

### **SourceNetworkInterfaceArns**

The ARNs of the <u>network interfaces</u> that DataSync created for your source location.

Type: Array of strings

Length Constraints: Maximum length of 128.

```
Pattern: ^arn:aws[\-a-z]{0,}:ec2:[a-z\-0-9]*:[0-9]{12}:network-interface/eni-[0-9a-f]+$
```

#### **Status**

The status of your task. For information about what each status means, see Task statuses.

Type: String

Valid Values: AVAILABLE | CREATING | QUEUED | RUNNING | UNAVAILABLE

#### **TaskArn**

The ARN of your task.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z \ -0-9]*:[0-9]\{12\}: task/task-[0-9a-f]\{17\}$
```

#### **TaskMode**

The task mode that you're using. For more information, see <u>Choosing a task mode for your data</u> transfer.

Type: String

Valid Values: BASIC | ENHANCED

### **TaskReportConfig**

The configuration of your task report, which provides detailed information about your DataSync transfer. For more information, see <u>Monitoring your DataSync transfers with task reports</u>.

Type: TaskReportConfig object

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **Examples**

#### Sample Request

The following example specifies the ARN of a DataSync task to get information about.

```
{
    "TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026"
}
```

#### Sample Response

The following example shows a DescribeTask response.

```
"TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026",
  "Name": "MyTask",
  "TaskMode": "BASIC",
  "Status": "RUNNING",
  "SourceLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-1111aaaa2222bbbb3",
  "DestinationLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/
loc-0000zzzz1111yyyy2",
  "CurrentTaskExecutionArn": "arn:aws:datasync:us-east-2:111222333444:task/
task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f",
  "CreationTime": 1532660733.39,
  "Options": {
    "Atime": "BEST_EFFORT",
    "BytesPerSecond": 1000,
    "Gid": "NONE",
    "Mtime": "PRESERVE",
```

```
"PosixPermissions": "PRESERVE",
    "PreserveDevices": "NONE",
    "PreserveDeletedFiles": "PRESERVE",
    "Uid": "NONE",
    "VerifyMode": "POINT_IN_TIME_CONSISTENT"
    },
    "CloudWatchLogGroupArn": "arn:aws:logs:us-east-2:111222333444:log-group:/log-group-name:*"
}
```

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

### DescribeTaskExecution

Provides information about an execution of your AWS DataSync task. You can use this operation to help monitor the progress of an ongoing data transfer or check the results of the transfer.



#### Note

Some DescribeTaskExecution response elements are only relevant to a specific task mode. For information, see Understanding task mode differences and Understanding data transfer performance counters.

### **Request Syntax**

```
{
   "TaskExecutionArn": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### **TaskExecutionArn**

Specifies the Amazon Resource Name (ARN) of the task execution that you want information about.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z
-0-9: [0-9]12}: task/task-[0-9a-f]17}/execution/exec-[0-9a-f]17}$
```

Required: Yes

### **Response Syntax**

```
"BytesCompressed": number,
"BytesTransferred": number,
"BytesWritten": number,
"EndTime": number,
"EstimatedBytesToTransfer": number,
"EstimatedFilesToDelete": number,
"EstimatedFilesToTransfer": number,
"Excludes": [
   {
      "FilterType": "string",
      "Value": "string"
   }
],
"FilesDeleted": number,
"FilesFailed": {
   "Delete": number,
   "Prepare": number,
   "Transfer": number,
   "Verify": number
},
"FilesListed": {
   "AtDestinationForDelete": number,
   "AtSource": number
},
"FilesPrepared": number,
"FilesSkipped": number,
"FilesTransferred": number,
"FilesVerified": number,
"Includes": [
   {
      "FilterType": "string",
      "Value": "string"
   }
],
"LaunchTime": number,
"ManifestConfig": {
   "Action": "string",
   "Format": "string",
   "Source": {
      "S3": {
         "BucketAccessRoleArn": "string",
         "ManifestObjectPath": "string",
         "ManifestObjectVersionId": "string",
         "S3BucketArn": "string"
```

```
}
   }
},
"Options": {
   "Atime": "string",
   "BytesPerSecond": number,
   "Gid": "string",
   "LogLevel": "string",
   "Mtime": "string",
   "ObjectTags": "string",
   "OverwriteMode": "string",
   "PosixPermissions": "string",
   "PreserveDeletedFiles": "string",
   "PreserveDevices": "string",
   "SecurityDescriptorCopyFlags": "string",
   "TaskQueueing": "string",
   "TransferMode": "string",
   "Uid": "string",
   "VerifyMode": "string"
},
"ReportRes<u>ult</u>": {
   "ErrorCode": "string",
   "ErrorDetail": "string",
   "Status": "string"
},
"Result": {
   "ErrorCode": "string",
   "ErrorDetail": "string",
   "PrepareDuration": number,
   "PrepareStatus": "string",
   "TotalDuration": number,
   "TransferDuration": number,
   "TransferStatus": "string",
   "VerifyDuration": number,
   "VerifyStatus": "string"
},
"StartTime": number,
"Status": "string",
"TaskExecutionArn": "string",
"TaskMode": "string",
"TaskReportConfig": {
   "Destination": {
      "S3": {
         "BucketAccessRoleArn": "string",
```

```
"S3BucketArn": "string",
             "Subdirectory": "string"
         }
      },
      "ObjectVersionIds": "string",
      "OutputType": "string",
      "Overrides": {
         "Deleted": {
            "ReportLevel": "string"
         },
         "Skipped": {
            "ReportLevel": "string"
         },
         "Transferred": {
             "ReportLevel": "string"
         },
         "Verified": {
            "ReportLevel": "string"
         }
      },
      "ReportLevel": "string"
   }
}
```

### **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **BytesCompressed**

The number of physical bytes that DataSync transfers over the network after compression (if compression is possible). This number is typically less than <a href="BytesTransferred">BytesTransferred</a> unless the data isn't compressible.

Type: Long

### **BytesTransferred**

The number of bytes that DataSync sends to the network before compression (if compression is possible). For the number of bytes transferred over the network, see BytesCompressed.

Type: Long

#### **BytesWritten**

The number of logical bytes that DataSync actually writes to the destination location.

Type: Long

### **EndTime**

The time that the transfer task ends.

Type: Timestamp

#### EstimatedBytesToTransfer

The number of logical bytes that DataSync expects to write to the destination location.

Type: Long

#### **EstimatedFilesToDelete**

The number of files, objects, and directories that DataSync expects to delete in your destination location. If you don't configure your task to <u>delete data in the destination that isn't in the source</u>, the value is always 0.

Type: Long

### **EstimatedFilesToTransfer**

The number of files, objects, and directories that DataSync expects to transfer over the network. This value is calculated while DataSync prepares the transfer.

How this gets calculated depends primarily on your task's <u>transfer mode</u> configuration:

- If TranserMode is set to CHANGED The calculation is based on comparing the content of the source and destination locations and determining the difference that needs to be transferred. The difference can include:
  - · Anything that's added or modified at the source location.
  - Anything that's in both locations and modified at the destination after an initial transfer (unless OverwriteMode is set to NEVER).
  - (Basic task mode only) The number of items that DataSync expects to delete (if PreserveDeletedFiles is set to REMOVE).
- If TranserMode is set to ALL The calculation is based only on the items that DataSync finds at the source location.

Type: Long

#### **Excludes**

A list of filter rules that exclude specific data during your transfer. For more information and examples, see Filtering data transferred by DataSync.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

#### **FilesDeleted**

The number of files, objects, and directories that DataSync actually deletes in your destination location. If you don't configure your task to <u>delete data in the destination that isn't in the source</u>, the value is always 0.

Type: Long

#### **FilesFailed**

The number of objects that DataSync fails to prepare, transfer, verify, and delete during your task execution.



Applies only to Enhanced mode tasks.

Type: TaskExecutionFilesFailedDetail object

#### **FilesListed**

The number of objects that DataSync finds at your locations.



Applies only to Enhanced mode tasks.

Type: TaskExecutionFilesListedDetail object

### **FilesPrepared**

The number of objects that DataSync will attempt to transfer after comparing your source and destination locations.



#### Note

Applies only to Enhanced mode tasks.

This counter isn't applicable if you configure your task to transfer all data. In that scenario, DataSync copies everything from the source to the destination without comparing differences between the locations.

Type: Long

### **FilesSkipped**

The number of files, objects, and directories that DataSync skips during your transfer.

Type: Long

#### **FilesTransferred**

The number of files, objects, and directories that DataSync actually transfers over the network. This value is updated periodically during your task execution when something is read from the source and sent over the network.

If DataSync fails to transfer something, this value can be less than EstimatedFilesToTransfer. In some cases, this value can also be greater than EstimatedFilesToTransfer. This element is implementation-specific for some location types, so don't use it as an exact indication of what's transferring or to monitor your task execution.

Type: Long

### **FilesVerified**

The number of files, objects, and directories that DataSync verifies during your transfer.



#### Note

When you configure your task to verify only the data that's transferred, DataSync doesn't verify directories in some situations or files that fail to transfer.

Type: Long

#### **Includes**

A list of filter rules that include specific data during your transfer. For more information and examples, see Filtering data transferred by DataSync.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

#### LaunchTime

The time that the task execution actually begins. For non-queued tasks, LaunchTime and StartTime are typically the same. For queued tasks, LaunchTime is typically later than StartTime because previously queued tasks must finish running before newer tasks can begin.

Type: Timestamp

### ManifestConfig

The configuration of the manifest that lists the files or objects to transfer. For more information, see Specifying what DataSync transfers by using a manifest.

Type: ManifestConfig object

### **Options**

Indicates how your transfer task is configured. These options include how DataSync handles files, objects, and their associated metadata during your transfer. You also can specify how to verify data integrity, set bandwidth limits for your task, among other options.

Each option has a default value. Unless you need to, you don't have to configure any option before calling StartTaskExecution.

You also can override your task options for each task execution. For example, you might want to adjust the LogLevel for an individual execution.

Type: Options object

### ReportResult

Indicates whether DataSync generated a complete <u>task report</u> for your transfer.

Type: ReportResult object

#### Result

The result of the task execution.

Type: TaskExecutionResultDetail object

#### **StartTime**

The time that DataSync sends the request to start the task execution. For non-queued tasks, LaunchTime and StartTime are typically the same. For queued tasks, LaunchTime is typically later than StartTime because previously queued tasks must finish running before newer tasks can begin.

Type: Timestamp

#### <u>Status</u>

The status of the task execution.

For detailed information about task execution statuses, see Task execution statuses.

Type: String

Valid Values: QUEUED | CANCELLING | LAUNCHING | PREPARING | TRANSFERRING | VERIFYING | SUCCESS | ERROR

#### **TaskExecutionArn**

The ARN of the task execution that you wanted information about. TaskExecutionArn is hierarchical and includes TaskArn for the task that was executed.

```
For example, a TaskExecution value with the ARN arn:aws:datasync:us-east-1:111222333444:task/task-0208075f79cedf4a2/execution/exec-08ef1e88ec491019b executed the task with the ARN arn:aws:datasync:us-east-1:111222333444:task/task-0208075f79cedf4a2.
```

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$
```

#### **TaskMode**

The task mode that you're using. For more information, see <u>Choosing a task mode for your data</u> transfer.

Type: String

Valid Values: BASIC | ENHANCED

### **TaskReportConfig**

The configuration of your task report, which provides detailed information about for your DataSync transfer. For more information, see Creating a task report.

Type: TaskReportConfig object

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **Examples**

### **Sample Request**

This example illustrates a DescribeTaskExecution request.

```
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-
aaaabbbbbccccddddf/execution/exec-1234abcd1234abcd1"
}
```

#### Sample Response 1 for an Enhanced mode task execution

The following example describes a DataSync task execution that uses Enhanced mode. The execution is also transferring specific data by using exclude and include filters.

```
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-
aaaabbbbccccddddf/execution/exec-1234abcd1234abcd1",
    "Status": "SUCCESS",
    "Options": {
        "VerifyMode": "ONLY_FILES_TRANSFERRED",
        "OverwriteMode": "ALWAYS",
        "Atime": "BEST_EFFORT",
        "Mtime": "PRESERVE",
        "Uid": "NONE",
        "Gid": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "PreserveDevices": "NONE",
        "PosixPermissions": "NONE",
        "BytesPerSecond": -1,
        "TaskQueueing": "ENABLED",
        "LogLevel": "BASIC",
        "TransferMode": "CHANGED",
        "SecurityDescriptorCopyFlags": "NONE",
        "ObjectTags": "PRESERVE"
    },
    "Excludes": [{
        "FilterType": "SIMPLE_PATTERN",
        "Value": "/archive-files"
    }],
    "Includes": [{
        "FilterType": "SIMPLE_PATTERN",
        "Value": "/files"
    }],
    "StartTime": "2024-10-16T11:19:56.844000-04:00",
    "EstimatedFilesToTransfer": 7,
    "EstimatedBytesToTransfer": 30,
    "FilesTransferred": 7,
    "BytesWritten": 30,
    "BytesTransferred": 30,
    "BytesCompressed": 30,
    "Result": {
        "PrepareDuration": 0,
        "PrepareStatus": "SUCCESS",
```

```
"TotalDuration": 3310,
        "TransferDuration": 0,
        "TransferStatus": "SUCCESS",
        "VerifyDuration": 0,
        "VerifyStatus": "SUCCESS"
    },
    "FilesDeleted": 0,
    "FilesSkipped": 0,
    "FilesVerified": 7,
    "EstimatedFilesToDelete": 0,
    "TaskMode": "ENHANCED",
    "FilesPrepared": 7,
    "FilesListed": {
        "AtSource": 7,
        "AtDestinationForDelete": 0
    },
    "FilesFailed": {
        "Prepare": 0,
        "Transfer": 0,
        "Verify": 0,
        "Delete": 0
    }
}
```

### Sample Response 2 for an Enhanced mode task execution

The following example describes another DataSync task execution that uses Enhanced mode. In this situation, the execution is transferring specific data by using a manifest instead of filters.

```
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-
aaaabbbbccccddddf/execution/exec-1234abcd1234abcd1",
    "Status": "SUCCESS",
    "Options": {
        "VerifyMode": "ONLY_FILES_TRANSFERRED",
        "OverwriteMode": "ALWAYS",
        "Atime": "BEST_EFFORT",
        "Mtime": "PRESERVE",
        "Uid": "NONE",
        "Gid": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "PreserveDevices": "NONE",
        "PosixPermissions": "NONE",
        "PosixPermissions": "NONE",
```

```
"BytesPerSecond": -1,
        "TaskQueueing": "ENABLED",
        "LogLevel": "TRANSFER",
        "TransferMode": "CHANGED",
        "SecurityDescriptorCopyFlags": "NONE",
        "ObjectTags": "PRESERVE"
    },
    "Excludes": [],
    "Includes": [],
    "ManifestConfig": {
        "Action": "TRANSFER",
        "Format": "CSV",
        "S3AccessRoleArn": "arn:aws:iam::111222333444:role/service-role/
DataSyncS3ManifestAccess",
        "S3Bucket": "arn:aws:s3:::manifests-datasync",
        "VersionId": "Ixs7NQzE0j8BkL9r4ywX2FtDh_cPf3mG",
        "Source": {
            "S3": {
                "ManifestObjectPath": "manifest-folder/manifest-versioned-files",
                "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/my-manifest-
role/DataSyncS3ManifestAccess",
                "S3BucketArn": "arn:aws:s3:::manifests-datasync",
                "ManifestObjectVersionId": "Ixs7NQzEOj8BkL9r4ywX2FtDh_cPf3mG"
            }
        }
    },
    "StartTime": "2024-10-16T09:29:56.757000-04:00",
    "EstimatedFilesToTransfer": 1,
    "EstimatedBytesToTransfer": 6,
    "FilesTransferred": 1,
    "BytesWritten": 6,
    "BytesTransferred": 6,
    "BytesCompressed": 6,
    "Result": {
        "PrepareDuration": 0,
        "PrepareStatus": "SUCCESS",
        "TotalDuration": 3089,
        "TransferDuration": 0,
        "TransferStatus": "SUCCESS",
        "VerifyDuration": 0,
        "VerifyStatus": "SUCCESS"
    "TaskReportConfig": {
        "Destination": {
```

```
"S3": {
                "Subdirectory": "reports/",
                "S3BucketArn": "arn:aws:s3:::my-task-report",
                "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/my-task-report-
role/DataSyncTaskReportS3BucketAccess"
            }
        },
        "OutputType": "STANDARD",
        "ReportLevel": "SUCCESSES_AND_ERRORS",
        "ObjectVersionIds": "INCLUDE"
    },
    "FilesDeleted": 0,
    "FilesSkipped": 0,
    "FilesVerified": 1,
    "ReportResult": {
        "Status": "SUCCESS"
    },
    "EstimatedFilesToDelete": 0,
    "TaskMode": "ENHANCED",
    "FilesPrepared": 1,
    "FilesListed": {
        "AtSource": 1,
        "AtDestinationForDelete": 0
    },
    "FilesFailed": {
        "Prepare": 0,
        "Transfer": 0,
        "Verify": 0,
        "Delete": 0
    }
}
```

#### Sample Response for a Basic mode task execution

The following example describes a DataSync task execution that uses Basic mode.

```
{
   "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-
aaaabbbbccccddddf/execution/exec-1234abcd1234abcd1",
   "BytesCompressed": 3500,
   "BytesTransferred": 5000,
   "BytesWritten": 5000,
   "EstimatedBytesToTransfer": 5000,
   "EstimatedFilesToDelete": 10,
```

```
"EstimatedFilesToTransfer": 100,
    "FilesDeleted": 10,
    "FilesSkipped": 0,
    "FilesTransferred": 100,
    "FilesVerified": 100,
    "Result": {
        "PrepareDuration": 100,
        "PrepareStatus": "SUCCESS",
        "TransferDuration": 60,
        "TransferStatus": "SUCCESS",
        "VerifyDuration": 30,
        "VerifyStatus": "SUCCESS"
    },
    "StartTime": "2024-10-16T11:19:56.844000-04:00",
    "Status": "SUCCESS",
    "OverrideOptions": {
        "Atime": "BEST_EFFORT",
        "BytesPerSecond": "1000",
        "Gid": "NONE",
        "Mtime": "PRESERVE",
        "PosixPermissions": "PRESERVE",
        "PreserveDeletedFiles": "PRESERVE",
        "Uid": "NONE",
        "VerifyMode": "POINT_IN_TIME_CONSISTENT"
    },
    "TaskReportConfig": {
        "Destination": {
            "S3": {
                "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/my-datasync-
role",
                "S3BucketArn": "arn:aws:s3:::my-task-reports-bucket/*",
                "Subdirectory": "reports"
            }
        },
        "ObjectVersionIds": "INCLUDE",
        "OutputType": "STANDARD",
        "Overrides": {
            "Deleted": {
                "ReportLevel": "ERRORS_ONLY"
            },
            "Skipped": {
                "ReportLevel": "SUCCESSES_AND_ERRORS"
            },
            "Transferred": {
```

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

### ListAgents

Returns a list of AWS DataSync agents that belong to an AWS account in the AWS Region specified in the request.

With pagination, you can reduce the number of agents returned in a response. If you get a truncated list of agents in a response, the response contains a marker that you can specify in your next request to fetch the next page of agents.

ListAgents is eventually consistent. This means the result of running the operation might not reflect that you just created or deleted an agent. For example, if you create an agent with <a href="Milestagent"><u>CreateAgent</u></a> and then immediately run ListAgents, that agent might not show up in the list right away. In situations like this, you can always confirm whether an agent has been created (or deleted) by using DescribeAgent.

### **Request Syntax**

```
{
    "MaxResults": number,
    "NextToken": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### **MaxResults**

Specifies the maximum number of DataSync agents to list in a response. By default, a response shows a maximum of 100 agents.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

#### **NextToken**

Specifies an opaque string that indicates the position to begin the next list of results in the response.

ListAgents 619

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

Required: No

### **Response Syntax**

### **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **Agents**

A list of DataSync agents in your AWS account in the AWS Region specified in the request. The list is ordered by the agents' Amazon Resource Names (ARNs).

Type: Array of AgentListEntry objects

### **NextToken**

The opaque string that indicates the position to begin the next list of results in the response.

Type: String

ListAgents 620

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListAgents 621

### ListLocations

Returns a list of source and destination locations.

If you have more locations than are returned in a response (that is, the response returns only a truncated list of your agents), the response contains a token that you can specify in your next request to fetch the next page of locations.

### **Request Syntax**

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **Filters**

You can use API filters to narrow down the list of resources returned by ListLocations. For example, to retrieve all tasks on a specific source location, you can use ListLocations with filter name LocationType S3 and Operator Equals.

Type: Array of <u>LocationFilter</u> objects

Required: No

#### **MaxResults**

The maximum number of locations to return.

Type: Integer

ListLocations 622

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

#### NextToken

An opaque string that indicates the position at which to begin the next list of locations.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

Required: No

### **Response Syntax**

### **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

#### **Locations**

An array that contains a list of locations.

Type: Array of LocationListEntry objects

#### NextToken

An opaque string that indicates the position at which to begin returning the next list of locations.

ListLocations 623

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListLocations 624

## ListTagsForResource

Returns all the tags associated with an AWS resource.

### **Request Syntax**

```
{
    "MaxResults": number,
    "NextToken": "string",
    "ResourceArn": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **MaxResults**

Specifies how many results that you want in the response.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

#### NextToken

Specifies an opaque string that indicates the position to begin the next list of results in the response.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

Required: No

### ResourceArn

Specifies the Amazon Resource Name (ARN) of the resource that you want tag information on.

ListTagsForResource 625

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+:[0-9]{12}: (agent|task|location|system)/((agent|task|loc)-[a-f0-9]{17}|storage-system-[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{17})?
```

Required: Yes

### **Response Syntax**

### **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **NextToken**

The opaque string that indicates the position to begin the next list of results in the response.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

### Tags

An array of tags applied to the specified resource.

ListTagsForResource 626

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListTagsForResource 627

### ListTaskExecutions

Returns a list of executions for an AWS DataSync transfer task.

### **Request Syntax**

```
{
    "MaxResults": number,
    "NextToken": "string",
    "TaskArn": "string"
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see **Common Parameters**.

The request accepts the following data in JSON format.

#### **MaxResults**

Specifies how many results you want in the response.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

#### **NextToken**

Specifies an opaque string that indicates the position at which to begin the next list of results in the response.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

Required: No

#### **TaskArn**

Specifies the Amazon Resource Name (ARN) of the task that you want execution information about.

ListTaskExecutions 628

```
Type: String
```

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]*: [0-9]{12}: task/task-[0-9a-f]{17}$
```

# Response Syntax

Required: No

### **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **NextToken**

The opaque string that indicates the position to begin the next list of results in the response.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

### **TaskExecutions**

A list of the task's executions.

Type: Array of <u>TaskExecutionListEntry</u> objects

ListTaskExecutions 629

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListTaskExecutions 630

### ListTasks

Returns a list of the AWS DataSync tasks you created.

### **Request Syntax**

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **Filters**

You can use API filters to narrow down the list of resources returned by ListTasks. For example, to retrieve all tasks on a specific source location, you can use ListTasks with filter name LocationId and Operator Equals with the ARN for the location.

Type: Array of TaskFilter objects

Required: No

#### **MaxResults**

The maximum number of tasks to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

ListTasks 631

#### NextToken

An opaque string that indicates the position at which to begin the next list of tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

Required: No

### **Response Syntax**

### **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### **NextToken**

An opaque string that indicates the position at which to begin returning the next list of tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\_-]+

ListTasks 632

#### **Tasks**

A list of all the tasks that are returned.

Type: Array of TaskListEntry objects

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListTasks 633

### StartTaskExecution

Starts an AWS DataSync transfer task. For each task, you can only run one task execution at a time.

There are several steps to a task execution. For more information, see Task execution statuses.

#### Important

If you're planning to transfer data to or from an Amazon S3 location, review how DataSync can affect your S3 request charges and the DataSync pricing page before you begin.

### **Request Syntax**

```
{
   "Excludes": [
         "FilterType": "string",
         "Value": "string"
   ],
   "Includes": [
         "FilterType": "string",
         "Value": "string"
      }
   "ManifestConfig": {
      "Action": "string",
      "Format": "string",
      "Source": {
         "S3": {
            "BucketAccessRoleArn": "string",
            "ManifestObjectPath": "string",
            "ManifestObjectVersionId": "string",
            "S3BucketArn": "string"
         }
      }
   },
   "OverrideOptions": {
      "Atime": "string",
      "BytesPerSecond": number,
```

StartTaskExecution 634

```
"Gid": "string",
   "LogLevel": "string",
   "Mtime": "string",
   "ObjectTags": "string",
   "OverwriteMode": "string",
   "PosixPermissions": "string",
   "PreserveDeletedFiles": "string",
   "PreserveDevices": "string",
   "SecurityDescriptorCopyFlags": "string",
   "TaskQueueing": "string",
   "TransferMode": "string",
   "Uid": "string",
   "VerifyMode": "string"
},
"Tags": [
   {
      "Key": "string",
      "Value": "string"
   }
],
"TaskArn": "string",
"TaskReportConfig": {
   "Destination": {
      "S3": {
         "BucketAccessRoleArn": "string",
         "S3BucketArn": "string",
         "Subdirectory": "string"
   },
   "ObjectVersionIds": "string",
   "OutputType": "string",
   "Overrides": {
      "Deleted": {
         "ReportLevel": "string"
      },
      "Skipped": {
         "ReportLevel": "string"
      },
      "Transferred": {
         "ReportLevel": "string"
      },
      "Verified": {
         "ReportLevel": "string"
      }
```

```
},
   "ReportLevel": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **Excludes**

Specifies a list of filter rules that determines which files to exclude from a task. The list contains a single filter string that consists of the patterns to exclude. The patterns are delimited by "|" (that is, a pipe), for example, "/folder1|/folder2".

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

#### **Includes**

Specifies a list of filter rules that determines which files to include when running a task. The pattern should contain a single filter string that consists of the patterns to include. The patterns are delimited by "|" (that is, a pipe), for example, "/folder1|/folder2".

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

## ManifestConfig

Configures a manifest, which is a list of files or objects that you want DataSync to transfer. For more information and configuration examples, see <a href="Specifying what DataSync transfers by using">Specifying what DataSync transfers by using a manifest</a>.

When using this parameter, your caller identity (the role that you're using DataSync with) must have the iam: PassRole permission. The <u>AWSDataSyncFullAccess</u> policy includes this permission.

To remove a manifest configuration, specify this parameter with an empty value.

Type: ManifestConfig object

Required: No

## **OverrideOptions**

Indicates how your transfer task is configured. These options include how DataSync handles files, objects, and their associated metadata during your transfer. You also can specify how to verify data integrity, set bandwidth limits for your task, among other options.

Each option has a default value. Unless you need to, you don't have to configure any option before calling <a href="StartTaskExecution">StartTaskExecution</a>.

You also can override your task options for each task execution. For example, you might want to adjust the LogLevel for an individual execution.

Type: Options object

Required: No

#### Tags

Specifies the tags that you want to apply to the Amazon Resource Name (ARN) representing the task execution.

Tags are key-value pairs that help you manage, filter, and search for your DataSync resources.

Type: Array of <u>TagListEntry</u> objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

## **TaskArn**

Specifies the Amazon Resource Name (ARN) of the task that you want to start.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]\*:[0-9]{12}: task/task-[0-9a-f]{17}\$

Required: Yes

#### **TaskReportConfig**

Specifies how you want to configure a task report, which provides detailed information about your DataSync transfer. For more information, see <a href="Monitoring your DataSync transfers with task reports">Monitoring your DataSync transfers with task reports</a>.

When using this parameter, your caller identity (the role that you're using DataSync with) must have the iam: PassRole permission. The <u>AWSDataSyncFullAccess</u> policy includes this permission.

To remove a task report configuration, specify this parameter as empty.

Type: TaskReportConfig object

Required: No

# **Response Syntax**

```
{
    "TaskExecutionArn": "string"
}
```

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

# **TaskExecutionArn**

The ARN of the running task execution.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]*: [0-9]{12}: task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$
```

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

# **Examples**

### **Sample Request**

The following example starts a task execution using the default options for the specified task.

```
{
   "OverrideOptions": {
        "Atime": "BEST_EFFORT",
        "BytesPerSecond": 1000,
        "Gid": "NONE",
        "Mtime": "PRESERVE",
        "PosixPermissions": "PRESERVE",
        "PreserveDevices": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "Uid": "NONE",
        "VerifyMode": "POINT_IN_TIME_CONSISTENT"
    },
        "TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026"
}
```

#### Sample Response

This example illustrates one usage of StartTaskExecution.

```
{
```

```
"TaskExecutionArn": "arn:aws:datasync:us-east-2:111222333444:task/
task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f"
}
```

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- · AWS SDK for Python
- AWS SDK for Ruby V3

# **TagResource**

Applies a *tag* to an AWS resource. Tags are key-value pairs that can help you manage, filter, and search for your resources.

These include AWS DataSync resources, such as locations, tasks, and task executions.

### **Request Syntax**

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### ResourceArn

Specifies the Amazon Resource Name (ARN) of the resource to apply the tag to.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: (agent|task|location|system)/((agent|task|loc)-[a-f0-9]{17}|storage-system-[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{17})?
```

Required: Yes

#### **Tags**

Specifies the tags that you want to apply to the resource.

TagResource 641

Type: Array of TagListEntry objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: Yes

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

TagResource 642

- AWS SDK for Python
- AWS SDK for Ruby V3

TagResource 643

# **UntagResource**

Removes tags from an AWS resource.

## **Request Syntax**

```
{
    "Keys": [ "string" ],
    "ResourceArn": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### **Keys**

Specifies the keys in the tags that you want to remove.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:/-]+\$

Required: Yes

#### ResourceArn

Specifies the Amazon Resource Name (ARN) of the resource to remove the tags from.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z <math>\\-0-9]+: [0-9]\{12\}: (agent|task|location|system)/((agent|task|loc)-[a-f0-9]\{17\}|storage-system-[a-f0-9]\{8\}-[a-f0-9]\{4\}-[a-f0-9]\{4\}-[a-f0-9]\{4\}-[a-f0-9]\{12\})/(execution/exec-[a-f0-9]\{17\})?
```

UntagResource 644

Required: Yes

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

# InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UntagResource 645

# **UpdateAgent**

Updates the name of an AWS DataSync agent.

## **Request Syntax**

```
{
    "AgentArn": "string",
    "Name": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **AgentArn**

The Amazon Resource Name (ARN) of the agent to update.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}: agent/agent-[0-9a-z]{17}$
```

Required: Yes

#### Name

The name that you want to use to configure the agent.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

Required: No

UpdateAgent 646

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateAgent 647

# **UpdateLocationAzureBlob**

Modifies the following configurations of the Microsoft Azure Blob Storage transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with Azure Blob Storage.

# **Request Syntax**

```
{
   "AccessTier": "string",
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "BlobType": "string",
   "CmkSecretConfig": {
      "KmsKeyArn": "string",
      "SecretArn": "string"
   },
   "CustomSecretConfig": {
      "SecretAccessRoleArn": "string",
      "SecretA<u>rn</u>": "string"
   },
   "LocationArn": "string",
   "SasConfiguration": {
      "Token": "string"
   "Subdirectory": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **AccessTier**

Specifies the access tier that you want your objects or files transferred into. This only applies when using the location as a transfer destination. For more information, see <u>Access tiers</u>.

Type: String

Valid Values: HOT | COOL | ARCHIVE

Required: No

#### **AgentArns**

(Optional) Specifies the Amazon Resource Name (ARN) of the DataSync agent that can connect with your Azure Blob Storage container. If you are setting up an agentless cross-cloud transfer, you do not need to specify a value for this parameter.

You can specify more than one agent. For more information, see Using multiple agents for your transfer.



#### Note

You cannot add or remove agents from a storage location after you initially create it.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z -0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}\$

Required: No

# AuthenticationType

Specifies the authentication method DataSync uses to access your Azure Blob Storage. DataSync can access blob storage using a shared access signature (SAS).

Type: String

Valid Values: SAS | NONE

Required: No

# BlobType

Specifies the type of blob that you want your objects or files to be when transferring them into Azure Blob Storage. Currently, DataSync only supports moving data into Azure Blob Storage as block blobs. For more information on blob types, see the Azure Blob Storage documentation.

Type: String

Valid Values: BLOCK

Required: No

## CmkSecretConfig

Specifies configuration information for a DataSync-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location, and a customer-managed AWS KMS key.

Type: CmkSecretConfig object

Required: No

## <u>CustomSecretConfig</u>

Specifies configuration information for a customer-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location, and a customer-managed AWS KMS key.

Type: CustomSecretConfig object

Required: No

#### LocationArn

Specifies the ARN of the Azure Blob Storage transfer location that you're updating.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

# **SasConfiguration**

Specifies the SAS configuration that allows DataSync to access your Azure Blob Storage.

Type: <u>AzureBlobSasConfiguration</u> object

Required: No

#### **Subdirectory**

Specifies path segments if you want to limit your transfer to a virtual directory in your container (for example, /my/images).

Type: String

Length Constraints: Maximum length of 1024.

Pattern:  $^[\p{L}\p{M}\p{Z}\p{S}\p{N}\p{P}\p{C}]*$$ 

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

# InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

# InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationEfs**

Modifies the following configuration parameters of the Amazon EFS transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with Amazon EFS.

## **Request Syntax**

```
{
    "AccessPointArn": "string",
    "FileSystemAccessRoleArn": "string",
    "InTransitEncryption": "string",
    "LocationArn": "string",
    "Subdirectory": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **AccessPointArn**

Specifies the Amazon Resource Name (ARN) of the access point that DataSync uses to mount your Amazon EFS file system.

For more information, see Accessing restricted Amazon EFS file systems.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: (^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):elasticfilesystem: [a-z\-0-9]+:[0-9]{12}:access-point/fsap-[0-9a-f]{8,40}$)|(<math>^$)
```

Required: No

## **FileSystemAccessRoleArn**

Specifies an AWS Identity and Access Management (IAM) role that allows DataSync to access your Amazon EFS file system.

For information on creating this role, see <u>Creating a DataSync IAM role for Amazon EFS file</u> system access.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: (^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]

{12}:role/.\*\$)|(^\$)

Required: No

## InTransitEncryption

Specifies whether you want DataSync to use Transport Layer Security (TLS) 1.2 encryption when it transfers data to or from your Amazon EFS file system.

If you specify an access point using AccessPointArn or an IAM role using FileSystemAccessRoleArn, you must set this parameter to TLS1\_2.

Type: String

Valid Values: NONE | TLS1\_2

Required: No

#### LocationArn

Specifies the Amazon Resource Name (ARN) of the Amazon EFS transfer location that you're updating.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

# Subdirectory

Specifies a mount path for your Amazon EFS file system. This is where DataSync reads or writes data on your file system (depending on if this is a source or destination location).

By default, DataSync uses the root directory (or <u>access point</u> if you provide one by using AccessPointArn). You can also include subdirectories using forward slashes (for example, / path/to/folder).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-}+\./\(\)p{Zs}]*$$ 

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationFsxLustre**

Modifies the following configuration parameters of the Amazon FSx for Lustre transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with FSx for Lustre.

## **Request Syntax**

```
{
    "LocationArn": "string",
    "Subdirectory": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

#### LocationArn

Specifies the Amazon Resource Name (ARN) of the FSx for Lustre transfer location that you're updating.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

# **Subdirectory**

Specifies a mount path for your FSx for Lustre file system. The path can include subdirectories.

When the location is used as a source, DataSync reads data from the mount path. When the location is used as a destination, DataSync writes data to the mount path. If you don't include this parameter, DataSync uses the file system's root directory (/).

Type: String

UpdateLocationFsxLustre 657

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9]-+\./(())$ p{Zs}]+\$

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

UpdateLocationFsxLustre 658

- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateLocationFsxLustre 659

# ${\bf Update Location FsxOntap}$

Modifies the following configuration parameters of the Amazon FSx for NetApp ONTAP transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with FSx for ONTAP.

## **Request Syntax**

```
{
   "LocationArn": "string",
   "Protocol": {
      "NFS": {
         "MountOptions": {
            "Version": "string"
         }
      },
      "SMB": {
         "Domain": "string",
         "MountOptions": {
            "Version": "string"
         "Password": "string",
         "User": "string"
      }
   },
   "Subdirectory": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see **Common Parameters**.

The request accepts the following data in JSON format.

# **LocationArn**

Specifies the Amazon Resource Name (ARN) of the FSx for ONTAP transfer location that you're updating.

Type: String

UpdateLocationFsxOntap 660

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

#### **Protocol**

Specifies the data transfer protocol that DataSync uses to access your Amazon FSx file system.

Type: FsxUpdateProtocol object

Required: No

## **Subdirectory**

Specifies a path to the file share in the storage virtual machine (SVM) where you want to transfer data to or from.

You can specify a junction path (also known as a mount point), qtree path (for NFS file shares), or share name (for SMB file shares). For example, your mount path might be /vol1, /vol1/ tree1, or /share1.

#### Note

Don't specify a junction path in the SVM's root volume. For more information, see Managing FSx for ONTAP storage virtual machines in the Amazon FSx for NetApp ONTAP User Guide.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^[^\u0000\u0085\u2028\u2029\r\n]{1,255}\$

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

UpdateLocationFsxOntap 661

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

# InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateLocationFsxOntap 662

# ${\bf Update Location FsxOpen Zfs}$

Modifies the following configuration parameters of the Amazon FSx for OpenZFS transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with FSx for OpenZFS.



#### Note

Request parameters related to SMB aren't supported with the UpdateLocationFsxOpenZfs operation.

# **Request Syntax**

```
{
   "LocationArn": "string",
   "Protocol": {
      "NFS": {
         "MountOptions": {
            "Version": "string"
      },
      "SMB": {
         "Domain": "string",
         "MountOptions": {
            "Version": "string"
         },
         "Password": "string",
         "User": "string"
      }
   },
   "Subdirectory": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

UpdateLocationFsxOpenZfs 663

#### LocationArn

Specifies the Amazon Resource Name (ARN) of the FSx for OpenZFS transfer location that you're updating.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

## **Protocol**

Specifies the data transfer protocol that AWS DataSync uses to access your Amazon FSx file system.

Type: FsxProtocol object

Required: No

## **Subdirectory**

Specifies a subdirectory in the location's path that must begin with /fsx. DataSync uses this subdirectory to read or write data (depending on whether the file system is a source or destination location).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())$p{Zs}]+$$ 

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

UpdateLocationFsxOpenZfs 664

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationFsxWindows**

Modifies the following configuration parameters of the Amazon FSx for Windows File Server transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with FSx for Windows File Server.

## **Request Syntax**

```
{
    "Domain": "string",
    "LocationArn": "string",
    "Password": "string",
    "Subdirectory": "string",
    "User": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see **Common Parameters**.

The request accepts the following data in JSON format.

#### **Domain**

Specifies the name of the Windows domain that your FSx for Windows File Server file system belongs to.

If you have multiple Active Directory domains in your environment, configuring this parameter makes sure that DataSync connects to the right file system.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $([A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\})?$ \$

Required: No

#### **LocationArn**

Specifies the ARN of the FSx for Windows File Server transfer location that you're updating.

UpdateLocationFsxWindows 666

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

#### **Password**

Specifies the password of the user with the permissions to mount and access the files, folders, and file metadata in your FSx for Windows File Server file system.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^. {0,104}\$

Required: No

## **Subdirectory**

Specifies a mount path for your file system using forward slashes. DataSync uses this subdirectory to read or write data (depending on whether the file system is a source or destination location).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())$p{Zs}]+$$ 

Required: No

# <u>User</u>

Specifies the user with the permissions to mount and access the files, folders, and file metadata in your FSx for Windows File Server file system.

For information about choosing a user with the right level of access for your transfer, see required permissions for FSx for Windows File Server locations.

Type: String

UpdateLocationFsxWindows 667

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

#### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

UpdateLocationFsxWindows 668

- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationHdfs**

Modifies the following configuration parameters of the Hadoop Distributed File System (HDFS) transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with an HDFS cluster.

## **Request Syntax**

```
{
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "BlockSize": number,
   "KerberosKeytab": blob,
   "KerberosKrb5Conf": blob,
   "KerberosPrincipal": "string",
   "KmsKeyProviderUri": "string",
   "LocationArn": "string",
   "NameNodes": [
      {
         "Hostname": "string",
         "Port": number
   ],
   "QopConfiguration": {
      "DataTransferProtection": "string",
      "RpcProtection": "string"
   },
   "ReplicationFactor": number,
   "SimpleUser": "string",
   "Subdirectory": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

# **AgentArns**

The Amazon Resource Names (ARNs) of the DataSync agents that can connect to your HDFS cluster.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}\$

Required: No

### **AuthenticationType**

The type of authentication used to determine the identity of the user.

Type: String

Valid Values: SIMPLE | KERBEROS

Required: No

#### **BlockSize**

The size of the data blocks to write into the HDFS cluster.

Type: Integer

Valid Range: Minimum value of 1048576. Maximum value of 1073741824.

Required: No

## KerberosKeytab

The Kerberos key table (keytab) that contains mappings between the defined Kerberos principal and the encrypted keys. You can load the keytab from a file by providing the file's address.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 65536.

Required: No

#### KerberosKrb5Conf

The krb5.conf file that contains the Kerberos configuration information. You can load the krb5.conf file by providing the file's address. If you're using the AWS CLI, it performs the base64 encoding for you. Otherwise, provide the base64-encoded text.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 131072.

Required: No

### KerberosPrincipal

The Kerberos principal with access to the files and folders on the HDFS cluster.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^.+\$

Required: No

### **KmsKeyProviderUri**

The URI of the HDFS cluster's Key Management Server (KMS).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

```
Pattern: ^kms:\/\/http[s]?@(([a-zA-Z0-9\-]*[a-zA-Z0-9])\.)*([A-Za-z0-9\-]*[A-Za-z0-9])(;(([a-zA-Z0-9\-]*[a-zA-Z0-9])\.)*([A-Za-z0-9\-]*[A-Za-z0-9]))*:[0-9]{1,5}\/kms$
```

Required: No

## LocationArn

The Amazon Resource Name (ARN) of the source HDFS cluster location.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$
```

Required: Yes

#### **NameNodes**

The NameNode that manages the HDFS namespace. The NameNode performs operations such as opening, closing, and renaming files and directories. The NameNode contains the information to map blocks of data to the DataNodes. You can use only one NameNode.

Type: Array of HdfsNameNode objects

Array Members: Minimum number of 1 item.

Required: No

## **QopConfiguration**

The Quality of Protection (QOP) configuration specifies the Remote Procedure Call (RPC) and data transfer privacy settings configured on the Hadoop Distributed File System (HDFS) cluster.

Type: QopConfiguration object

Required: No

### ReplicationFactor

The number of DataNodes to replicate the data to when writing to the HDFS cluster.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 512.

Required: No

## <u>SimpleUser</u>

The user name used to identify the client on the host operating system.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[\_.A-Za-z0-9][-\_.A-Za-z0-9]\*\$

Required: No

## **Subdirectory**

A subdirectory in the HDFS cluster. This subdirectory is used to read data from or write data to the HDFS cluster.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_\-\+\./\(\)\$\p{Zs}]+$$ 

Required: No

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin

- AWS SDK for PHP V3
- AWS SDK for Python

• AWS SDK for Ruby V3

# **UpdateLocationNfs**

Modifies the following configuration parameters of the Network File System (NFS) transfer location that you're using with AWS DataSync.

For more information, see Configuring transfers with an NFS file server.

### **Request Syntax**

```
{
    "LocationArn": "string",
    "MountOptions": {
        "Version": "string"
},
    "OnPremConfig": {
            "AgentArns": [ "string" ]
},
    "ServerHostname": "string",
    "Subdirectory": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see **Common Parameters**.

The request accepts the following data in JSON format.

### LocationArn

Specifies the Amazon Resource Name (ARN) of the NFS transfer location that you want to update.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z \-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}$
```

Required: Yes

### **MountOptions**

Specifies how DataSync can access a location using the NFS protocol.

Type: NfsMountOptions object

Required: No

## **OnPremConfig**

The AWS DataSync agents that can connect to your Network File System (NFS) file server.

Type: OnPremConfig object

Required: No

### ServerHostname

Specifies the DNS name or IP version 4 (IPv4) address of the NFS file server that your DataSync agent connects to.

Type: String

Length Constraints: Maximum length of 255.

Pattern:  $^{(([a-zA-Z0-9])^*[a-zA-Z0-9])^*([A-Za-z0-9])^*}$ 

Required: No

## **Subdirectory**

Specifies the export path in your NFS file server that you want DataSync to mount.

This path (or a subdirectory of the path) is where DataSync transfers data to or from. For information on configuring an export for DataSync, see Accessing NFS file servers.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9]-+./(())p{Zs}]+$ \$

Required: No

# **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationObjectStorage**

Modifies the following configuration parameters of the object storage transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with an object storage system.

## **Request Syntax**

```
{
   "AccessKey": "string",
   "AgentArns": [ "string" ],
   "CmkSecretConfig": {
      "KmsKeyArn": "string",
      "SecretArn": "string"
   },
   "CustomSecretConfig": {
      "SecretAccessRoleArn": "string",
      "SecretArn": "string"
   "LocationArn": "string",
   "SecretKey": "string",
   "ServerCertificate": blob,
   "ServerHostname": "string",
   "ServerPort": number,
   "ServerProtocol": "string",
   "Subdirectory": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see <u>Common Parameters</u>.

The request accepts the following data in JSON format.

## **AccessKey**

Specifies the access key (for example, a user name) if credentials are required to authenticate with the object storage server.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 200.

Pattern: ^.\*\$

Required: No

### AgentArns

(Optional) Specifies the Amazon Resource Names (ARNs) of the DataSync agents that can connect with your object storage system. If you are setting up an agentless cross-cloud transfer, you do not need to specify a value for this parameter.



#### Note

You cannot add or remove agents from a storage location after you initially create it.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z -0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}\$

Required: No

## **CmkSecretConfig**

Specifies configuration information for a DataSync-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location, and a customer-managed AWS KMS key.

Type: CmkSecretConfig object

Required: No

## CustomSecretConfig

Specifies configuration information for a customer-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location, and a customer-managed AWS KMS key.

Type: CustomSecretConfig object

#### Required: No

#### LocationArn

Specifies the ARN of the object storage system location that you're updating.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Required: Yes

### SecretKey

Specifies the secret key (for example, a password) if credentials are required to authenticate with the object storage server.



#### Note

If you provide a secret using SecretKey, but do not provide secret configuration details using CmkSecretConfig or CustomSecretConfig, then DataSync stores the token using your AWS account's Secrets Manager secret.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 200.

Pattern: ^.\*\$

Required: No

## ServerCertificate

Specifies a certificate chain for DataSync to authenticate with your object storage system if the system uses a private or self-signed certificate authority (CA). You must specify a single .pem file with a full certificate chain (for example, file:///home/user/.ssh/ object\_storage\_certificates.pem).

The certificate chain might include:

- · The object storage system's certificate
- All intermediate certificates (if there are any)
- The root certificate of the signing CA

You can concatenate your certificates into a .pem file (which can be up to 32768 bytes before base64 encoding). The following example cat command creates an object\_storage\_certificates.pem file that includes three certificates:

cat object\_server\_certificate.pem intermediate\_certificate.pem
ca\_root\_certificate.pem > object\_storage\_certificates.pem

To use this parameter, configure ServerProtocol to HTTPS.

Updating this parameter doesn't interfere with tasks that you have in progress.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 32768.

Required: No

### ServerHostname

Specifies the domain name or IP version 4 (IPv4) address of the object storage server that your DataSync agent connects to.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-:]\*[A-Za-z0-9])\$

Required: No

### ServerPort

Specifies the port that your object storage server accepts inbound network traffic on (for example, port 443).

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 65536.

Required: No

#### ServerProtocol

Specifies the protocol that your object storage server uses to communicate.

Type: String

Valid Values: HTTPS | HTTP

Required: No

### **Subdirectory**

Specifies the object prefix for your object storage server. If this is a source location, DataSync only copies objects with this prefix. If this is a destination location, DataSync writes all objects with this prefix.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())p{Zs}]*$$ 

Required: No

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationS3**

Modifies the following configuration parameters of the Amazon S3 transfer location that you're using with AWS DataSync.

## Important

Before you begin, make sure that you read the following topics:

- Storage class considerations with Amazon S3 locations
- Evaluating S3 request costs when using DataSync

### **Request Syntax**

```
"LocationArn": "string",
"S3Config": {
    "BucketAccessRoleArn": "string"
},
"S3StorageClass": "string",
"Subdirectory": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### LocationArn

Specifies the Amazon Resource Name (ARN) of the Amazon S3 transfer location that you're updating.

Type: String

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+:[0-9]{12}: location/loc-[0-9a-z]{17}$
```

#### Required: Yes

### S3Config

Specifies the Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that DataSync uses to access your S3 bucket.

For more information, see Providing DataSync access to S3 buckets.

Type: S3Config object

Required: No

### S3StorageClass

Specifies the storage class that you want your objects to use when Amazon S3 is a transfer destination.

For buckets in AWS Regions, the storage class defaults to STANDARD. For buckets on AWS Outposts, the storage class defaults to OUTPOSTS.

For more information, see <u>Storage class considerations with Amazon S3 transfers</u>.

Type: String

Valid Values: STANDARD | STANDARD\_IA | ONEZONE\_IA | INTELLIGENT\_TIERING | GLACIER | DEEP\_ARCHIVE | OUTPOSTS | GLACIER\_INSTANT\_RETRIEVAL

Required: No

## **Subdirectory**

Specifies a prefix in the S3 bucket that DataSync reads from or writes to (depending on whether the bucket is a source or destination location).

## Note

DataSync can't transfer objects with a prefix that begins with a slash (/) or includes //, /./, or /../ patterns. For example:

- /photos
- photos//2006/January
- photos/./2006/February

### • photos/../2006/March

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9]-+\./\(\)p{Zs}]*$ \$

Required: No

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateLocationSmb**

Modifies the following configuration parameters of the Server Message Block (SMB) transfer location that you're using with AWS DataSync.

For more information, see Configuring DataSync transfers with an SMB file server.

### **Request Syntax**

```
{
   "AgentArns": [ "string" ],
   "AuthenticationType": "string",
   "DnsIpAddresses": [ "string" ],
   "Domain": "string",
   "KerberosKeytab": blob,
   "KerberosKrb5Conf": blob,
   "KerberosPrincipal": "string",
   "LocationArn": "string",
   "MountOptions": {
      "Version": "string"
   },
   "Password": "string",
   "ServerHostname": "string",
   "Subdirectory": "string",
   "User": "string"
}
```

# **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

## **AgentArns**

Specifies the DataSync agent (or agents) that can connect to your SMB file server. You specify an agent by using its Amazon Resource Name (ARN).

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+: [0-9]{12}: agent/agent-[0-9a-z]{17}$
```

Required: No

### **AuthenticationType**

Specifies the authentication protocol that DataSync uses to connect to your SMB file server. DataSync supports NTLM (default) and KERBEROS authentication.

For more information, see Providing DataSync access to SMB file servers.

Type: String

Valid Values: NTLM | KERBEROS

Required: No

### **DnsIpAddresses**

Specifies the IPv4 addresses for the DNS servers that your SMB file server belongs to. This parameter applies only if AuthenticationType is set to KERBEROS.

If you have multiple domains in your environment, configuring this parameter makes sure that DataSync connects to the right SMB file server.

Type: Array of strings

Array Members: Maximum number of 2 items.

Length Constraints: Minimum length of 7. Maximum length of 39.

```
Pattern: \A((25[0-5]|2[0-4]\d|[0-1]?\d?\d)(\.(25[0-5]|2[0-4]\d|[0-1]?\d?\d)){3}|([0-9a-fA-F]{1,4}:){7,7}[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,7}:|([0-9a-fA-F]{1,4}:){1,6}:[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,5}(:[0-9a-fA-F]{1,4}){1,2}|([0-9a-fA-F]{1,4}:){1,4}(:[0-9a-fA-F]{1,4}){1,3}|([0-9a-fA-F]{1,4}:){1,3}(:[0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}:){1,2}(:[0-9a-fA-F]{1,4}:){1,2}(:[0-9a-fA-F]{1,4}){1,5}|[0-9a-fA-F]{1,4}:((:[0-9a-fA-F]{1,4}){1,6}))\z
```

Required: No

#### **Domain**

Specifies the Windows domain name that your SMB file server belongs to. This parameter applies only if AuthenticationType is set to NTLM.

If you have multiple domains in your environment, configuring this parameter makes sure that DataSync connects to the right file server.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}$ \$

Required: No

### KerberosKeytab

Specifies your Kerberos key table (keytab) file, which includes mappings between your Kerberos principal and encryption keys.

To avoid task execution errors, make sure that the Kerberos principal that you use to create the keytab file matches exactly what you specify for KerberosPrincipal.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 65536.

Required: No

## KerberosKrb5Conf

Specifies a Kerberos configuration file (krb5.conf) that defines your Kerberos realm configuration.

The file must be base64 encoded. If you're using the AWS CLI, the encoding is done for you.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 131072.

Required: No

### KerberosPrincipal

Specifies a Kerberos prinicpal, which is an identity in your Kerberos realm that has permission to access the files, folders, and file metadata in your SMB file server.

A Kerberos principal might look like HOST/kerberosuser@MYDOMAIN.ORG.

Principal names are case sensitive. Your DataSync task execution will fail if the principal that you specify for this parameter doesn't exactly match the principal that you use to create the keytab file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^.+\$

Required: No

### **LocationArn**

Specifies the ARN of the SMB location that you want to update.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}\$

Required: Yes

## **MountOptions**

Specifies the version of the Server Message Block (SMB) protocol that AWS DataSync uses to access an SMB file server.

Type: <a href="mailto:SmbMountOptions">SmbMountOptions</a> object

Required: No

#### **Password**

Specifies the password of the user who can mount your SMB file server and has permission to access the files and folders involved in your transfer. This parameter applies only if AuthenticationType is set to NTLM.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^. {0,104}\$

Required: No

### ServerHostname

Specifies the domain name or IP address of the SMB file server that your DataSync agent connects to.

Remember the following when configuring this parameter:

- You can't specify an IP version 6 (IPv6) address.
- If you're using Kerberos authentication, you must specify a domain name.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^(([a-zA-Z0-9\-]\*[a-zA-Z0-9])\.)\*([A-Za-z0-9\-:]\*[A-Za-z0-9])\$

Required: No

## **Subdirectory**

Specifies the name of the share exported by your SMB file server where DataSync will read or write data. You can include a subdirectory in the share path (for example, /path/to/subdirectory). Make sure that other SMB clients in your network can also mount this path.

To copy all data in the specified subdirectory, DataSync must be able to mount the SMB share and access all of its data. For more information, see <a href="Providing DataSync access to SMB fileservers">Providing DataSync access to SMB fileservers</a>.

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-+}./(())$p{Zs}]+$$ 

Required: No

#### User

Specifies the user name that can mount your SMB file server and has permission to access the files and folders involved in your transfer. This parameter applies only if AuthenticationType is set to NTLM.

For information about choosing a user with the right level of access for your transfer, see Providing DataSync access to SMB file servers.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

Required: No

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateTask**

Updates the configuration of a *task*, which defines where and how AWS DataSync transfers your data.

## **Request Syntax**

```
{
   "CloudWatchLogGroupArn": "string",
   "Excludes": [
      {
         "FilterType": "string",
         "Value": "string"
   ],
   "Includes": [
      {
         "FilterType": "string",
         "Value": "string"
      }
   "ManifestConfig": {
      "Action": "string",
      "Format": "string",
      "Source": {
         "S3": {
            "BucketAccessRoleArn": "string",
            "ManifestObjectPath": "string",
            "ManifestObjectVersionId": "string",
            "S3BucketArn": "string"
         }
      }
   },
   "Name": "string",
   "Options": {
      "Atime": "string",
      "BytesPerSecond": number,
      "Gid": "string",
      "LogLevel": "string",
      "Mtime": "string",
      "ObjectTags": "string",
      "OverwriteMode": "string",
      "PosixPermissions": "string",
```

```
"PreserveDeletedFiles": "string",
      "PreserveDevices": "string",
      "SecurityDescriptorCopyFlags": "string",
      "TaskQueueing": "string",
      "TransferMode": "string",
      "Uid": "string",
      "VerifyMode": "string"
   },
   "Schedule": {
      "ScheduleExpression": "string",
      "Status": "string"
   },
   "TaskArn": "string",
   "TaskReportConfig": {
      "Destination": {
         "S3": {
            "BucketAccessRoleArn": "string",
            "S3BucketArn": "string",
            "Subdirectory": "string"
         }
      },
      "ObjectVersionIds": "string",
      "OutputType": "string",
      "Overrides": {
         "Deleted": {
            "ReportLevel": "string"
         },
         "Skipped": {
            "ReportLevel": "string"
         },
         "Transferred": {
            "ReportLevel": "string"
         },
         "Verified": {
            "ReportLevel": "string"
         }
      },
      "ReportLevel": "string"
   }
}
```

### **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### CloudWatchLogGroupArn

Specifies the Amazon Resource Name (ARN) of an Amazon CloudWatch log group for monitoring your task.

For Enhanced mode tasks, you must use /aws/datasync as your log group name. For example:

```
arn:aws:logs:us-east-1:111222333444:log-group:/aws/datasync:*
```

For more information, see Monitoring data transfers with CloudWatch Logs.

Type: String

Length Constraints: Maximum length of 562.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):logs: <math>[a-z\-0-9]+: [0-9]{12}:log-group: ([^:\*]*)(:\*)?$
```

Required: No

### **Excludes**

Specifies exclude filters that define the files, objects, and folders in your source location that you don't want DataSync to transfer. For more information and examples, see <a href="Specifying what">Specifying what</a> <a href="DataSync transfers">DataSync transfers</a> by using filters.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

## **Includes**

Specifies include filters define the files, objects, and folders in your source location that you want DataSync to transfer. For more information and examples, see <a href="Specifying what DataSync transfers">Specifying what DataSync transfers</a> by using filters.

Type: Array of FilterRule objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

### ManifestConfig

Configures a manifest, which is a list of files or objects that you want DataSync to transfer. For more information and configuration examples, see <a href="Specifying what DataSync transfers by using">Specifying what DataSync transfers by using a manifest</a>.

When using this parameter, your caller identity (the IAM role that you're using DataSync with) must have the iam: PassRole permission. The <u>AWSDataSyncFullAccess</u> policy includes this permission.

To remove a manifest configuration, specify this parameter as empty.

Type: ManifestConfig object

Required: No

### Name

Specifies the name of your task.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

Required: No

## **Options**

Indicates how your transfer task is configured. These options include how DataSync handles files, objects, and their associated metadata during your transfer. You also can specify how to verify data integrity, set bandwidth limits for your task, among other options.

Each option has a default value. Unless you need to, you don't have to configure any option before calling <a href="StartTaskExecution">StartTaskExecution</a>.

You also can override your task options for each task execution. For example, you might want to adjust the LogLevel for an individual execution.

Type: Options object

Required: No

### **Schedule**

Specifies a schedule for when you want your task to run. For more information, see <u>Scheduling</u> your task.

Type: <u>TaskSchedule</u> object

Required: No

### **TaskArn**

Specifies the ARN of the task that you want to update.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]\*:[0-9]{12}: task/task-[0-9a-f]{17}\$

Required: Yes

## **TaskReportConfig**

Specifies how you want to configure a task report, which provides detailed information about your DataSync transfer. For more information, see <a href="Monitoring your DataSync transfers with task reports">Monitoring your DataSync transfers with task reports</a>.

When using this parameter, your caller identity (the IAM role that you're using DataSync with) must have the iam: PassRole permission. The <u>AWSDataSyncFullAccess</u> policy includes this permission.

To remove a task report configuration, specify this parameter as empty.

Type: <u>TaskReportConfig</u> object

Required: No

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see Common Errors.

### InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

#### InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **UpdateTaskExecution**

Updates the configuration of a running AWS DataSync task execution.



### Note

Currently, the only Option that you can modify with UpdateTaskExecution is BytesPerSecond, which throttles bandwidth for a running or queued task execution.

### **Request Syntax**

```
{
   "Options": {
      "Atime": "string",
      "Bytes<u>PerSecond</u>": number,
      "Gid": "string",
      "LogLevel": "string",
      "Mtime": "string",
      "ObjectTags": "string",
      "OverwriteMode": "string",
      "PosixPermissions": "string",
      "PreserveDeletedFiles": "string",
      "PreserveDevices": "string",
      "SecurityDescriptorCopyFlags": "string",
      "TaskQueueing": "string",
      "TransferMode": "string",
      "Uid": "string",
      "VerifyMode": "string"
   },
   "TaskExecutionArn": "string"
}
```

## **Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

UpdateTaskExecution 702

### **Options**

Indicates how your transfer task is configured. These options include how DataSync handles files, objects, and their associated metadata during your transfer. You also can specify how to verify data integrity, set bandwidth limits for your task, among other options.

Each option has a default value. Unless you need to, you don't have to configure any option before calling <a href="StartTaskExecution">StartTaskExecution</a>.

You also can override your task options for each task execution. For example, you might want to adjust the LogLevel for an individual execution.

Type: Options object

Required: Yes

### **TaskExecutionArn**

Specifies the Amazon Resource Name (ARN) of the task execution that you're updating.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z \ -0-9]*:[0-9]{12}: task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$$ 

Required: Yes

## **Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

#### **Errors**

For information about the errors that are common to all actions, see **Common Errors**.

## InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

UpdateTaskExecution 703

## InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

### **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# **Data Types**

The following data types are supported:

- AgentListEntry
- AzureBlobSasConfiguration
- CmkSecretConfig
- CustomSecretConfig
- Ec2Config
- FilterRule
- FsxProtocol
- FsxProtocolNfs

Data Types 704

- FsxProtocolSmb
- FsxUpdateProtocol
- FsxUpdateProtocolSmb
- HdfsNameNode
- LocationFilter
- LocationListEntry
- ManagedSecretConfig
- ManifestConfig
- NfsMountOptions
- OnPremConfig
- Options
- Platform
- PrivateLinkConfig
- QopConfiguration
- ReportDestination
- ReportDestinationS3
- ReportOverride
- ReportOverrides
- ReportResult
- S3Config
- S3ManifestConfig
- SmbMountOptions
- SourceManifestConfig
- TagListEntry
- TaskExecutionFilesFailedDetail
- TaskExecutionFilesListedDetail
- TaskExecutionListEntry
- TaskExecutionResultDetail
- TaskFilter
- TaskListEntry

Data Types 705

- TaskReportConfig
- TaskSchedule

• TaskScheduleDetails

Data Types 706

# AgentListEntry

Represents a single entry in a list (or array) of AWS DataSync agents when you call the <u>ListAgents</u> operation.

### **Contents**

# **AgentArn**

The Amazon Resource Name (ARN) of a DataSync agent.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}\$

Required: No

#### Name

The name of an agent.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

Required: No

#### **Platform**

The platform-related details about the agent, such as the version number.

Type: <u>Platform</u> object

Required: No

#### **Status**

The status of an agent.

• If the status is ONLINE, the agent is configured properly and ready to use.

AgentListEntry 707

• If the status is OFFLINE, the agent has been out of contact with DataSync for five minutes or longer. This can happen for a few reasons. For more information, see What do I do if my agent is offline?

Type: String

Valid Values: ONLINE | OFFLINE

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AgentListEntry 708

# AzureBlobSasConfiguration

The shared access signature (SAS) configuration that allows AWS DataSync to access your Microsoft Azure Blob Storage.

For more information, see SAS tokens for accessing your Azure Blob Storage.

### **Contents**

#### **Token**

Specifies a SAS token that provides permissions to access your Azure Blob Storage.

The token is part of the SAS URI string that comes after the storage resource URI and a question mark. A token looks something like this:

```
sp=r&st=2023-12-20T14:54:52Z&se=2023-12-20T22:54:52Z&spr=https&sv=2021-06-08&s%2FXTI9E%2F%2Fmq171%2BZU178wcwqU%3D
```

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^.+\$

Required: Yes

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AzureBlobSasConfiguration 709

# CmkSecretConfig

Specifies configuration information for a DataSync-managed secret, such as an authentication token or secret key that DataSync uses to access a specific storage location, with a customermanaged AWS KMS key.



### Note

You can use either CmkSecretConfig or CustomSecretConfig to provide credentials for a CreateLocation request. Do not provide both parameters for the same request.

# Contents

# **KmsKeyArn**

Specifies the ARN for the customer-managed AWS KMS key that DataSync uses to encrypt the DataSync-managed secret stored for SecretArn. DataSync provides this key to AWS Secrets Manager.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^(arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):kms:[a-z\-0-9]+:  $[0-9]{12}: \text{key}/.*|)$ \$

Required: No

#### SecretArn

Specifies the ARN for the DataSync-managed AWS Secrets Manager secret that that is used to access a specific storage location. This property is generated by DataSync and is read-only. DataSync encrypts this secret with the KMS key that you specify for KmsKeyArn.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^(arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):secretsmanager:[a $z - 0-9 + :[0-9] {12} : secret : .*|)$ 

CmkSecretConfig 710

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CmkSecretConfig 711

# CustomSecretConfig

Specifies configuration information for a customer-managed Secrets Manager secret where a storage location authentication token or secret key is stored in plain text. This configuration includes the secret ARN, and the ARN for an IAM role that provides access to the secret.



# Note

You can use either CmkSecretConfig or CustomSecretConfig to provide credentials for a CreateLocation request. Do not provide both parameters for the same request.

#### Contents

### SecretAccessRoleArn

Specifies the ARN for the AWS Identity and Access Management role that DataSync uses to access the secret specified for SecretArn.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^(arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9] {12}:role/[a-zA-Z0-9+=,.@\_-]+|)\$

Required: No

#### SecretArn

Specifies the ARN for an AWS Secrets Manager secret.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^(arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):secretsmanager:[a $z - 0-9 + [0-9] {12} : secret : .*|)$ 

Required: No

CustomSecretConfig 712

# **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CustomSecretConfig 713

# **Ec2Config**

The subnet and security groups that AWS DataSync uses to connect to one of your Amazon EFS file system's mount targets.

### **Contents**

# **SecurityGroupArns**

Specifies the Amazon Resource Names (ARNs) of the security groups associated with an Amazon EFS file system's mount target.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]*:
[0-9]{12}:security-group/sg-[a-f0-9]+$
```

Required: Yes

#### SubnetArn

Specifies the ARN of a subnet where DataSync creates the network interfaces for managing traffic during your transfer.

The subnet must be located:

- In the same virtual private cloud (VPC) as the Amazon EFS file system.
- In the same Availability Zone as at least one mount target for the Amazon EFS file system.



# Note

You don't need to specify a subnet that includes a file system mount target.

Type: String

Length Constraints: Maximum length of 128.

Ec2Config 714

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]*: [0-9]{12}:subnet/.*$
```

Required: Yes

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Ec2Config 715

# **FilterRule**

Specifies which files, folders, and objects to include or exclude when transferring files from source to destination.

### **Contents**

# FilterType

The type of filter rule to apply. AWS DataSync only supports the SIMPLE\_PATTERN rule type.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^[A-Z0-9\_]+\$

Valid Values: SIMPLE\_PATTERN

Required: No

#### Value

A single filter string that consists of the patterns to include or exclude. The patterns are delimited by "|" (that is, a pipe), for example: /folder1|/folder2

Type: String

Length Constraints: Maximum length of 102400.

Pattern:  $^[^\x00]+$ \$

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

FilterRule 716

• AWS SDK for Ruby V3

FilterRule 717

# **FsxProtocol**

Specifies the data transfer protocol that AWS DataSync uses to access your Amazon FSx file system.

# **Contents**

### **NFS**

Specifies the Network File System (NFS) protocol configuration that DataSync uses to access your FSx for OpenZFS file system or FSx for ONTAP file system's storage virtual machine (SVM).

Type: FsxProtocolNfs object

Required: No

#### **SMB**

Specifies the Server Message Block (SMB) protocol configuration that DataSync uses to access your FSx for ONTAP file system's SVM.

Type: FsxProtocolSmb object

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FsxProtocol 718

# **FsxProtocolNfs**

Specifies the Network File System (NFS) protocol configuration that DataSync uses to access your FSx for OpenZFS file system or FSx for ONTAP file system's storage virtual machine (SVM).

# **Contents**

# **MountOptions**

Specifies how DataSync can access a location using the NFS protocol.

Type: NfsMountOptions object

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FsxProtocolNfs 719

# **FsxProtocolSmb**

Specifies the Server Message Block (SMB) protocol configuration that AWS DataSync uses to access your Amazon FSx for NetApp ONTAP file system's storage virtual machine (SVM). For more information, see Providing DataSync access to FSx for ONTAP file systems.

### **Contents**

### **Password**

Specifies the password of a user who has permission to access your SVM.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^. {0, 104}\$

Required: Yes

#### User

Specifies a user that can mount and access the files, folders, and metadata in your SVM.

For information about choosing a user with the right level of access for your transfer, see <u>Using</u> the SMB protocol.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

Required: Yes

### **Domain**

Specifies the name of the Windows domain that your storage virtual machine (SVM) belongs to.

If you have multiple domains in your environment, configuring this setting makes sure that DataSync connects to the right SVM.

If you have multiple Active Directory domains in your environment, configuring this parameter makes sure that DataSync connects to the right SVM.

FsxProtocolSmb 720

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^[A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\}$ \$

Required: No

# **MountOptions**

Specifies the version of the Server Message Block (SMB) protocol that AWS DataSync uses to access an SMB file server.

Type: <u>SmbMountOptions</u> object

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FsxProtocolSmb 721

# **FsxUpdateProtocol**

Specifies the data transfer protocol that AWS DataSync uses to access your Amazon FSx file system.



# Note

You can't update the Network File System (NFS) protocol configuration for FSx for ONTAP locations. DataSync currently only supports NFS version 3 with this location type.

# **Contents**

#### NFS

Specifies the Network File System (NFS) protocol configuration that DataSync uses to access your FSx for OpenZFS file system or FSx for ONTAP file system's storage virtual machine (SVM).

Type: FsxProtocolNfs object

Required: No

### **SMB**

Specifies the Server Message Block (SMB) protocol configuration that DataSync uses to access your FSx for ONTAP file system's storage virtual machine (SVM).

Type: FsxUpdateProtocolSmb object

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FsxUpdateProtocol 722

# **FsxUpdateProtocolSmb**

Specifies the Server Message Block (SMB) protocol configuration that AWS DataSync uses to access your Amazon FSx for NetApp ONTAP file system's storage virtual machine (SVM). For more information, see Providing DataSync access to FSx for ONTAP file systems.

# **Contents**

### **Domain**

Specifies the name of the Windows domain that your storage virtual machine (SVM) belongs to.

If you have multiple Active Directory domains in your environment, configuring this parameter makes sure that DataSync connects to the right SVM.

Type: String

Length Constraints: Maximum length of 253.

Pattern:  $^([A-Za-z0-9]((\.|-+)?[A-Za-z0-9])\{0,252\})?$ \$

Required: No

# **MountOptions**

Specifies the version of the Server Message Block (SMB) protocol that AWS DataSync uses to access an SMB file server.

Type: <u>SmbMountOptions</u> object

Required: No

#### **Password**

Specifies the password of a user who has permission to access your SVM.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^. {0,104}\$

Required: No

FsxUpdateProtocolSmb 723

#### User

Specifies a user that can mount and access the files, folders, and metadata in your SVM.

For information about choosing a user with the right level of access for your transfer, see <u>Using</u> the SMB protocol.

Type: String

Length Constraints: Maximum length of 104.

Pattern:  $^[^\x22\x5B\x5D/\:;|=,+*?\x3C\x3E]{1,104}$ \$

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FsxUpdateProtocolSmb 724

# HdfsNameNode

The NameNode of the Hadoop Distributed File System (HDFS). The NameNode manages the file system's namespace. The NameNode performs operations such as opening, closing, and renaming files and directories. The NameNode contains the information to map blocks of data to the DataNodes.

### **Contents**

#### Hostname

The hostname of the NameNode in the HDFS cluster. This value is the IP address or Domain Name Service (DNS) name of the NameNode. An agent that's installed on-premises uses this hostname to communicate with the NameNode in the network.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern:  $^(([a-zA-Z0-9])^*[a-zA-Z0-9])).)*([A-Za-z0-9])$ 

Required: Yes

#### **Port**

The port that the NameNode uses to listen to client requests.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 65536.

Required: Yes

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

HdfsNameNode 725

HdfsNameNode 726

# LocationFilter

Narrow down the list of resources returned by ListLocations. For example, to see all your Amazon S3 locations, create a filter using "Name": "LocationType", "Operator": "Equals", and "Values": "S3".

For more information, see filtering resources.

#### **Contents**

#### Name

The name of the filter being used. Each API call supports a list of filters that are available for it (for example, LocationType for ListLocations).

Type: String

Valid Values: LocationUri | LocationType | CreationTime

Required: Yes

# Operator

The operator that is used to compare filter values (for example, Equals or Contains).

Type: String

Valid Values: Equals | NotEquals | In | LessThanOrEqual | LessThan |
GreaterThanOrEqual | GreaterThan | Contains | NotContains | BeginsWith

Required: Yes

#### **Values**

The values that you want to filter for. For example, you might want to display only Amazon S3 locations.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Required: Yes

LocationFilter 727

# **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

LocationFilter 728

# LocationListEntry

Represents a single entry in a list of locations. LocationListEntry returns an array that contains a list of locations when the ListLocations operation is called.

#### **Contents**

#### LocationArn

The Amazon Resource Name (ARN) of the location. For Network File System (NFS) or Amazon EFS, the location is the export path. For Amazon S3, the location is the prefix path that you want to mount and use as the root of the location.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: <math>[a-z]-0-9]+: [0-9]{12}: location/loc-[0-9a-z]{17}\$

Required: No

#### LocationUri

Represents a list of URIs of a location. LocationUri returns an array that contains a list of locations when the ListLocations operation is called.

Format: TYPE://GLOBAL\_ID/SUBDIR.

TYPE designates the type of location (for example, nfs or s3).

GLOBAL\_ID is the globally unique identifier of the resource that backs the location. An example for EFS is us-east-2.fs-abcd1234. An example for Amazon S3 is the bucket name, such as myBucket. An example for NFS is a valid IPv4 address or a hostname that is compliant with Domain Name Service (DNS).

SUBDIR is a valid file system path, delimited by forward slashes as is the \*nix convention. For NFS and Amazon EFS, it's the export path to mount the location. For Amazon S3, it's the prefix path that you mount to and treat as the root of the location.

Type: String

LocationListEntry 729

Length Constraints: Maximum length of 4360.

Pattern:  $^(efs|nfs|s3|smb|hdfs|fsx[a-z0-9-]+)://[a-zA-Z0-9.:/\-]+$ \$

Required: No

# **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

LocationListEntry 730

# ManagedSecretConfig

Specifies configuration information for a DataSync-managed secret, such as an authentication token or set of credentials that DataSync uses to access a specific transfer location. DataSync uses the default AWS-managed KMS key to encrypt this secret in AWS Secrets Manager.

### **Contents**

#### SecretArn

Specifies the ARN for an AWS Secrets Manager secret.

Type: String

Length Constraints: Maximum length of 2048.

```
Pattern: ^(arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):secretsmanager:[a-z\-0-9]+:[0-9]{12}:secret:.*|)$
```

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ManagedSecretConfig 731

# ManifestConfig

Configures a manifest, which is a list of files or objects that you want AWS DataSync to transfer. For more information and configuration examples, see Specifying what DataSync transfers by using a manifest.

# **Contents**

#### **Action**

Specifies what DataSync uses the manifest for.

Type: String

Valid Values: TRANSFER

Required: No

#### **Format**

Specifies the file format of your manifest. For more information, see Creating a manifest.

Type: String

Valid Values: CSV

Required: No

# **Source**

Specifies the manifest that you want DataSync to use and where it's hosted.



# Note

You must specify this parameter if you're configuring a new manifest on or after February 7, 2024.

If you don't, you'll get a 400 status code and ValidationException error stating that you're missing the IAM role for DataSync to access the S3 bucket where you're hosting your manifest. For more information, see Providing DataSync access to your manifest.

Type: SourceManifestConfig object

ManifestConfig 732

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ManifestConfig 733

# **NfsMountOptions**

Specifies how DataSync can access a location using the NFS protocol.

# **Contents**

### Version

Specifies the NFS version that you want DataSync to use when mounting your NFS share. If the server refuses to use the version specified, the task fails.

You can specify the following options:

- AUTOMATIC (default): DataSync chooses NFS version 4.1.
- NFS3: Stateless protocol version that allows for asynchronous writes on the server.
- NFSv4\_0: Stateful, firewall-friendly protocol version that supports delegations and pseudo file systems.
- NFSv4\_1: Stateful protocol version that supports sessions, directory delegations, and parallel data processing. NFS version 4.1 also includes all features available in version 4.0.

### Note

DataSync currently only supports NFS version 3 with Amazon FSx for NetApp ONTAP locations.

Type: String

Valid Values: AUTOMATIC | NFS3 | NFS4\_0 | NFS4\_1

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

NfsMountOptions 734

• AWS SDK for Ruby V3

NfsMountOptions 735

# **OnPremConfig**

The AWS DataSync agents that can connect to your Network File System (NFS) file server.

### **Contents**

# **AgentArns**

The Amazon Resource Names (ARNs) of the DataSync agents that can connect to your NFS file server.

You can specify more than one agent. For more information, see <u>Using multiple DataSync</u> agents.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync: [a-z \\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$
```

Required: Yes

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

OnPremConfig 736

# **Options**

Indicates how your transfer task is configured. These options include how DataSync handles files, objects, and their associated metadata during your transfer. You also can specify how to verify data integrity, set bandwidth limits for your task, among other options.

Each option has a default value. Unless you need to, you don't have to configure any option before calling StartTaskExecution.

You also can override your task options for each task execution. For example, you might want to adjust the LogLevel for an individual execution.

#### **Contents**

#### **Atime**

Specifies whether to preserve metadata indicating the last time a file was read or written to.



#### Note

The behavior of Atime isn't fully standard across platforms, so DataSync can only do this on a best-effort basis.

- BEST\_EFFORT (default) DataSync attempts to preserve the original Atime attribute on all source files (that is, the version before the PREPARING steps of the task execution). This option is recommended.
- NONE Ignores Atime.



# Note

If Atime is set to BEST\_EFFORT, Mtime must be set to PRESERVE. If Atime is set to NONE, Mtime must also be NONE.

Type: String

Valid Values: NONE | BEST\_EFFORT

Required: No

# **BytesPerSecond**

Limits the bandwidth used by a DataSync task. For example, if you want DataSync to use a maximum of 1 MB, set this value to 1048576 (=1024\*1024).



### Note

Not applicable to Enhanced mode tasks.

Type: Long

Valid Range: Minimum value of -1.

Required: No

#### Gid

Specifies the POSIX group ID (GID) of the file's owners.

- INT\_VALUE (default) Preserves the integer value of user ID (UID) and GID, which is recommended.
- NONE Ignores UID and GID.

For more information, see Understanding how DataSync handles file and object metadata.

Type: String

Valid Values: NONE | INT\_VALUE | NAME | BOTH

Required: No

# LogLevel

Specifies the type of logs that DataSync publishes to a Amazon CloudWatch Logs log group. To specify the log group, see CloudWatchLogGroupArn.

- BASIC Publishes logs with only basic information (such as transfer errors).
- TRANSFER Publishes logs for all files or objects that your DataSync task transfers and performs data-integrity checks on.
- OFF No logs are published.

Type: String

Valid Values: OFF | BASIC | TRANSFER

Required: No

### Mtime

Specifies whether to preserve metadata indicating the last time that a file was written to before the PREPARING step of your task execution. This option is required when you need to run the a task more than once.

- PRESERVE (default) Preserves original Mtime, which is recommended.
- NONE Ignores Mtime.



# Note

If Mtime is set to PRESERVE, Atime must be set to BEST\_EFFORT. If Mtime is set to NONE, Atime must also be set to NONE.

Type: String

Valid Values: NONE | PRESERVE

Required: No

# **ObjectTags**

Specifies whether you want DataSync to PRESERVE object tags (default behavior) when transferring between object storage systems. If you want your DataSync task to ignore object tags, specify the NONE value.

Type: String

Valid Values: PRESERVE | NONE

Required: No

# OverwriteMode

Specifies whether DataSync should modify or preserve data at the destination location.

 ALWAYS (default) - DataSync modifies data in the destination location when source data (including metadata) has changed.

If DataSync overwrites objects, you might incur additional charges for certain Amazon S3 storage classes (for example, for retrieval or early deletion). For more information, see Storage class considerations with Amazon S3 transfers.

 NEVER - DataSync doesn't overwrite data in the destination location even if the source data has changed. You can use this option to protect against overwriting changes made to files or objects in the destination.

Type: String

Valid Values: ALWAYS | NEVER

Required: No

#### **PosixPermissions**

Specifies which users or groups can access a file for a specific purpose such as reading, writing, or execution of the file.

For more information, see Understanding how DataSync handles file and object metadata.

- PRESERVE (default) Preserves POSIX-style permissions, which is recommended.
- NONE Ignores POSIX-style permissions.



### Note

DataSync can preserve extant permissions of a source location.

Type: String

Valid Values: NONE | PRESERVE

Required: No

#### **PreserveDeletedFiles**

Specifies whether files in the destination location that don't exist in the source should be preserved. This option can affect your Amazon S3 storage cost. If your task deletes objects,

you might incur minimum storage duration charges for certain storage classes. For detailed information, see Considerations when working with Amazon S3 storage classes in DataSync.

- PRESERVE (default) Ignores such destination files, which is recommended.
- REMOVE Deletes destination files that aren't present in the source.

### Note

If you set this parameter to REMOVE, you can't set TransferMode to ALL. When you transfer all data, DataSync doesn't scan your destination location and doesn't know what to delete.

Type: String

Valid Values: PRESERVE | REMOVE

Required: No

#### **PreserveDevices**

Specifies whether DataSync should preserve the metadata of block and character devices in the source location and recreate the files with that device name and metadata on the destination. DataSync copies only the name and metadata of such devices.



# Note

DataSync can't copy the actual contents of these devices because they're nonterminal and don't return an end-of-file (EOF) marker.

- NONE (default) Ignores special devices (recommended).
- PRESERVE Preserves character and block device metadata. This option currently isn't supported for Amazon EFS.

Type: String

Valid Values: NONE | PRESERVE

Required: No

# **SecurityDescriptorCopyFlags**

Specifies which components of the SMB security descriptor are copied from source to destination objects.

This value is only used for transfers between SMB and Amazon FSx for Windows File Server locations or between two FSx for Windows File Server locations. For more information, see Understanding how DataSync handles file and object metadata.

- OWNER\_DACL (default) For each copied object, DataSync copies the following metadata:
  - The object owner.
  - NTFS discretionary access control lists (DACLs), which determine whether to grant access to an object.

DataSync won't copy NTFS system access control lists (SACLs) with this option.

- OWNER\_DACL\_SACL For each copied object, DataSync copies the following metadata:
  - The object owner.
  - NTFS discretionary access control lists (DACLs), which determine whether to grant access to an object.
  - SACLs, which are used by administrators to log attempts to access a secured object.

Copying SACLs requires granting additional permissions to the Windows user that DataSync uses to access your SMB location. For information about choosing a user with the right permissions, see required permissions for <a href="SMB">SMB</a>, <a href="FSX for Windows File Server">FSX for Windows File Server</a>, or <a href="FSX for ONTAP">FSX for ONTAP</a> (depending on the type of location in your transfer).

• NONE - None of the SMB security descriptor components are copied. Destination objects are owned by the user that was provided for accessing the destination location. DACLs and SACLs are set based on the destination server's configuration.

Type: String

Valid Values: NONE | OWNER\_DACL | OWNER\_DACL\_SACL

Required: No

# **TaskQueueing**

Specifies whether your transfer tasks should be put into a queue during certain scenarios when running multiple tasks. This is ENABLED by default.

Type: String

Valid Values: ENABLED | DISABLED

Required: No

#### **TransferMode**

Specifies whether DataSync transfers only the data (including metadata) that differs between locations following an initial copy or transfers all data every time you run the task. If you're planning on recurring transfers, you might only want to transfer what's changed since your previous task execution.

- CHANGED (default) After your initial full transfer, DataSync copies only the data and metadata that differs between the source and destination location.
- ALL DataSync copies everything in the source to the destination without comparing differences between the locations.

Type: String

Valid Values: CHANGED | ALL

Required: No

#### Uid

Specifies the POSIX user ID (UID) of the file's owner.

- INT\_VALUE (default) Preserves the integer value of UID and group ID (GID), which is recommended.
- NONE Ignores UID and GID.

For more information, see Metadata copied by DataSync.

Type: String

Valid Values: NONE | INT\_VALUE | NAME | BOTH

Required: No

## VerifyMode

Specifies if and how DataSync checks the integrity of your data at the end of your transfer.

 ONLY\_FILES\_TRANSFERRED (recommended) - DataSync calculates the checksum of transferred data (including metadata) at the source location. At the end of the transfer,

Options 743

DataSync then compares this checksum to the checksum calculated on that data at the destination.



#### Note

This is the default option for Enhanced mode tasks.

We recommend this option when transferring to S3 Glacier Flexible Retrieval or S3 Glacier Deep Archive storage classes. For more information, see Storage class considerations with Amazon S3 locations.

 POINT\_IN\_TIME\_CONSISTENT - At the end of the transfer, DataSync checks the entire source and destination to verify that both locations are fully synchronized.



#### Note

The is the default option for Basic mode tasks and isn't currently supported with Enhanced mode tasks.

If you use a manifest, DataSync only scans and verifies what's listed in the manifest.

You can't use this option when transferring to S3 Glacier Flexible Retrieval or S3 Glacier Deep Archive storage classes. For more information, see Storage class considerations with Amazon S3 locations.

 NONE - DataSync performs data integrity checks only during your transfer. Unlike other options, there's no additional verification at the end of your transfer.

Type: String

Valid Values: POINT\_IN\_TIME\_CONSISTENT | ONLY\_FILES\_TRANSFERRED | NONE

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

Options 744

- AWS SDK for C++
- AWS SDK for Java V2

• AWS SDK for Ruby V3

Options 745

## **Platform**

The platform-related details about the AWS DataSync agent, such as the version number.

## **Contents**

## Version

The version of the DataSync agent.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Platform 746

# **PrivateLinkConfig**

Specifies how your AWS DataSync agent connects to AWS using a <u>virtual private cloud (VPC)</u> service endpoint. An agent that uses a VPC endpoint isn't accessible over the public internet.

#### **Contents**

## PrivateLinkEndpoint

Specifies the VPC endpoint provided by AWS PrivateLink that your agent connects to.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 15.

Pattern:  $A(25[0-5]|2[0-4]\d|[0-1]?\d?\d)(\.(25[0-5]|2[0-4]\d|[0-1]?\d?\d))$  {3}\z

Required: No

## **SecurityGroupArns**

Specifies the Amazon Resource Names (ARN) of the security group that provides DataSync access to your VPC endpoint. You can only specify one ARN.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2: <math>[a-z\-0-9]$ \*: [0-9]{12}: security-group/sg-[a-f0-9]+\$

Required: No

#### **SubnetArns**

Specifies the ARN of the subnet where your VPC endpoint is located. You can only specify one ARN.

Type: Array of strings

Array Members: Fixed number of 1 item.

PrivateLinkConfig 747

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn:  $(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]*: [0-9]{12}:subnet/.*$$ 

Required: No

## **VpcEndpointId**

Specifies the ID of the VPC endpoint that your agent connects to.

Type: String

Pattern: ^vpce-[0-9a-f]{17}\$

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PrivateLinkConfig 748

# **QopConfiguration**

The Quality of Protection (QOP) configuration specifies the Remote Procedure Call (RPC) and data transfer privacy settings configured on the Hadoop Distributed File System (HDFS) cluster.

## **Contents**

#### **DataTransferProtection**

The data transfer protection setting configured on the HDFS cluster. This setting corresponds to your dfs.data.transfer.protection setting in the hdfs-site.xml file on your Hadoop cluster.

Type: String

Valid Values: DISABLED | AUTHENTICATION | INTEGRITY | PRIVACY

Required: No

## **RpcProtection**

The RPC protection setting configured on the HDFS cluster. This setting corresponds to your hadoop.rpc.protection setting in your core-site.xml file on your Hadoop cluster.

Type: String

Valid Values: DISABLED | AUTHENTICATION | INTEGRITY | PRIVACY

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

QopConfiguration 749

# ReportDestination

Specifies where DataSync uploads your task report.

## **Contents**

**S3** 

Specifies the Amazon S3 bucket where DataSync uploads your task report.

Type: ReportDestinationS3 object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReportDestination 750

# ReportDestinationS3

Specifies the Amazon S3 bucket where DataSync uploads your task report.

#### **Contents**

#### **BucketAccessRoleArn**

Specifies the Amazon Resource Name (ARN) of the IAM policy that allows DataSync to upload a task report to your S3 bucket. For more information, see <u>Allowing DataSync to upload a task</u> report to an Amazon S3 bucket.

Type: String

Length Constraints: Maximum length of 2048.

```
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]
```

{12}:role/.\*\$

Required: Yes

#### S3BucketArn

Specifies the ARN of the S3 bucket where DataSync uploads your report.

Type: String

Length Constraints: Maximum length of 268.

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3: [a-z\-0-9]*: [0-9]{12}:accesspoint[/:][a-zA-Z0-9\-.]{1,63}$|^arn: <math>(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3-outposts: [a-z\-0-9]+:[0-9]{12}:outpost[/:][a-zA-Z0-9\-]{1,63}$|^arn: <math>(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3:::[a-zA-Z0-9\-]{1,255}$
```

Required: Yes

## Subdirectory

Specifies a bucket prefix for your report.

Type: String

Length Constraints: Maximum length of 4096.

ReportDestinationS3 751

Pattern:  $^[a-zA-Z0-9_{-}+\./\(\)p{Zs}]*$ \$

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReportDestinationS3 752

# **ReportOverride**

Specifies the level of detail for a particular aspect of your DataSync task report.

## **Contents**

## ReportLevel

Specifies whether your task report includes errors only or successes and errors.

For example, your report might mostly include only what didn't go well in your transfer (ERRORS\_ONLY). At the same time, you want to verify that your <u>task filter</u> is working correctly. In this situation, you can get a list of what files DataSync successfully skipped and if something transferred that you didn't to transfer (SUCCESSES\_AND\_ERRORS).

Type: String

Valid Values: ERRORS\_ONLY | SUCCESSES\_AND\_ERRORS

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReportOverride 753

# **ReportOverrides**

The level of detail included in each aspect of your DataSync task report.

#### **Contents**

#### **Deleted**

Specifies the level of reporting for the files, objects, and directories that DataSync attempted to delete in your destination location. This only applies if you <u>configure your task</u> to delete data in the destination that isn't in the source.

Type: ReportOverride object

Required: No

## Skipped

Specifies the level of reporting for the files, objects, and directories that DataSync attempted to skip during your transfer.

Type: ReportOverride object

Required: No

#### **Transferred**

Specifies the level of reporting for the files, objects, and directories that DataSync attempted to transfer.

Type: ReportOverride object

Required: No

## Verified

Specifies the level of reporting for the files, objects, and directories that DataSync attempted to verify at the end of your transfer.

Type: ReportOverride object

Required: No

ReportOverrides 754

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReportOverrides 755

# ReportResult

Indicates whether DataSync created a complete task report for your transfer.

## **Contents**

#### **ErrorCode**

Indicates the code associated with the error if DataSync can't create a complete report.

Type: String

Required: No

#### **ErrorDetail**

Provides details about issues creating a report.

Type: String

Required: No

#### **Status**

Indicates whether DataSync is still working on your report, created a report, or can't create a complete report.

Type: String

Valid Values: PENDING | SUCCESS | ERROR

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReportResult 756

# S3Config

Specifies the Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that DataSync uses to access your S3 bucket.

For more information, see Providing DataSync access to S3 buckets.

## **Contents**

#### BucketAccessRoleArn

Specifies the ARN of the IAM role that DataSync uses to access your S3 bucket.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]

{12}:role/.\*\$

Required: Yes

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

S3Config 757

# **S3ManifestConfig**

Specifies the S3 bucket where you're hosting the manifest that you want AWS DataSync to use. For more information and configuration examples, see <a href="Specifying what DataSync transfers by using a manifest">Specifying what DataSync transfers by using a manifest</a>.

## **Contents**

#### **BucketAccessRoleArn**

Specifies the AWS Identity and Access Management (IAM) role that allows DataSync to access your manifest. For more information, see Providing DataSync access to your manifest.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9] {12}:role/.\*\$

Required: Yes

## ManifestObjectPath

Specifies the Amazon S3 object key of your manifest. This can include a prefix (for example, prefix/my-manifest.csv).

Type: String

Length Constraints: Maximum length of 4096.

Pattern:  $^[a-zA-Z0-9_{-}+\./\(\)p{Zs}]*$$ 

Required: Yes

#### S3BucketArn

Specifies the Amazon Resource Name (ARN) of the S3 bucket where you're hosting your manifest.

Type: String

Length Constraints: Maximum length of 268.

S3ManifestConfig 758

```
Pattern: ^arn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3: [a-z\-0-9]*: [0-9]{12}:accesspoint[/:][a-zA-Z0-9\-.]{1,63}$|^arn: <math>(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3-outposts: [a-z\-0-9]+:[0-9]{12}:outpost[/:][a-zA-Z0-9\-]{1,63}[/:]accesspoint[/:][a-zA-Z0-9\-]{1,63}$|^arn: <math>(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3:::[a-zA-Z0-9\cdot\-_]{1,255}$
```

Required: Yes

## ManifestObjectVersionId

Specifies the object version ID of the manifest that you want DataSync to use. If you don't set this, DataSync uses the latest version of the object.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: ^.+\$

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

S3ManifestConfig 759

# **SmbMountOptions**

Specifies the version of the Server Message Block (SMB) protocol that AWS DataSync uses to access an SMB file server.

#### **Contents**

#### Version

By default, DataSync automatically chooses an SMB protocol version based on negotiation with your SMB file server. You also can configure DataSync to use a specific SMB version, but we recommend doing this only if DataSync has trouble negotiating with the SMB file server automatically.

These are the following options for configuring the SMB version:

• AUTOMATIC (default): DataSync and the SMB file server negotiate the highest version of SMB that they mutually support between 2.1 and 3.1.1.

This is the recommended option. If you instead choose a specific version that your file server doesn't support, you may get an Operation Not Supported error.

- SMB3: Restricts the protocol negotiation to only SMB version 3.0.2.
- SMB2: Restricts the protocol negotiation to only SMB version 2.1.
- SMB2\_0: Restricts the protocol negotiation to only SMB version 2.0.
- SMB1: Restricts the protocol negotiation to only SMB version 1.0.



## Note

The SMB1 option isn't available when creating an Amazon FSx for NetApp ONTAP location.

Type: String

Valid Values: AUTOMATIC | SMB2 | SMB3 | SMB1 | SMB2 0

Required: No

**SmbMountOptions** 760

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SmbMountOptions 761

# SourceManifestConfig

Specifies the manifest that you want AWS DataSync to use and where it's hosted. For more information and configuration examples, see <a href="Specifying what DataSync transfers by using a manifest">Specifying what DataSync transfers by using a manifest</a>.

## **Contents**

## **S3**

Specifies the S3 bucket where you're hosting your manifest.

Type: S3ManifestConfig object

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SourceManifestConfig 762

# **TagListEntry**

A key-value pair representing a single tag that's been applied to an AWS resource.

## **Contents**

## Key

The key for an AWS resource tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:/-]+\$

Required: Yes

#### Value

The value for an AWS resource tag.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=.\_:@/-]+\$

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TagListEntry 763

## **TaskExecutionFilesFailedDetail**

The number of objects that DataSync fails to prepare, transfer, verify, and delete during your task execution.



## Note

Applies only to Enhanced mode tasks.

## **Contents**

### **Delete**

The number of objects that DataSync fails to delete during your task execution.

Type: Long

Required: No

## **Prepare**

The number of objects that DataSync fails to prepare during your task execution.

Type: Long

Required: No

#### **Transfer**

The number of objects that DataSync fails to transfer during your task execution.

Type: Long

Required: No

## Verify

The number of objects that DataSync fails to verify during your task execution.

Type: Long

Required: No

TaskExecutionFilesFailedDetail 764

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskExecutionFilesFailedDetail 765

## **TaskExecutionFilesListedDetail**

The number of objects that DataSync finds at your locations.



#### Note

Applies only to Enhanced mode tasks.

#### Contents

#### **AtDestinationForDelete**

The number of objects that DataSync finds at your destination location. This counter is only applicable if you configure your task to delete data in the destination that isn't in the source.

Type: Long

Required: No

#### **AtSource**

The number of objects that DataSync finds at your source location.

- With a manifest, DataSync lists only what's in your manifest (and not everything at your source location).
- With an include filter, DataSync lists only what matches the filter at your source location.
- With an exclude filter, DataSync lists everything at your source location before applying the filter.

Type: Long

Required: No

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

TaskExecutionFilesListedDetail 766

• AWS SDK for Ruby V3

TaskExecutionFilesListedDetail 767

# **TaskExecutionListEntry**

Represents a single entry in a list of AWS DataSync task executions that's returned with the ListTaskExecutions operation.

#### **Contents**

#### **Status**

The status of a task execution. For more information, see Task execution statuses.

Type: String

Valid Values: QUEUED | CANCELLING | LAUNCHING | PREPARING | TRANSFERRING | VERIFYING | SUCCESS | ERROR

Required: No

#### **TaskExecutionArn**

The Amazon Resource Name (ARN) of a task execution.

Type: String

Length Constraints: Maximum length of 128.

Pattern:  $^a$ rn: (aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b): datasync: [a-z -0-9]\*: [0-9]{12}: task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}\$

Required: No

### **TaskMode**

The task mode that you're using. For more information, see <u>Choosing a task mode for your data transfer</u>.

Type: String

Valid Values: BASIC | ENHANCED

Required: No

TaskExecutionListEntry 768

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskExecutionListEntry 769

## **TaskExecutionResultDetail**

Provides detailed information about the result of your AWS DataSync task execution.

### **Contents**

#### **ErrorCode**

An error that DataSync encountered during your task execution. You can use this information to help troubleshoot issues.

Type: String

Required: No

### **ErrorDetail**

The detailed description of an error that DataSync encountered during your task execution. You can use this information to help troubleshoot issues.

Type: String

Required: No

## **PrepareDuration**

The time in milliseconds that your task execution was in the PREPARING step. For more information, see Task execution statuses.

For Enhanced mode tasks, the value is always 0. For more information, see <u>How DataSync</u> prepares your data transfer.

Type: Long

Valid Range: Minimum value of 0.

Required: No

## **PrepareStatus**

The status of the PREPARING step for your task execution. For more information, see <u>Task</u> execution statuses.

Type: String

TaskExecutionResultDetail 770

Valid Values: PENDING | SUCCESS | ERROR

Required: No

#### **TotalDuration**

The time in milliseconds that your task execution ran.

Type: Long

Valid Range: Minimum value of 0.

Required: No

## **TransferDuration**

The time in milliseconds that your task execution was in the TRANSFERRING step. For more information, see Task execution statuses.

For Enhanced mode tasks, the value is always 0. For more information, see <u>How DataSync</u> transfers your data.

Type: Long

Valid Range: Minimum value of 0.

Required: No

#### **TransferStatus**

The status of the TRANSFERRING step for your task execution. For more information, see <u>Task</u> execution statuses.

Type: String

Valid Values: PENDING | SUCCESS | ERROR

Required: No

## VerifyDuration

The time in milliseconds that your task execution was in the VERIFYING step. For more information, see Task execution statuses.

For Enhanced mode tasks, the value is always 0. For more information, see <u>How DataSync</u> verifies your data's integrity.

TaskExecutionResultDetail 771

Type: Long

Valid Range: Minimum value of 0.

Required: No

## **VerifyStatus**

The status of the VERIFYING step for your task execution. For more information, see <u>Task</u> execution statuses.

Type: String

Valid Values: PENDING | SUCCESS | ERROR

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskExecutionResultDetail 772

# **TaskFilter**

You can use API filters to narrow down the list of resources returned by ListTasks. For example, to retrieve all tasks on a source location, you can use ListTasks with filter name LocationId and Operator Equals with the ARN for the location.

For more information, see filtering DataSync resources.

#### **Contents**

#### Name

The name of the filter being used. Each API call supports a list of filters that are available for it. For example, LocationId for ListTasks.

Type: String

Valid Values: LocationId | CreationTime

Required: Yes

## Operator

The operator that is used to compare filter values (for example, Equals or Contains).

Type: String

Valid Values: Equals | NotEquals | In | LessThanOrEqual | LessThan |
GreaterThanOrEqual | GreaterThan | Contains | NotContains | BeginsWith

Required: Yes

#### **Values**

The values that you want to filter for. For example, you might want to display only tasks for a specific destination location.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Required: Yes

TaskFilter 773

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskFilter 774

# **TaskListEntry**

Represents a single entry in a list of tasks. TaskListEntry returns an array that contains a list of tasks when the <u>ListTasks</u> operation is called. A task includes the source and destination file systems to sync and the options to use for the tasks.

## **Contents**

#### Name

The name of the task.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern:  $^[a-zA-Z0-9\s+=._:e/-]+$$ 

Required: No

#### **Status**

The status of the task.

Type: String

Valid Values: AVAILABLE | CREATING | QUEUED | RUNNING | UNAVAILABLE

Required: No

#### **TaskArn**

The Amazon Resource Name (ARN) of the task.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z

\-0-9]\*:[0-9]{12}:task/task-[0-9a-f]{17}\$

Required: No

TaskListEntry 775

## **TaskMode**

The task mode that you're using. For more information, see <u>Choosing a task mode for your data</u> transfer.

Type: String

Valid Values: BASIC | ENHANCED

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskListEntry 776

# **TaskReportConfig**

Specifies how you want to configure a task report, which provides detailed information about for your AWS DataSync transfer.

For more information, see Task reports.

#### Contents

#### **Destination**

Specifies the Amazon S3 bucket where DataSync uploads your task report. For more information, see Task reports.

Type: ReportDestination object

Required: No

## **ObjectVersionIds**

Specifies whether your task report includes the new version of each object transferred into an S3 bucket. This only applies if you <u>enable versioning on your bucket</u>. Keep in mind that setting this to INCLUDE can increase the duration of your task execution.

Type: String

Valid Values: INCLUDE | NONE

Required: No

## OutputType

Specifies the type of task report that you want:

- SUMMARY\_ONLY: Provides necessary details about your task, including the number of files, objects, and directories transferred and transfer duration.
- STANDARD: Provides complete details about your task, including a full list of files, objects, and directories that were transferred, skipped, verified, and more.

Type: String

Valid Values: SUMMARY\_ONLY | STANDARD

Required: No

TaskReportConfig 777

#### **Overrides**

Customizes the reporting level for aspects of your task report. For example, your report might generally only include errors, but you could specify that you want a list of successes and errors just for the files that DataSync attempted to delete in your destination location.

Type: ReportOverrides object

Required: No

## ReportLevel

Specifies whether you want your task report to include only what went wrong with your transfer or a list of what succeeded and didn't.

- ERRORS\_ONLY: A report shows what DataSync was unable to transfer, skip, verify, and delete.
- SUCCESSES\_AND\_ERRORS: A report shows what DataSync was able and unable to transfer, skip, verify, and delete.

Type: String

Valid Values: ERRORS\_ONLY | SUCCESSES\_AND\_ERRORS

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskReportConfig 778

# **TaskSchedule**

Configures your AWS DataSync task to run on a schedule (at a minimum interval of 1 hour).

#### **Contents**

# ScheduleExpression

Specifies your task schedule by using a cron or rate expression.

Use cron expressions for task schedules that run on a specific time and day. For example, the following cron expression creates a task schedule that runs at 8 AM on the first Wednesday of every month:

```
cron(0 8 * * 3#1)
```

Use rate expressions for task schedules that run on a regular interval. For example, the following rate expression creates a task schedule that runs every 12 hours:

```
rate(12 hours)
```

For information about cron and rate expression syntax, see the <u>Amazon EventBridge User Guide</u>.

Type: String

Length Constraints: Maximum length of 256.

Required: Yes

#### **Status**

Specifies whether to enable or disable your task schedule. Your schedule is enabled by default, but there can be situations where you need to disable it. For example, you might need to pause a recurring transfer to fix an issue with your task or perform maintenance on your storage system.

DataSync might disable your schedule automatically if your task fails repeatedly with the same error. For more information, see TaskScheduleDetails.

Type: String

TaskSchedule 779

Valid Values: ENABLED | DISABLED

Required: No

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TaskSchedule 780

# **TaskScheduleDetails**

Provides information about your AWS DataSync task schedule.

# **Contents**

# DisabledBy

Indicates how your task schedule was disabled.

• USER - Your schedule was manually disabled by using the <a href="UpdateTask">UpdateTask</a> operation or DataSync console.

• SERVICE - Your schedule was automatically disabled by DataSync because the task failed repeatedly with the same error.

Type: String

Valid Values: USER | SERVICE

Required: No

#### DisabledReason

Provides a reason if the task schedule is disabled.

If your schedule is disabled by USER, you see a Manually disabled by user. message.

If your schedule is disabled by SERVICE, you see an error message to help you understand why the task keeps failing. For information on resolving DataSync errors, see <u>Troubleshooting issues</u> with DataSync transfers.

Type: String

Length Constraints: Maximum length of 8192.

Pattern: ^[\w\s., '?!:;\/=|<>()-]\*\$

Required: No

# StatusUpdateTime

Indicates the last time the status of your task schedule changed. For example, if DataSync automatically disables your schedule because of a repeated error, you can see when the schedule was disabled.

TaskScheduleDetails 781

Type: Timestamp

Required: No

#### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# **Common Errors**

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

# AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

#### **IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

#### **InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

#### **InvalidAction**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

Common Errors 782

#### **InvalidClientTokenId**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

#### **NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

# **OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

# RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

#### ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

# **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 400

#### ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

Common Errors 783

# **Common Parameters**

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see <u>Signing AWS API requests</u> in the *IAM User Guide*.

#### **Action**

The action to be performed.

Type: string

Required: Yes

#### Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

# X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

#### X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4\_request"). The value is expressed in the following format: access\_key/YYYYMMDD/region/service/aws4\_request.

For more information, see Create a signed AWS API request in the IAM User Guide.

Common Parameters 784

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

#### X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see <u>Elements of an AWS API request signature</u> in the *IAM User Guide*.

Type: string

Required: Conditional

# X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see <u>AWS services that work with IAM</u> in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

# X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Common Parameters 785

Required: Conditional

# X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see <a href="Create a signed AWS API request">Create a signed AWS API request</a> in the IAM User Guide.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Parameters 786

# **Document history**

The following table describes important additions to the AWS DataSync documentation. We also update the documentation frequently to address feedback that you send us.

To get notified about updates to this documentation, subscribe to the RSS feed.

Change	Description	Date
Discovery no longer available	As of May 20, 2025, Discovery is no longer an available DataSync feature.	May 20, 2025
AWS managed policy update - Update to existing policy	The AWSDataSyncFullAcc ess policy has updated permission statements that remove tagging condition s from the permissions DataSync uses to create Secrets Manager secrets.	May 13, 2025
AWS managed policy update - Update to existing policy	The AWSDataSyncFullAcc ess policy has new permissions that allow DataSync to work with AWS Secrets Manager.	May 7, 2025
AWS managed policy update - Update to existing policy	The AWSDataSyncFullAcc ess policy has new permissions that allow DataSync to work with AWS Secrets Manager.	April 23, 2025
AWS managed policy update - Update to existing policy	The AWSDataSyncFullAcc ess policy has new permissions that allow DataSync to work with AWS	April 23, 2025

	Secrets Manager and AWS Key Management Service.	
AWS managed policy update - Update to existing policy	The AWSDataSyncService RolePolicy has new permissions that allow DataSync to work with AWS Secrets Manager.	April 15, 2025
Performing large data migrations	Learn how to plan large- scale data migrations using DataSync for transferring files or objects from on-premis es or other cloud storage to AWS.	February 19, 2025
Support for Kerberos with SMB locations	DataSync can now use Kerberos authentication when connecting to Server Message Block (SMB) file servers.	January 28, 2025
New AWS Region	AWS DataSync is available for data transfers in the Mexico (Central) Region.	January 14, 2025
New AWS Region	AWS DataSync is available for data transfers in the Asia Pacific (Thailand) Region.	January 7, 2025
Update AWS storage locations	You can now update your Amazon S3, Amazon EFS, and Amazon FSx transfer locations	December 18, 2024
No longer supporting Snowball Edge	As of November 12, 2024, DataSync no longer supports AWS Snowball Edge.	November 13, 2024

New AWS managed policy	The DataSync service-linked role named AWSServic eRoleForDataSync uses a new managed policy named AWSDataSyncService RolePolicy .	October 30, 2024
Introducing Enhanced mode	With Enhanced mode, you can transfer virtually unlimited numbers of objects between Amazon S3 locations.	October 30, 2024
AWS managed policy updates - Update to an existing policy	The AWSDataSyncFullAcc ess policy has a new permission for services that work with DataSync.	October 30, 2024
Support for Azure storage general-purpose v1 accounts	DataSync can work with Azure storage general-purpose v1 accounts when transferring to or from Microsoft Azure Blob Storage.	October 4, 2024
New way to configure task schedules	You can configure your DataSync task schedules by using rate expressions.	August 22, 2024
New AWS Region	AWS DataSync is available for data transfers in the Asia Pacific (Malaysia) Region.	August 21, 2024
Support for agentless cross- Region transfers that include an opt-in Region	You no longer need a DataSync agent for transfers between AWS storage services when at least one storage location is in an opt- in AWS Region.	July 24, 2024

AWS managed policy updates - Update to an existing policy	The AWSDataSyncFullAcc ess policy has a new permission for services that work with DataSync.	July 22, 2024
Updated S3 cross-account tutorial	Removed some source account user permissions that are no longer required for this transfer.	June 10, 2024
New task execution status	The CANCELLING status indicates when a task execution is being cancelled.	May 15, 2024
New option for pausing task schedules	You can disable your AWS DataSync task schedule when you need to troubleshoot issues or perform storage system maintenance.	April 24, 2024
Updated TLS cipher for FIPS endpoints	AWS DataSync uses the TLS_AES_128_GCM_SH A256 (secp256r1) cipher for Federal Information Processin g Standard (FIPS) service endpoints.	April 22, 2024
AWS managed policy updates - Update to an existing policy	The AWSDataSyncFullAcc ess policy has a new permission for services that work with DataSync.	February 16, 2024
Transfer specific files or objects with a manifest	AWS DataSync can transfer a list of files or objects by using a manifest.	February 7, 2024

New AWS Region	AWS DataSync is now available for data transfers in the Canada West (Calgary) Region.	December 20, 2023
Support for transfers with additional cloud providers	AWS DataSync can now transfer data between AWS storage services and IBM Cloud Object Storage or Seagate Lyve Cloud.	November 7, 2023
Support for transfers with Alibaba Cloud Object Storage Service	AWS DataSync can now transfer data between AWS storage services and Alibaba Cloud Object Storage Service.	September 25, 2023
Support for task reports	Monitor your AWS DataSync transfers with task reports.	August 30, 2023
New AWS Region	AWS DataSync is now available for data transfers in the Israel (Tel Aviv) Region.	August 23, 2023
Support for transfers with additional cloud providers	AWS DataSync can now transfer data between AWS storage services and several other cloud providers (such as Wasabi Cloud Storage, DigitalOcean Spaces, and Oracle Cloud Infrastructure Object Storage).	August 8, 2023
General availability of Microsoft Azure Blob Storage support	AWS DataSync can now transfer objects to and from Microsoft Azure Blob Storage.	July 25, 2023

TLS 1.3 support	When transferring between storage locations, AWS DataSync now encrypts all network traffic with Transport Layer Security (TLS) 1.3.	June 28, 2023
New DataSync Discovery metrics	AWS DataSync Discovery can now tell you how many LUNs (logical unit numbers) are in a storage resource cluster, storage virtual machine (SVM), or volume.	June 28, 2023
New AWS Region	AWS DataSync is now available for data transfers in the Asia Pacific (Melbourne) Region.	May 24, 2023
Support for with S3 compatible storage on Snowball Edge	You can use AWS DataSync to transfer data between Amazon S3 compatible storage on AWS Snowball Edge and AWS storage services.	May 18, 2023
AWS managed policy updates - Update to an existing policy	The AWSDataSyncFullAcc ess policy has new permissions for services that work with DataSync.	May 2, 2023
General availability of AWS  DataSync Discovery	Use DataSync Discovery to help accelerate your migration to AWS.	April 25, 2023
Public preview release of Microsoft Azure Blob Storage support	AWS DataSync can now transfer objects from Microsoft Azure Blob Storage.	March 29, 2023

New IAM policy	To support the DataSync Discovery feature, DataSync uses the service-linked role named AWSServic eRoleForDataSyncDi scovery .	March 21, 2023
New AWS Regions	AWS DataSync is now available in the following AWS Regions: Asia Pacific (Hyderabad), Europe (Spain), and Europe (Zurich).	February 6, 2023
Using tags in task executions	You can now tag your AWS DataSync task executions.	December 16, 2022
Support for S3 Glacier Instant Retrieval	You can now transfer objects directly into the S3 Glacier Instant Retrieval storage class.	December 16, 2022
Copying object system metadata	AWS DataSync can now copy system metadata when transferring between an object storage system and Amazon S3.	December 16, 2022
New AWS Regions	AWS DataSync is now available in the China (Beijing) and China (Ningxia) Regions.	December 14, 2022
New AWS Region	AWS DataSync is now available in the Middle East (UAE) Region.	November 16, 2022
Support for self-signed certificates with object storage locations	AWS DataSync can connect to object storage locations that use self-signed or private certificates.	October 25, 2022

Get data compression information	AWS DataSync can provide the physical number of bytes transferred over the network after compression was applied.	October 25, 2022
Public preview release of AWS  DataSync Discovery	Use DataSync Discovery to help accelerate your migration to AWS.	September 21, 2022
New option for migrating data to or from Google Cloud Storage	You can transfer data to or from Google Cloud Storage by deploying an AWS DataSync agent in Google Cloud.	July 21, 2022
Support for Amazon FSx for NetApp ONTAP file systems	AWS DataSync can now transfer files and folders to and from FSx for ONTAP file systems.	June 28, 2022
New security options for Amazon EFS locations	AWS DataSync can access Amazon EFS file systems using TLS, access points, and IAM roles.	May 31, 2022
Migrating data to or from Google Cloud Storage and Azure Files	With AWS DataSync, you can transfer data to or from Google Cloud Storage and Azure Files. For more information, see <u>Creating a location for object storage</u> and <u>Creating a location for SMB</u> .	May 24, 2022

New AWS DataSync task setting	With the <b>Copy object tags</b> option, you can specify whether to maintain object tags when transferring between object storage systems.	May 5, 2022
New AWS Region	AWS DataSync is now available in the Asia Pacific (Jakarta) Region.	April 19, 2022
Support for Amazon FSx for OpenZFS file systems	AWS DataSync can now transfer files and folders to and from FSx for OpenZFS file systems.	April 5, 2022
Support for Amazon FSx for Lustre file systems	AWS DataSync can now transfer files and folders to and from FSx for Lustre file systems.	December 10, 2021
Support for Hadoop Distribut ed File Systems (HDFS)	AWS DataSync now supports transferring files and folders to and from HDFS clusters.	November 3, 2021
New AWS Region	AWS DataSync is now available in the Asia Pacific (Osaka) Region.	July 28, 2021
Fully automated transfers between AWS storage services	AWS DataSync can now transfer files or objects between Amazon S3, Amazon EFS, or FSx for Windows File Server with just a few clicks in the DataSync console.	November 9, 2020

Adjusting the network	
bandwidth used by a runnin	g
task	

AWS DataSync now enables customers to adjust the network bandwidth used by a running DataSync task. This helps to minimize impact on other users or applications when a task spans multiple days.

November 9, 2020

Enhanced support for onpremises DataSync virtual machine (VM) functions The AWS DataSync agent VM host console now supports enhanced functions, including activating an agent from the local console.

October 19, 2020

AWS DataSync can now transfer data to and from AWS Outposts

DataSync now supports transferring objects to and from Amazon S3 on AWS Outposts. September 30, 2020

Support for API filtering

AWS DataSync now supports filtering for the ListTasks and ListLocations API calls, enabling you to easily retrieve configuration of data transfer tasks by using filters such as the source or destinati on for the data transfer.

August 18, 2020

Support for copying data from your self-managed object storage

AWS DataSync now supports data transfer between self-managed object storage and Amazon S3, Amazon Elastic File System, or FSx for Windows File Server.

July 27, 2020

Support for Linux Kernel-ba sed Virtual Machine (KVM) and Microsoft Hyper-V hypervisors AWS DataSync now provides the ability to deploy on-premises agents on the KVM and Microsoft Hyper-V virtualization platforms, in addition to the existing VMware and Amazon EC2 options.

July 1, 2020

AWS DataSync can now automatically configure your Amazon CloudWatch Logs configuration

When using DataSync, you now have the option of automatically generating the CloudWatch log group and resource policy required to publish logs for your data transfer, simplifying task creation and monitoring setup.

July 1, 2020

AWS DataSync can now transfer data to and from AWS Snowball Edge DataSync now supports transferring files to and from AWS Snowball Edge, the smallest member of the AWS Snow Family of edge computing and data transfer devices. Snowball Edge is portable, ruggedized, and secure—small and light enough to fit in a backpack and able to withstand harsh environments.

June 17, 2020

**New AWS Region** 

AWS DataSync is now available in the Africa (Cape Town) Region and the Europe (Milan) Region.

June 16, 2020

	nced monitoring pilities with file-level ng	You can now enable detailed logging for files and objects copied between your NFS servers, SMB servers, Amazon S3 buckets, Amazon EFS file systems, and FSx for Windows File Server file systems.	April 24, 2020
betwe	ort for copying data een your SMB share and con FSx for Windows File	You can now copy data between your SMB share and FSx for Windows File Server.	January 24, 2020
Suppo	ort for scheduling tasks	You can now run tasks manually or schedule them to run based on a specified schedule.	November 20, 2019
New A	AWS Region	AWS DataSync is now available in the Asia Pacific (Hong Kong) Region, Asia Pacific (Mumbai) Region, Europe (Stockholm) Region, South America (São Paulo) Region, and AWS GovCloud (US-East) Region.	November 20, 2019
New A	AWS Region	AWS DataSync is now available in the Canada (Central) Region, Europe (London) Region, and Europe (Paris) Region.	October 2, 2019
	ort for Amazon S3 ge classes	You can now transfer objects directly into Amazon S3 storage classes.	September 24, 2019

New AWS Region	AWS DataSync is now available in the Middle East (Bahrain) Region.	August 28, 2019
Support for copying data between your Server Message Block (SMB) share and Amazon S3 or Amazon EFS	You can now copy data between your SMB file share and Amazon S3 or Amazon EFS.	August 22, 2019
Support for using virtual private cloud (VPC) endpoints	You can now create a private connection between your agent and AWS and run tasks in a private network. Doing this increases the security of your data as it's copied over the network.	August 5, 2019
Support for Federal Informati on Processing Standard (FIPS) endpoints	You can now use FIPS endpoints to create agents and run tasks.	August 5, 2019
New AWS Region	AWS DataSync is now available in the AWS GovCloud (US-West) Region.	June 11, 2019
Support for filtering	You can now apply filters to transfer only a subset of the files in your source location when you transfer data from your source to your destinati on location.	May 22, 2019
First release of AWS DataSync	General release of the AWS DataSync service.	November 26, 2018

# **AWS Glossary**

For the latest AWS terminology, see the <u>AWS glossary</u> in the *AWS Glossary Reference*.