aws

Administrator Guide

Amazon WorkMail

Version 1.0

Amazon WorkMail: Administrator Guide

Table of Contents

What is Amazon WorkMail? 1
Amazon WorkMail system requirements 1
Amazon WorkMail concepts 2
Related AWS services 3
Amazon WorkMail pricing 4
Resources 4
Prerequisites 6
Sign up for an AWS account 6
Create a user with administrative access 6
Grant IAM users permissions for Amazon WorkMail 8
Security 9
Data protection 10
How Amazon WorkMail uses AWS KMS 11
Identity and access management 20
Audience 21
Authenticating With identities 21
Managing access using policies 24
How Amazon WorkMail works with IAM 27
Identity-based policy examples 32
Troubleshooting 40
AWS managed policies 41
AmazonWorkMailFullAccess 42
AmazonWorkMailReadOnlyAccess 42
AmazonWorkMailEventsServiceRolePolicy 42
Policy updates 42
Using service-linked roles 43
Service-linked role permissions for Amazon WorkMail 44
Creating a service-linked role for Amazon WorkMail 44
Editing a service-linked role for Amazon WorkMail 45
Deleting a service-linked role for Amazon WorkMail 45
Supported Regions for Amazon WorkMail service-linked roles 46
Logging and monitoring 46
Monitoring with CloudWatch metrics 48
Monitoring Amazon WorkMail email event logs 51

Monitoring Amazon WorkMail audit logs ... 57
Using CloudWatch Insights with Amazon WorkMail ... 63
Logging Amazon WorkMail API calls with AWS CloudTrail ... 67
Enabling email event logging ... 71
Enabling audit logging ... 75
Compliance validation ... 89
Resilience ... 90
Infrastructure security ... 90
Getting started ... 92
Getting started with Amazon WorkMail ... 92
Step 1: Sign in to the Amazon WorkMail console ... 93
Step 2: Set up your Amazon WorkMail site ... 93
Step 3: Set up Amazon WorkMail user access ... 94
More resources ... 95
Migrating to Amazon WorkMail ... 95
Step 1: Create or enable users in Amazon WorkMail ... 95
Step 2: Migrate to Amazon WorkMail ... 95
Step 3: Complete the migration to Amazon WorkMail ... 96
Interoperability between Amazon WorkMail and Microsoft Exchange ... 96
Prerequisites ... 97
Add domains and enable mailboxes ... 98
Enable interoperability ... 99
Create service accounts in Microsoft Exchange and Amazon WorkMail ... 99
Limitations in interoperability mode ... 99
Configure availability settings on Amazon WorkMail ... 100
Configure an EWS-based availability provider ... 100
Configuring a Custom Availability Provider ... 102
Building a CAP Lambda function ... 102
Configure availability settings in Microsoft Exchange ... 111
Enable email routing between Microsoft Exchange and Amazon WorkMail users ... 111
Enable email routing for a user ... 112
Post setup configuration ... 113
Mail client configuration ... 114
Disabling interoperability mode and decommissioning your mail server ... 114
Troubleshooting ... 116
Amazon WorkMail quotas ... 116

Amazon WorkMail organization and user quotas ... 117
WorkMail organization setting quotas ... 119
Per-user quotas ... 119
Message quotas ... 120
Working with organizations ... 122
Creating an organization ... 122
Creating an organization ... 123
Viewing an organization's details ... 125
Integrating a WorkSpaces directory ... 125
Organization states and descriptions ... 126
Deleting an organization ... 126
Finding an email address ... 127
Working with organization settings ... 128
Enabling mailbox migration ... 128
Enabling journaling ... 128
Enabling interoperability ... 129
Enabling SMTP gateways ... 129
Managing email flows ... 130
Enforcing DMARC policies on incoming email ... 154
Tagging an organization ... 155
Working with access control rules ... 157
Creating access control rules ... 158
Editing access control rules ... 158
Testing access control rules ... 159
Deleting access control rules ... 160
Setting mailbox retention policies ... 160
Working with domains ... 162
Adding a domain ... 162
Removing a domain ... 166
Choosing the default domain ... 167
Verifying domains ... 167
Verifying TXT records and MX records with your DNS service ... 169
Troubleshooting domain verification ... 171
Enabling AutoDiscover to configure endpoints ... 173
AutoDiscover phase 2 troubleshooting ... 176
Editing domain identity policies ... 178

Custom Amazon SES service-principal policy ... 179
Authenticating email with SPF ... 180
Configuring a custom MAIL FROM domain ... 180
Working with users ... 182
Viewing a list of users ... 182
Adding a user ... 183
Enabling users ... 184
Managing user aliases ... 184
Disabling users ... 185
Editing user details ... 186
Resetting user password ... 189
Troubleshooting Amazon WorkMail password policies ... 190
Working with notifications ... 191
Enabling signed or encrypted email ... 195
Working with groups ... 197
Viewing a list of groups ... 197
Adding a group ... 198
Enabling groups ... 199
Adding members to a group ... 199
Editing group details ... 200
Removing members from a group ... 201
Managing group aliases ... 201
Disabling groups ... 202
Deleting a group ... 203
Working with resources ... 204
Viewing a list of resources ... 204
Adding a resource ... 205
Editing resource details ... 205
Managing resource aliases ... 207
Enabling a resource ... 209
Disabling a resource ... 209
Deleting a resource ... 210
Working with IAM Identity Center ... 211
Enabling IAM Identity Center in Amazon WorkMail ... 213
Assigning IAM Identity Center users and groups to Amazon WorkMail application ... 213
Associating Amazon WorkMail users with IAM Identity Center users ... 215

Authentication mode ... 217
Configuring personal access tokens ... 218
Disabling IAM Identity Center ... 219
Working with mobile devices ... 221
Editing your organization's mobile device policy ... 221
Managing mobile devices ... 222
Remotely wiping mobile devices ... 222
Removing user devices from the devices list ... 223
Viewing mobile device details ... 224
Managing mobile device access rules ... 225
How mobile device access rules work ... 226
Using mobile device access rules ... 227
Managing mobile device access overrides ... 229
How mobile device access overrides work ... 229
Managing overrides ... 230
Integrating with mobile device management solutions ... 231
Mobile device management solutions overview ... 231
Configuring a WorkMail organization to integrate with a third-party MDM solution in direct mode ... 233
Working with mailbox permissions ... 235
About mailbox and folder permissions ... 236
Managing mailbox permissions for users ... 236
Adding permissions ... 237
Editing mailbox permissions for users ... 237
Managing mailbox permissions for groups ... 239
Programmatic access to mailboxes ... 240
Managing impersonation roles ... 240
Impersonation roles overview ... 240
Security considerations ... 241
Creating impersonation roles ... 242
Editing impersonation roles ... 243
Testing impersonation roles ... 244
Deleting impersonation roles ... 245
Using impersonation roles ... 245
Exporting mailbox content ... 248
Prerequisites ... 248

IAM policy examples and role creation 249
Example: Exporting mailbox content 251
Considerations 252
Troubleshooting 176
Viewing email headers 253
Mail routing 253
Using email journaling with Amazon WorkMail 255
Using journaling 255
Document history 257

What is Amazon WorkMail?

Amazon WorkMail is a secure, managed business email and calendaring service with support for existing desktop and mobile email clients. Amazon WorkMail users can access their email, contacts, and calendars using Microsoft Outlook, their browser, or their native iOS and Android email applications. You can integrate Amazon WorkMail with your existing corporate directory and control both the keys that encrypt your data and the location in which your data is stored.

For a list of supported AWS Regions and endpoints, see AWS Regions and Endpoints.

Topics

- Amazon WorkMail system requirements
- Amazon WorkMail concepts
- Related AWS services
- Amazon WorkMail pricing
- Amazon WorkMail resources

Amazon WorkMail system requirements

When your Amazon WorkMail administrator invites you to sign in to your Amazon WorkMail account, you can sign in using the Amazon WorkMail web client.

Amazon WorkMail also works with all major mobile devices and operating systems that support the Exchange ActiveSync protocol. These devices include the iPad, iPhone, Android, and Windows Phone. Users of macOS can add their Amazon WorkMail account to their Mail, Calendar, and Contacts apps.

Amazon WorkMail supports the following operating system versions:

- Windows – Windows 7 SP1 or later
- MacOS – MacOS 10.12 (Sierra) or later
- Android – Andriod 5.0 or later
- iPhone – iOS 5 or later
- Windows phone – Windows 8.1 or later
- Blackberry – Blackberry OS 10.3.3.3216

If you have a valid Microsoft Outlook license, you can access Amazon WorkMail using the following versions of Microsoft Outlook:

- Outlook 2013 or later
- Outlook 2013 Click-to-Run or later
- Outlook for Mac 2016 or later

You can access the Amazon WorkMail web client using the following browser versions:

- Google Chrome – Version 22 or later
- Mozilla Firefox – Version 27 or later
- Safari – Version 7 or later
- Internet Explorer – Version 11
- Microsoft Edge

You can also use Amazon WorkMail with your preferred IMAP client.

Amazon WorkMail concepts

The terminology and concepts that are central to your understanding and use of Amazon WorkMail are described below.

Organization

A tenant setup for Amazon WorkMail.

Alias

A globally unique name to identify your organization. The alias is used to access the Amazon WorkMail web application (https://*alias*.awsapps.com/mail).

Domain

The web address that comes after the **@** symbol in an email address. You can add a domain that receives mail and delivers it to mailboxes in your organization.

Test mail domain

A domain is automatically configured during setup that can be used for testing Amazon WorkMail. The test mail domain is *alias*.awsapps.com and is used as the default domain if you

don't configure your own domain. The test mail domain is subject to different limits. For more information, see Amazon WorkMail quotas.

Directory

An AWS Simple AD, AWS Managed AD, or AD Connector created in AWS Directory Service. If you create an organization using the Amazon WorkMail Quick setup, we create a WorkMail directory for you. You can't view a WorkMail directory in AWS Directory Service.

User

A user created in the AWS Directory Service. The user can be created in an *USER* or *REMOTE_USER* role. When a user is created and enabled with an *USER* role, they receive their own mailbox to access. When a user is disabled, they can't access Amazon WorkMail.

User created and enabled with a *REMOTE_USER* role is listed in the address book but do not get a mailbox in Amazon WorkMail. The *REMOTE_USER* can have the mailbox hosted outside Amazon WorkMail but will still be listed as any other user with mailbox in the Amazon WorkMail address book and can look up each others calendar to find free or busy information.

Group

A group used in AWS Directory Service. A group can be used as a distribution list or a security group in Amazon WorkMail. Groups don't have their own mailboxes.

Resource

A resource represents a meeting room or equipment resource that can be booked by Amazon WorkMail users.

Mobile device policy

Various IT policy rules that control the security features and behavior of a mobile device.

Related AWS services

The following services are used along with Amazon WorkMail:

- **AWS Directory Service**—You can integrate Amazon WorkMail with an existing AWS Simple AD, AWS Managed AD, or AD Connector. Create a directory in the AWS Directory Service and then enable Amazon WorkMail for this directory. After you've configured this integration, you can choose which users you would like to enable for Amazon WorkMail from a list of users in your

existing directory, and users can log in using their existing Active Directory credentials. For more information, see AWS Directory Service Administration Guide.

- **Amazon Simple Email Service**—Amazon WorkMail uses Amazon SES to send all outgoing email. The test mail domain and your domains are available for management in the Amazon SES console. There is no cost for outgoing email sent from Amazon WorkMail. For more information, see Amazon Simple Email Service Developer Guide.
- **AWS Identity and Access Management**—The AWS Management Console requires your user name and password so that any service you use can determine whether you have permission to access its resources. We recommend that you avoid using AWS account credentials to access AWS because AWS account credentials can't be revoked or limited in any way. Instead, we recommend that you create an IAM user and add the user to an IAM group with administrative permissions. You can then access the console using the IAM user credentials.

  If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM console. For more information, see Create individual IAM users in the *IAM User Guide*.
- **AWS Key Management Service**—Amazon WorkMail is integrated with AWS KMS for encryption of customer data. Key management can be performed from the AWS KMS console. For more information, see What is the AWS Key Management Service in the *AWS Key Management Service Developer Guide*.

Amazon WorkMail pricing

With Amazon WorkMail, there are no upfront fees or commitments. You pay only for active user accounts. For more specific information about pricing, see Pricing.

Amazon WorkMail resources

The following related resources can help you as you work with this service.

- Classes & Workshops – Links to role-based and specialty courses, in addition to self-paced labs to help sharpen your AWS skills and gain practical experience.
- AWS Developer Center – Explore tutorials, download tools, and learn about AWS developer events.
- AWS Developer Tools – Links to developer tools, SDKs, IDE toolkits, and command line tools for developing and managing AWS applications.

- Getting Started Resource Center – Learn how to set up your AWS account, join the AWS community, and launch your first application.
- Hands-On Tutorials – Follow step-by-step tutorials to launch your first application on AWS.
- AWS Whitepapers – Links to a comprehensive list of technical AWS whitepapers, covering topics such as architecture, security, and economics and authored by AWS Solutions Architects or other technical experts.
- AWS Support Center – The hub for creating and managing your AWS Support cases. Also includes links to other helpful resources, such as forums, technical FAQs, service health status, and AWS Trusted Advisor.
- Support – The primary webpage for information about Support, a one-on-one, fast-response support channel to help you build and run applications in the cloud.
- Contact Us – A central contact point for inquiries concerning AWS billing, account, events, abuse, and other issues.
- AWS Site Terms – Detailed information about our copyright and trademark; your account, license, and site access; and other topics.

Prerequisites

To act as an Amazon WorkMail administrator, you need an AWS account. If you haven't signed up for AWS yet, complete the following tasks to get set up.

Topics

- Sign up for an AWS account
- Create a user with administrative access
- Grant IAM users permissions for Amazon WorkMail

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 https://aws.amazon.com/ 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

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

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

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

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

Create a user with administrative access

1. Enable IAM Identity Center.

   For instructions, see Enabling AWS IAM Identity Center 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 Signing in to the AWS access portal 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*.

Grant IAM users permissions for Amazon WorkMail

By default, IAM users don't have permissions to manage Amazon WorkMail resources. You must attach an AWS managed policy (**AmazonWorkMailFullAccess** or **AmazonWorkMailReadOnlyAccess**) or create a customer-managed policy that explicitly grants IAM users those permissions. You then attach the policy to the IAM users or groups that require those permissions. For more information, see Identity and access management for Amazon WorkMail.

Security in Amazon WorkMail

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 shared responsibility model 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 AWS compliance programs. To learn about the compliance programs that apply to Amazon WorkMail, 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 Amazon WorkMail. The following topics show you how to configure Amazon WorkMail to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Amazon WorkMail resources.

Topics

- Data protection in Amazon WorkMail
- Identity and access management for Amazon WorkMail
- AWS managed policies for Amazon WorkMail
- Using service-linked roles for Amazon WorkMail
- Logging and monitoring in Amazon WorkMail
- Compliance validation for Amazon WorkMail
- Resilience in Amazon WorkMail
- Infrastructure security in Amazon WorkMail

Data protection in Amazon WorkMail

The AWS shared responsibility model applies to data protection in Amazon WorkMail. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail. For information about using CloudTrail trails to capture AWS activities, see Working with CloudTrail trails in the *AWS CloudTrail User Guide*.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-3 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-3.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with Amazon WorkMail or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

How Amazon WorkMail uses AWS KMS

Amazon WorkMail transparently encrypts all messages in the mailboxes of all Amazon WorkMail organizations before the messages are written to disk, and it transparently decrypts the messages when users access them. You can't disable encryption. To protect the encryption keys that protect the messages, Amazon WorkMail is integrated with AWS Key Management Service (AWS KMS).

Amazon WorkMail also provides an option for enabling users to send signed or encrypted email. This encryption feature does not use AWS KMS. For more information, see Enabling signed or encrypted email.

Topics

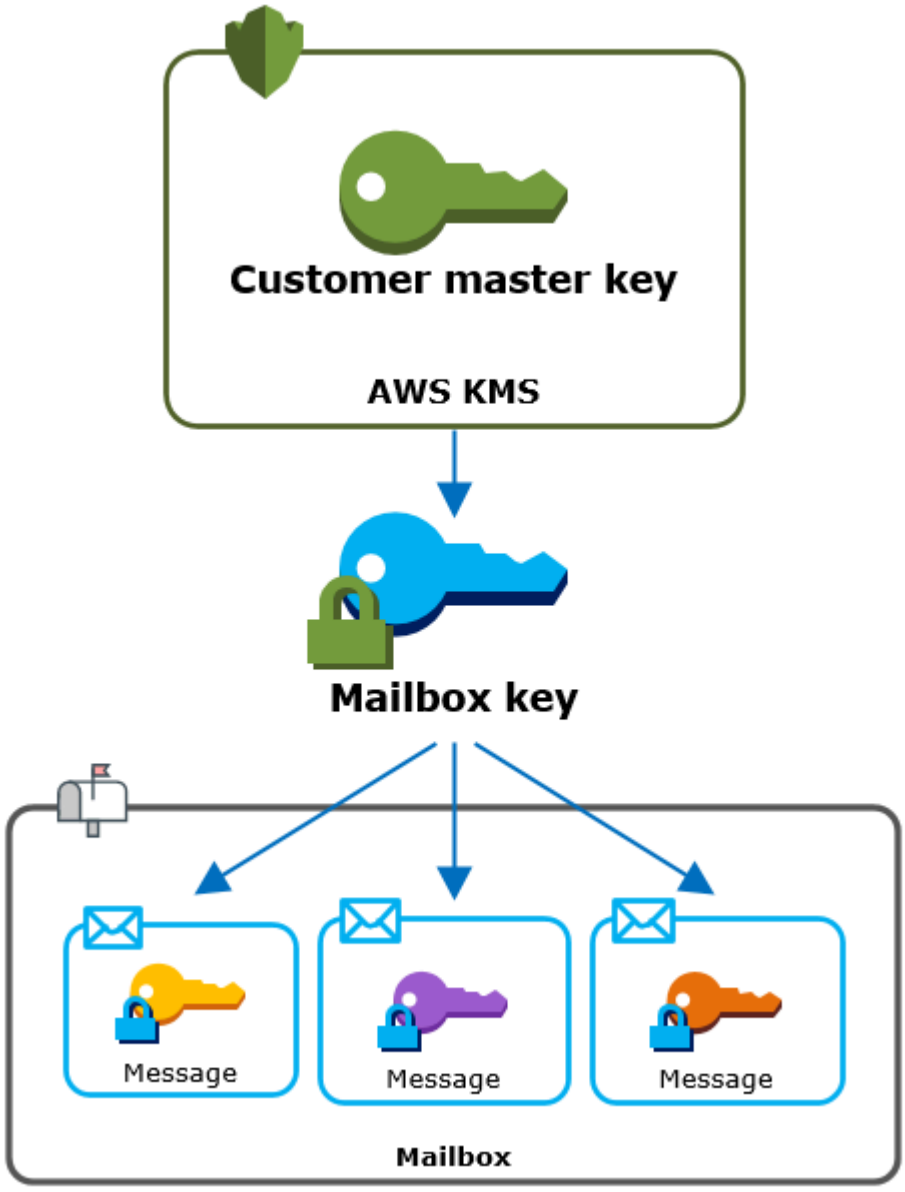
- Amazon WorkMail encryption
- Authorizing use of the CMK
- Amazon WorkMail encryption context
- Monitoring Amazon WorkMail interaction with AWS KMS

Amazon WorkMail encryption

In Amazon WorkMail, each organization can contain multiple mailboxes, one for each user in the organization. All messages, including email and calendar items, are stored in the user's mailbox.

To protect the contents of the mailboxes in your Amazon WorkMail organizations, Amazon WorkMail encrypts all mailbox messages before they are written to disk. No customer-provided information is stored in plaintext.

Each message is encrypted under a unique data encryption key. The message key is protected by a mailbox key, which is a unique encryption key that is used only for that mailbox. The mailbox key is encrypted under an AWS KMS customer master key (CMK) for the organization that never leaves AWS KMS unencrypted. The following diagram shows the relationship of the encrypted messages, encrypted message keys, encrypted mailbox key, and the CMK for the organization in AWS KMS.



Setting a CMK for the organization

When you create an Amazon WorkMail organization, you have the option to select an AWS KMS customer master key (CMK) for the organization. This CMK protects all mailbox keys in that organization.

You can select the default AWS managed CMK for Amazon WorkMail, or you can select an existing customer managed CMK that you own and manage. For more information, see customer master keys (CMKs) in the *AWS Key Management Service Developer Guide*. You can select the same CMK or a different CMK for each of your organizations, but you cannot change the CMK once you select it.

Important

Amazon WorkMail supports only symmetric CMKs. You cannot use an asymmetric CMK. For help determining whether a CMK is symmetric or asymmetric, see Identifying symmetric and asymmetric CMKs in the *AWS Key Management Service Developer Guide*.

To find the CMK for your organization, use the AWS CloudTrail log entry that records calls to AWS KMS.

A unique encryption key for each mailbox

When you create a mailbox, Amazon WorkMail generates a unique 256-bit Advanced Encryption Standard (AES) symmetric encryption key for the mailbox, known as its *mailbox key*, outside of AWS KMS. Amazon WorkMail uses the mailbox key to protect the encryption keys for each message in the mailbox.

To protect the mailbox key, Amazon WorkMail calls AWS KMS to encrypt the mailbox key under the CMK for the organization. Then it stores the encrypted mailbox key in the mailbox metadata.

Note

Amazon WorkMail uses a symmetric mailbox encryption key to protect message keys. Previously, Amazon WorkMail protected each mailbox with an asymmetric key pair. It used the public key to encrypt each message key and the private key to decrypt it. The private mailbox key was protected by the CMK for the organization. Older mailboxes may use an asymmetric mailbox key pair. This change does not affect the security of the mailbox or its messages.

Encrypting each message

When a user adds a message to a mailbox, Amazon WorkMail generates a unique 256-bit AES symmetric encryption key for the message outside of AWS KMS. It uses this *message key* to encrypt the message. Amazon WorkMail encrypts the message key under the mailbox key and stores the encrypted message key with the message. Then, it encrypts the mailbox key under the CMK for the organization.

Creating a new mailbox

When Amazon WorkMail creates a mailbox, it uses the following process to prepare the mailbox to hold encrypted messages.

- Amazon WorkMail generates a unique 256-bit AES symmetric encryption key for the mailbox outside of AWS KMS.
- Amazon WorkMail calls the AWS KMS Encrypt operation. It passes in the mailbox key and the identifier of the customer master key (CMK) for the organization. AWS KMS returns a ciphertext of the mailbox key encrypted under the CMK.
- Amazon WorkMail stores the encrypted mailbox key with the mailbox metadata.

Encrypting a mailbox message

To encrypt a message, Amazon WorkMail uses the following process.

1. Amazon WorkMail generates a unique 256-bit AES symmetric key for the message. It uses the plaintext message key and the Advanced Encryption Standard (AES) algorithm to encrypt the message outside of AWS KMS.
2. To protect the message key under the mailbox key, Amazon WorkMail needs to decrypt the mailbox key, which is always stored in its encrypted form.

   Amazon WorkMail calls the AWS KMS Decrypt operation and passes in the encrypted mailbox key. AWS KMS uses the CMK for the organization to decrypt the mailbox key and it returns the plaintext mailbox key to Amazon WorkMail.
3. Amazon WorkMail uses the plaintext mailbox key and the Advanced Encryption Standard (AES) algorithm to encrypt the message key outside of AWS KMS.
4. Amazon WorkMail stores the encrypted message key in the metadata of the encrypted message so it is available to decrypt it.

Decrypting a mailbox message

To decrypt a message, Amazon WorkMail uses the following process.

1. Amazon WorkMail calls the AWS KMS Decrypt operation and passes in the encrypted mailbox key. AWS KMS uses the CMK for the organization to decrypt the mailbox key and it returns the plaintext mailbox key to Amazon WorkMail.

2. Amazon WorkMail uses the plaintext mailbox key and the Advanced Encryption Standard (AES) algorithm to decrypt the encrypted message key outside of AWS KMS.
3. Amazon WorkMail uses the plaintext message key to decrypt the encrypted message.

Caching mailbox keys

To improve performance and minimize calls to AWS KMS, Amazon WorkMail caches each plaintext mailbox key for each client locally for up to one minute. At the end of the caching period, the mailbox key is removed. If the mailbox key for that client is required during the caching period, Amazon WorkMail can get it from the cache instead of calling AWS KMS. The mailbox key is protected in the cache and is never written to disk in plaintext.

Authorizing use of the CMK

When Amazon WorkMail uses a customer master key (CMK) in cryptographic operations, it acts on behalf of the mailbox administrator.

To use the AWS KMS customer master key (CMK) for a secret on your behalf, the administrator must have the following permissions. You can specify these required permissions in an IAM policy or key policy.

- `kms:Encrypt`
- `kms:Decrypt`
- `kms:CreateGrant`

To allow the CMK to be used only for requests that originate in Amazon WorkMail, you can use the kms:ViaService condition key with the `workmail.<region>.amazonaws.com` value.

You can also use the keys or values in the encryption context as a condition for using the CMK for cryptographic operations. For example, you can use a string condition operator in an IAM or key policy document or use a grant constraint in a grant.

Key policy for the AWS managed CMK

The key policy for the AWS managed CMK for Amazon WorkMail gives users permission to use the CMK for specified operations only when Amazon WorkMail makes the request on the user's behalf. The key policy does not allow any user to use the CMK directly.

This key policy, like the policies of all AWS managed keys, is established by the service. You can't change the key policy, but you can view it at any time. For details, see Viewing a key policy in the *AWS Key Management Service Developer Guide*.

The policy statements in the key policy have the following effect:

- Allow users in the account and Region to use the CMK for cryptographic operations and to create grants, but only when the request comes from Amazon WorkMail on their behalf. The `kms:ViaService` condition key enforces this restriction.
- Allows the AWS account to create IAM policies that allow users to view CMK properties and revoke grants.

The following is a key policy for an example AWS managed CMK for Amazon WorkMail.

JSON

```
{
  "Version" : "2012-10-17",
  "Id" : "auto-workmail-1",
  "Statement" : [ {
    "Sid" : "Allow access through WorkMail for all principals in the account that
 are authorized to use WorkMail",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [ "kms:Decrypt", "kms:CreateGrant", "kms:ReEncrypt*",
 "kms:DescribeKey", "kms:Encrypt" ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "workmail.us-east-1.amazonaws.com",
        "kms:CallerAccount" : "111122223333"
      }
    }
  }, {
    "Sid" : "Allow direct access to key metadata to the account",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
```

```
    "Action" : [ "kms:Describe*", "kms:List*", "kms:Get*", "kms:RevokeGrant" ],
    "Resource" : "*"
  } ]
}
```

Using grants to authorize Amazon WorkMail

In addition to key policies, Amazon WorkMail uses grants to add permissions to the CMK for each organization. To view the grants on the CMK in your account, use the ListGrants operation.

Amazon WorkMail uses grants to add the following permissions to the CMK for the organization.

- Add the `kms:Encrypt` permission to allow Amazon WorkMail to encrypt the mailbox key.
- Add the `kms:Decrypt` permission to allow Amazon WorkMail to use the CMK to decrypt the mailbox key. Amazon WorkMail requires this permission in a grant because the request to read mailbox messages uses the security context of the user who is reading the message. The request does not use the credentials of the AWS account. Amazon WorkMail creates this grant when you select a CMK for the organization.

To create the grants, Amazon WorkMail calls CreateGrant on behalf of the user who created the organization. Permission to create the grant comes from the key policy. This policy allows account users to call `CreateGrant` on the CMK for the organization when Amazon WorkMail makes the request on an authorized user's behalf.

The key policy also allows the account root to revoke the grant on the AWS managed key. However, if you revoke the grant, Amazon WorkMail can't decrypt the encrypted data in your mailboxes.

Amazon WorkMail encryption context

An encryption context is a set of key-value pairs that contain arbitrary nonsecret data. When you include an encryption context in a request to encrypt data, AWS KMS cryptographically binds the encryption context to the encrypted data. To decrypt the data, you must pass in the same encryption context. For more information, see Encryption context in the *AWS Key Management Service Developer Guide*.

Amazon WorkMail uses the same encryption context format in all AWS KMS cryptographic operations. You can use the encryption context to identify a cryptographic operation in audit records and logs, such as AWS CloudTrail, and as a condition for authorization in policies and grants.

In its Encrypt and Decrypt requests to AWS KMS, Amazon WorkMail uses an encryption context where the key is `aws:workmail:arn` and the value is the Amazon Resource Name (ARN) of the organization.

```
"aws:workmail:arn":"arn:aws:workmail:region:account ID:organization/organization-ID"
```

For example, the following encryption context includes an example organization ARN in the Europe (Ireland) (`eu-west-1`) Region.

```
"aws:workmail:arn":"arn:aws:workmail:eu-west-1:111122223333:organization/m-
a123b4c5de678fg9h0ij1k2lm234no56"
```

Monitoring Amazon WorkMail interaction with AWS KMS

You can use AWS CloudTrail and Amazon CloudWatch Logs to track the requests that Amazon WorkMail sends to AWS KMS on your behalf.

Encrypt

When you create a mailbox, Amazon WorkMail generates a mailbox key and calls AWS KMS to encrypt the mailbox key. Amazon WorkMail sends an Encrypt request to AWS KMS with the plaintext mailbox key and an identifier for the CMK of the Amazon WorkMail organization.

The event that records the `Encrypt` operation is similar to the following example event. The user is the Amazon WorkMail service. The parameters include the CMK ID (`keyId`) and the encryption context for the Amazon WorkMail organization. Amazon WorkMail also passes in the mailbox key, but that is not recorded in the CloudTrail log.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "workmail.eu-west-1.amazonaws.com"
    },
    "eventTime": "2019-02-19T10:01:09Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Encrypt",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "workmail.eu-west-1.amazonaws.com",
    "userAgent": "workmail.eu-west-1.amazonaws.com",
```

```
    "requestParameters": {
        "encryptionContext": {
            "aws:workmail:arn": "arn:aws:workmail:eu-west-1:111122223333:organization/
m-a123b4c5de678fg9h0ij1k2lm234no56"
        },
        "keyId": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    },
    "responseElements": null,
    "requestID": "76e96b96-7e24-4faf-a2d6-08ded2eaf63c",
    "eventID": "d5a59c18-128a-4082-aa5b-729f7734626a",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "accountId": "111122223333",
            "type": "AWS::KMS::Key"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "sharedEventID": "d08e60f1-097e-4a00-b7e9-10bc3872d50c"
}
```


Decrypt

When you add, view, or delete a mailbox message, Amazon WorkMail asks AWS KMS to decrypt the mailbox key. Amazon WorkMail sends a Decrypt request to AWS KMS with the encrypted mailbox key and an identifier for the CMK of the Amazon WorkMail organization.

The event that records the `Decrypt` operation is similar to the following example event. The user is the Amazon WorkMail service. The parameters include the encrypted mailbox key (as a ciphertext blob), which is not recorded in the log, and the encryption context for the Amazon WorkMail organization. AWS KMS derives the ID of the CMK from the ciphertext.


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "workmail.eu-west-1.amazonaws.com"
    },
    "eventTime": "2019-02-20T11:51:10Z",
```

```
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "workmail.eu-west-1.amazonaws.com",
    "userAgent": "workmail.eu-west-1.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:workmail:arn": "arn:aws:workmail:eu-west-1:111122223333:organization/
m-a123b4c5de678fg9h0ij1k2lm234no56"
        }
    },
    "responseElements": null,
    "requestID": "4a32dda1-34d9-4100-9718-674b8e0782c9",
    "eventID": "ea9fd966-98e9-4b7b-b377-6e5a397a71de",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "accountId": "111122223333",
            "type": "AWS::KMS::Key"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "sharedEventID": "241e1e5b-ff64-427a-a5b3-7949164d0214"
}
```


Identity and access management for Amazon WorkMail

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be *authenticated* (signed in) and *authorized* (have permissions) to use Amazon WorkMail resources. IAM is an AWS service that you can use with no additional charge.

Topics

- Audience
- Authenticating With identities
- Managing access using policies
- How Amazon WorkMail works with IAM

- Amazon WorkMail identity-based policy examples
- Troubleshooting Amazon WorkMail identity and access

Audience

How you use AWS Identity and Access Management (IAM) differs, depending on the work that you do in Amazon WorkMail.

Service user – If you use the Amazon WorkMail service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more Amazon WorkMail features to do your work, you might need additional permissions. Understanding how access is managed can help you request the right permissions from your administrator. If you cannot access a feature in Amazon WorkMail, see Troubleshooting Amazon WorkMail identity and access.

Service administrator – If you're in charge of Amazon WorkMail resources at your company, you probably have full access to Amazon WorkMail. It's your job to determine which Amazon WorkMail features and resources your service users should access. You must then submit requests to your IAM administrator to change the permissions of your service users. Review the information on this page to understand the basic concepts of IAM. To learn more about how your company can use IAM with Amazon WorkMail, see How Amazon WorkMail works with IAM.

IAM administrator – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to Amazon WorkMail. To view example Amazon WorkMail identity-based policies that you can use in IAM, see Amazon WorkMail identity-based policy examples.

Authenticating With identities

Authentication is how you sign in to AWS using your identity credentials. You must be *authenticated* (signed in to AWS) as the AWS account root user, as an IAM user, or by assuming an IAM role.

You can sign in to AWS as a federated identity by using credentials provided through an identity source. AWS IAM Identity Center (IAM Identity Center) users, your company's single sign-on authentication, and your Google or Facebook credentials are examples of federated identities. When you sign in as a federated identity, your administrator previously set up identity federation using IAM roles. When you access AWS by using federation, you are indirectly assuming a role.

Depending on the type of user you are, you can sign in to the AWS Management Console or the AWS access portal. For more information about signing in to AWS, see How to sign in to your AWS account in the *AWS Sign-In User Guide*.

If you access AWS programmatically, AWS provides a software development kit (SDK) and a command line interface (CLI) to cryptographically sign your requests by using your credentials. If you don't use AWS tools, you must sign requests yourself. For more information about using the recommended method to sign requests yourself, see AWS Signature Version 4 for API requests in the *IAM User Guide*.

Regardless of the authentication method that you use, you might be required to provide additional security information. For example, AWS recommends that you use multi-factor authentication (MFA) to increase the security of your account. To learn more, see Multi-factor authentication in the *AWS IAM Identity Center User Guide* and AWS Multi-factor authentication in IAM in the *IAM User Guide*.

AWS account root user

When you create an AWS account, you begin with one sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account *root user* and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you don't use the root user for your everyday tasks. Safeguard your root user credentials and use them to perform the tasks that only the root user can perform. For the complete list of tasks that require you to sign in as the root user, see Tasks that require root user credentials in the *IAM User Guide*.

IAM users and groups

An *IAM user* is an identity within your AWS account that has specific permissions for a single person or application. Where possible, we recommend relying on temporary credentials instead of creating IAM users who have long-term credentials such as passwords and access keys. However, if you have specific use cases that require long-term credentials with IAM users, we recommend that you rotate access keys. For more information, see Rotate access keys regularly for use cases that require long-term credentials in the *IAM User Guide*.

An *IAM group* is an identity that specifies a collection of IAM users. You can't sign in as a group. You can use groups to specify permissions for multiple users at a time. Groups make permissions easier to manage for large sets of users. For example, you could have a group named *IAMAdmins* and give that group permissions to administer IAM resources.

Users are different from roles. A user is uniquely associated with one person or application, but a role is intended to be assumable by anyone who needs it. Users have permanent long-term credentials, but roles provide temporary credentials. To learn more, see Use cases for IAM users in the *IAM User Guide*.

IAM roles

An *IAM role* is an identity within your AWS account that has specific permissions. It is similar to an IAM user, but is not associated with a specific person. To temporarily assume an IAM role in the AWS Management Console, you can switch from a user to an IAM role (console). You can assume a role by calling an AWS CLI or AWS API operation or by using a custom URL. For more information about methods for using roles, see Methods to assume a role in the *IAM User Guide*.

IAM roles with temporary credentials are useful in the following situations:

- **Federated user access** – To assign permissions to a federated identity, you create a role and define permissions for the role. When a federated identity authenticates, the identity is associated with the role and is granted the permissions that are defined by the role. For information about roles for federation, see Create a role for a third-party identity provider (federation) in the *IAM User Guide*. If you use IAM Identity Center, you configure a permission set. To control what your identities can access after they authenticate, IAM Identity Center correlates the permission set to a role in IAM. For information about permissions sets, see Permission sets in the *AWS IAM Identity Center User Guide*.
- **Temporary IAM user permissions** – An IAM user or role can assume an IAM role to temporarily take on different permissions for a specific task.
- **Cross-account access** – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-account access. However, with some AWS services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see Cross account resource access in IAM in the *IAM User Guide*.
- **Cross-service access** – Some AWS services use features in other AWS services. For example, when you make a call in a service, it's common for that service to run applications in Amazon EC2 or store objects in Amazon S3. A service might do this using the calling principal's permissions, using a service role, or using a service-linked role.
  - **Forward access sessions (FAS)** – When you use an IAM user or role to perform actions in AWS, you are considered a principal. When you use some services, you might perform an action that then initiates another action in a different service. FAS uses the permissions of the

principal calling an AWS service, combined with the requesting AWS service to make requests to downstream services. FAS requests are only made when a service receives a request that requires interactions with other AWS services or resources to complete. In this case, you must have permissions to perform both actions. For policy details when making FAS requests, see Forward access sessions.

- **Service role** – A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Create a role to delegate permissions to an AWS service in the *IAM User Guide*.
- **Service-linked role** – A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your AWS account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

- **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the EC2 instance. To assign an AWS role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Use an IAM role to grant permissions to applications running on Amazon EC2 instances in the *IAM User Guide*.

Managing access using policies

You control access in AWS by creating policies and attaching them to AWS identities or resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. AWS evaluates these policies when a principal (user, root user, or role session) makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents. For more information about the structure and contents of JSON policy documents, see Overview of JSON policies in the *IAM User Guide*.

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

By default, users and roles have no permissions. To grant users permission to perform actions on the resources that they need, an IAM administrator can create IAM policies. The administrator can then add the IAM policies to roles, and users can assume the roles.

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, suppose that you have a policy that allows the `iam:GetRole` action. A user with that policy can get role information from the AWS Management Console, the AWS CLI, or the AWS API.

Identity-based policies

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, group of users, or role. These policies control what actions users and roles can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Define custom IAM permissions with customer managed policies in the *IAM User Guide*.

Identity-based policies can be further categorized as *inline policies* or *managed policies*. Inline policies are embedded directly into a single user, group, or role. Managed policies are standalone policies that you can attach to multiple users, groups, and roles in your AWS account. Managed policies include AWS managed policies and customer managed policies. To learn how to choose between a managed policy or an inline policy, see Choose between managed policies and inline policies in the *IAM User Guide*.

Resource-based policies

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM *role trust policies* and Amazon S3 *bucket policies*. In services that support resource-based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services.

Resource-based policies are inline policies that are located in that service. You can't use AWS managed policies from IAM in a resource-based policy.

Access Control Lists (ACLs)

Access control lists (ACLs) control which principals (account members, users, or roles) have permissions to access a resource. ACLs are similar to resource-based policies, although they do not use the JSON policy document format.

Amazon S3, AWS WAF, and Amazon VPC are examples of services that support ACLs. To learn more about ACLs, see Access control list (ACL) overview in the *Amazon Simple Storage Service Developer Guide*.

Other policy types

AWS supports additional, less-common policy types. These policy types can set the maximum permissions granted to you by the more common policy types.

- **Permissions boundaries** – A permissions boundary is an advanced feature in which you set the maximum permissions that an identity-based policy can grant to an IAM entity (IAM user or role). You can set a permissions boundary for an entity. The resulting permissions are the intersection of an entity's identity-based policies and its permissions boundaries. Resource-based policies that specify the user or role in the `Principal` field are not limited by the permissions boundary. An explicit deny in any of these policies overrides the allow. For more information about permissions boundaries, see Permissions boundaries for IAM entities in the *IAM User Guide*.
- **Service control policies (SCPs)** – SCPs are JSON policies that specify the maximum permissions for an organization or organizational unit (OU) in AWS Organizations. AWS Organizations is a service for grouping and centrally managing multiple AWS accounts that your business owns. If you enable all features in an organization, then you can apply service control policies (SCPs) to any or all of your accounts. The SCP limits permissions for entities in member accounts, including each AWS account root user. For more information about Organizations and SCPs, see Service control policies in the *AWS Organizations User Guide*.
- **Resource control policies (RCPs)** – RCPs are JSON policies that you can use to set the maximum available permissions for resources in your accounts without updating the IAM policies attached to each resource that you own. The RCP limits permissions for resources in member accounts and can impact the effective permissions for identities, including the AWS account root user, regardless of whether they belong to your organization. For more information about Organizations and RCPs, including a list of AWS services that support RCPs, see Resource control policies (RCPs) in the *AWS Organizations User Guide*.
- **Session policies** – Session policies are advanced policies that you pass as a parameter when you programmatically create a temporary session for a role or federated user. The resulting session's permissions are the intersection of the user or role's identity-based policies and the session policies. Permissions can also come from a resource-based policy. An explicit deny in any of these policies overrides the allow. For more information, see Session policies in the *IAM User Guide*.

Multiple policy types

When multiple types of policies apply to a request, the resulting permissions are more complicated to understand. To learn how AWS determines whether to allow a request when multiple policy types are involved, see Policy evaluation logic in the *IAM User Guide*.

How Amazon WorkMail works with IAM

Before you use IAM to manage access to Amazon WorkMail, you should understand what IAM features are available to use with Amazon WorkMail. To get a high-level view of how Amazon WorkMail and other AWS services work with IAM, see AWS services that work with IAM in the *IAM User Guide*.

Topics

- Amazon WorkMail identity-based policies
- Amazon WorkMail resource-based policies
- Authorization based on Amazon WorkMail tags
- Amazon WorkMail IAM roles

Amazon WorkMail identity-based policies

With IAM identity-based policies, you can specify allowed or denied actions and resources as well as the conditions under which actions are allowed or denied. Amazon WorkMail supports specific actions, resources, and condition keys. To learn about all of the elements that you use in a JSON policy, see IAM JSON Policy Elements Reference in the *IAM User Guide*.

Actions

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Action` element of a JSON policy describes the actions that you can use to allow or deny access in a policy. Policy actions usually have the same name as the associated AWS API operation. There are some exceptions, such as *permission-only actions* that don't have a matching API operation. There are also some operations that require multiple actions in a policy. These additional actions are called *dependent actions*.

Include actions in a policy to grant permissions to perform the associated operation.

Policy actions in Amazon WorkMail use the following prefix before the action: `workmail:`. For example, to grant someone permission to retrieve a list of users with the Amazon WorkMail `ListUsers` API operation, you include the `workmail:ListUsers` action in their policy. Policy statements must include either an `Action` or `NotAction` element. Amazon WorkMail defines its own set of actions that describe tasks that you can perform with this service.

To specify multiple actions in a single statement, separate them with commas as follows:

```
"Action": [
      "workmail:ListUsers",
      "workmail:DeleteUser"
```

You can specify multiple actions using wildcards (*). For example, to specify all actions that begin with the word `List`, include the following action:

```
"Action": "workmail:List*"
```

To see a list of Amazon WorkMail actions, see Actions defined by Amazon WorkMail in the *IAM User Guide*.

Resources

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Resource` JSON policy element specifies the object or objects to which the action applies. Statements must include either a `Resource` or a `NotResource` element. As a best practice, specify a resource using its Amazon Resource Name (ARN). You can do this for actions that support a specific resource type, known as *resource-level permissions*.

For actions that don't support resource-level permissions, such as listing operations, use a wildcard (*) to indicate that the statement applies to all resources.

```
"Resource": "*"
```

Amazon WorkMail supports resource-level permissions for Amazon WorkMail organizations.

The Amazon WorkMail organization resource has the following ARN:

```
arn:aws:workmail:${Region}:${Account}:organization/${OrganizationId}
```

For more information about the format of ARNs, see Amazon Resource Names (ARNs) and AWS service namespaces.

For example, to specify the `m-n1pq2345678r901st2u3vx45x6789yza` organization in your statement, use the following ARN.

```
"Resource": "arn:aws:workmail:us-east-1:111122223333:organization/m-
n1pq2345678r901st2u3vx45x6789yza"
```

To specify all organizations that belong to a specific account, use the wildcard (*):

```
"Resource": "arn:aws:workmail:us-east-1:111122223333:organization/*"
```

Some Amazon WorkMail actions, such as those for creating resources, can't be performed on a specific resource. In those cases, you must use the wildcard (*).

```
"Resource": "*"
```

To see a list of Amazon WorkMail resource types and their ARNs, see Resources defined by Amazon WorkMail in the *IAM User Guide*. To learn with which actions you can specify for the ARN of each resource, see Actions, resources, and condition keys for Amazon WorkMail.

Condition keys

Amazon WorkMail supports the following global condition keys.

- `aws:CurrentTime`
- `aws:EpochTime`
- `aws:MultiFactorAuthAge`
- `aws:MultiFactorAuthPresent`
- `aws:PrincipalOrgID`
- `aws:PrincipalArn`
- `aws:RequestedRegion`
- `aws:SecureTransport`
- `aws:UserAgent`

The following example policy grants access to the Amazon WorkMail console only from MFA authenticated IAM principals in the `eu-west-1` AWS Region.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ses:Describe*",
                "ses:Get*",
                "workmail:Describe*",
                "workmail:Get*",
                "workmail:List*",
                "workmail:Search*",
                "lambda:ListFunctions",
                "iam:ListRoles",
                "logs:DescribeLogGroups",
                "cloudwatch:GetMetricData"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestedRegion": [
                        "eu-west-1"
                    ]
                },
                "Bool": {
                    "aws:MultiFactorAuthPresent": true
                }
            }
        }
    ]
}
```

To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

`workmail:ImpersonationRoleId` is the only service-specific condition key supported by Amazon WorkMail.

The following example policy scopes-down `AssumeImpersonationRole` action to a particular WorkMail organization and impersonation role.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "workmail:AssumeImpersonationRole"
      ],
      "Resource": "arn:aws:workmail:us-east-1:111122223333:organization/m-
n1pq2345678r901st2u3vx45x6789yza",
      "Condition": {
        "StringEquals": {
          "workmail:ImpersonationRoleId":"12345678-1234-1234-1234-123456789012"
        }
      }
    }
  ]
}
```

Examples

To view examples of Amazon WorkMail identity-based policies, see Amazon WorkMail identity-based policy examples.

Amazon WorkMail resource-based policies

Amazon WorkMail does not support resource-based policies.

Authorization based on Amazon WorkMail tags

You can attach tags to Amazon WorkMail resources or pass tags in a request to Amazon WorkMail. To control access based on tags, you provide tag information in the condition element of a policy using the `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, or `aws:TagKeys` condition keys. For more information about tagging Amazon WorkMail resources, see Tagging an organization.

Amazon WorkMail IAM roles

An IAM role is an entity within your AWS account that has specific permissions.

Using temporary credentials with Amazon WorkMail

You can use temporary credentials to sign in with federation, assume an IAM role, or to assume a cross-account role. You obtain temporary security credentials by calling AWS STS API operations such as AssumeRole or GetFederationToken.

Amazon WorkMail supports using temporary credentials.

Service-linked roles

Service-linked roles allow AWS services to access resources in other services to complete an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view but not edit the permissions for service-linked roles.

Amazon WorkMail supports service-linked roles. For details about creating or managing Amazon WorkMail service-linked roles, see Using service-linked roles for Amazon WorkMail.

Service roles

This feature allows a service to assume a service role on your behalf. This role allows the service to access resources in other services to complete an action on your behalf. Service roles appear in your IAM account and are owned by the account. This means that an IAM administrator can change the permissions for this role. However, doing so might break the functionality of the service.

Amazon WorkMail supports service roles.

Amazon WorkMail identity-based policy examples

By default, IAM users and roles don't have permission to create or modify Amazon WorkMail resources. They also can't perform tasks using the AWS Management Console, AWS CLI, or AWS API. An IAM administrator must create IAM policies that grant users and roles permission to perform specific API operations on the specified resources they need. The administrator must then attach those policies to the IAM users or groups that require those permissions.

To learn how to create an IAM identity-based policy using these example JSON policy documents, see Creating policies on the JSON tab in the *IAM User Guide*.

Topics

- Policy best practices
- Using the Amazon WorkMail console
- Allow users to view their own permissions
- Allow users read-only access to Amazon WorkMail resources

Policy best practices

Identity-based policies determine whether someone can create, access, or delete Amazon WorkMail resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get started with AWS managed policies and move toward least-privilege permissions** – To get started granting permissions to your users and workloads, use the *AWS managed policies* that grant permissions for many common use cases. They are available in your AWS account. We recommend that you reduce permissions further by defining AWS customer managed policies that are specific to your use cases. For more information, see AWS managed policies or AWS managed policies for job functions in the *IAM User Guide*.
- **Apply least-privilege permissions** – When you set permissions with IAM policies, grant only the permissions required to perform a task. You do this by defining the actions that can be taken on specific resources under specific conditions, also known as *least-privilege permissions*. For more information about using IAM to apply permissions, see Policies and permissions in IAM in the *IAM User Guide*.
- **Use conditions in IAM policies to further restrict access** – You can add a condition to your policies to limit access to actions and resources. For example, you can write a policy condition to specify that all requests must be sent using SSL. You can also use conditions to grant access to service actions if they are used through a specific AWS service, such as AWS CloudFormation. For more information, see IAM JSON policy elements: Condition in the *IAM User Guide*.
- **Use IAM Access Analyzer to validate your IAM policies to ensure secure and functional permissions** – IAM Access Analyzer validates new and existing policies so that the policies adhere to the IAM policy language (JSON) and IAM best practices. IAM Access Analyzer provides more than 100 policy checks and actionable recommendations to help you author secure and functional policies. For more information, see Validate policies with IAM Access Analyzer in the *IAM User Guide*.
- **Require multi-factor authentication (MFA)** – If you have a scenario that requires IAM users or a root user in your AWS account, turn on MFA for additional security. To require MFA when API

operations are called, add MFA conditions to your policies. For more information, see Secure API access with MFA in the *IAM User Guide*.

For more information about best practices in IAM, see Security best practices in IAM in the *IAM User Guide*.

Using the Amazon WorkMail console

To access the Amazon WorkMail console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Amazon WorkMail resources in your AWS account. If you create an identity-based policy that is more restrictive than the minimum required permissions, the console won't function as intended for entities (IAM users or roles) with that policy.

To ensure that those entities can still use the Amazon WorkMail console, also attach the following AWS managed policy, **AmazonWorkMailFullAccess**, to the entities. For more information, see Adding permissions to a user in the *IAM User Guide*.

The **AmazonWorkMailFullAccess** policy grants an IAM user full access to Amazon WorkMail resources. This policy gives the user access to all Amazon WorkMail, AWS Key Management Service, Amazon Simple Email Service, and AWS Directory Service operations. This also includes several Amazon EC2 operations that Amazon WorkMail needs to perform on your behalf. The `logs` and `cloudwatch` permissions are required for email event logging, and viewing metrics in the Amazon WorkMail console. Audit logging uses CloudWatch Logs, Amazon S3, and Amazon Data FireHose to store `logs`. For more information, see Logging and monitoring in Amazon WorkMail.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "WorkMailAdministration",
      "Effect": "Allow",
      "Action": [
        "ds:AuthorizeApplication",
        "ds:CheckAlias",
        "ds:CreateAlias",
        "ds:CreateDirectory",
        "ds:CreateIdentityPoolDirectory",
```

```
            "ds:DeleteDirectory",
            "ds:DescribeDirectories",
            "ds:GetDirectoryLimits",
            "ds:ListAuthorizedApplications",
            "ds:UnauthorizeApplication",
            "ec2:AuthorizeSecurityGroupEgress",
            "ec2:AuthorizeSecurityGroupIngress",
            "ec2:CreateNetworkInterface",
            "ec2:CreateSecurityGroup",
            "ec2:CreateSubnet",
            "ec2:CreateTags",
            "ec2:CreateVpc",
            "ec2:DeleteSecurityGroup",
            "ec2:DeleteSubnet",
            "ec2:DeleteVpc",
            "ec2:DescribeAvailabilityZones",
            "ec2:DescribeRouteTables",
            "ec2:DescribeSubnets",
            "ec2:DescribeVpcs",
            "ec2:RevokeSecurityGroupEgress",
            "ec2:RevokeSecurityGroupIngress",
            "kms:DescribeKey",
            "kms:ListAliases",
            "lambda:ListFunctions",
            "route53:ChangeResourceRecordSets",
            "route53:ListHostedZones",
            "route53:ListResourceRecordSets",
            "route53:GetHostedZone",
            "route53domains:CheckDomainAvailability",
            "route53domains:ListDomains",
            "ses:*",
            "workmail:*",
            "iam:ListRoles",
            "logs:DescribeLogGroups",
            "logs:CreateLogGroup",
            "logs:PutRetentionPolicy",
            "logs:DeleteDeliveryDestination",
            "logs:DeleteDeliveryDestinationPolicy",
            "logs:DescribeDeliveryDestinations",
            "logs:GetDeliveryDestination",
            "logs:GetDeliveryDestinationPolicy",
            "logs:PutDeliveryDestination",
            "logs:PutDeliveryDestinationPolicy",
            "logs:CreateDelivery",
```

```
            "logs:DeleteDelivery",
            "logs:DescribeDeliveries",
            "logs:GetDelivery",
            "logs:DeleteDeliverySource",
            "logs:DescribeDeliverySources",
            "logs:GetDeliverySource",
            "logs:PutDeliverySource",
            "logs:DescribeResourcePolicies",
            "cloudwatch:GetMetricData",
            "firehose:DescribeDeliveryStream",
            "firehose:ListDeliveryStreams",
            "s3:ListAllMyBuckets"
        ],
        "Resource": "*"
    },
    {
        "Sid": "AuditLogDeliveryThroughCWLogs",
        "Effect": "Allow",
        "Action": [
            "firehose:TagDeliveryStream",
            "logs:PutResourcePolicy",
            "s3:GetBucketPolicy",
            "s3:PutBucketPolicy"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "aws:CalledViaLast": "logs.amazonaws.com"
            }
        }
    },
    {
        "Sid": "InboundOutboundEmailEventsLink",
        "Effect": "Allow",
        "Action": "iam:CreateServiceLinkedRole",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "iam:AWSServiceName": "events.workmail.amazonaws.com"
            }
        }
    },
    {
        "Sid": "AuditLoggingLink",
```

```
        "Effect": "Allow",
        "Action": "iam:CreateServiceLinkedRole",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "iam:AWSServiceName": "delivery.logs.amazonaws.com"
            }
        }
    },
    {
        "Sid": "InboundOutboundEmailEventsUnlink",
        "Effect": "Allow",
        "Action": [
            "iam:DeleteServiceLinkedRole",
            "iam:GetServiceLinkedRoleDeletionStatus"
        ],
        "Resource": "arn:aws:iam::*:role/aws-service-role/
events.workmail.amazonaws.com/AWSServiceRoleForAmazonWorkMailEvents*"
    },
    {
        "Sid": "InboundOutboundEmailEventsAuth",
        "Effect": "Allow",
        "Action": "iam:PassRole",
        "Resource": "arn:aws:iam::*:role/*workmail*",
        "Condition": {
            "StringLike": {
                "iam:PassedToService": "events.workmail.amazonaws.com"
            }
        }
    }
  ]
}
```

You don't need to allow minimum console permissions for users that are making calls only to the AWS CLI or the AWS API. Instead, allow access to only the actions that match the API operation that you're trying to perform.

Allow users to view their own permissions

This example shows how you might create a policy that allows IAM users to view the inline and managed policies that are attached to their user identity. This policy includes permissions to complete this action on the console or programmatically using the AWS CLI or AWS API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Allow users read-only access to Amazon WorkMail resources

The following policy statement grants an IAM user read-only access to Amazon WorkMail resources. This policy gives the same level of access as the AWS managed policy

AmazonWorkMailReadOnlyAccess. Either policy gives the user access to all of the Amazon WorkMail `Describe` operations. Access to the AWS Directory Service `DescribeDirectories` operation is needed to obtain information about your AWS Directory Service directories. Access to the Amazon SES service is needed to obtain information about the configured domains. Access to AWS Key Management Service is needed to obtain information about the used encryption keys. The `logs` and `cloudwatch` permissions are required for email event logging and viewing metrics in the Amazon WorkMail console. Audit logging uses CloudWatch Logs, Amazon S3, and Amazon Data FireHose to store `logs`. For more information, see Logging and monitoring in Amazon WorkMail.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "WorkMailReadOnly",
      "Effect": "Allow",
      "Action": [
        "ses:Describe*",
        "ses:Get*",
        "workmail:Describe*",
        "workmail:Get*",
        "workmail:List*",
        "workmail:Search*",
        "lambda:ListFunctions",
        "iam:ListRoles",
        "logs:DescribeLogGroups",
        "logs:DescribeDeliveryDestinations",
        "logs:GetDeliveryDestination",
        "logs:GetDeliveryDestinationPolicy",
        "logs:DescribeDeliveries",
        "logs:DescribeDeliverySources",
        "logs:GetDelivery",
        "logs:GetDeliverySource",
        "cloudwatch:GetMetricData"
      ],
      "Resource": "*"
    }
  ]
}
```

Troubleshooting Amazon WorkMail identity and access

Use the following information to help you diagnose and fix common issues that you might encounter when working with Amazon WorkMail and IAM.

Topics

- I am not authorized to perform an action in Amazon WorkMail
- I am not authorized to perform iam:PassRole
- I want to allow people outside of my AWS account to access my Amazon WorkMail resources

I am not authorized to perform an action in Amazon WorkMail

If the AWS Management Console tells you that you're not authorized to perform an action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password.

The following example error occurs when the `mateojackson` IAM user tries to use the console to view details about a group but does not have `workmail:DescribeGroup` permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 workmail:DescribeGroup on resource: group
```

In this case, Mateo asks his administrator to update his policies to allow him to access the `group` resource using the `workmail:DescribeGroup` action.

I am not authorized to perform iam:PassRole

If you receive an error that you're not authorized to perform the `iam:PassRole` action, your policies must be updated to allow you to pass a role to Amazon WorkMail.

Some AWS services allow you to pass an existing role to that service instead of creating a new service role or service-linked role. To do this, you must have permissions to pass the role to the service.

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in Amazon WorkMail. However, the action requires the service to have permissions that are granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In this case, Mary's policies must be updated to allow her to perform the `iam:PassRole` action.

If you need help, contact your AWS administrator. Your administrator is the person who provided you with your sign-in credentials.

I want to allow people outside of my AWS account to access my Amazon WorkMail resources

You can create a role that users in other accounts or people outside of your organization can use to access your resources. You can specify who is trusted to assume the role. For services that support resource-based policies or access control lists (ACLs), you can use those policies to grant people access to your resources.

To learn more, consult the following:

- To learn whether Amazon WorkMail supports these features, see How Amazon WorkMail works with IAM.
- To learn how to provide access to your resources across AWS accounts that you own, see Providing access to an IAM user in another AWS account that you own in the *IAM User Guide*.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing access to AWS accounts owned by third parties in the *IAM User Guide*.
- To learn how to provide access through identity federation, see Providing access to externally authenticated users (identity federation) in the *IAM User Guide*.
- To learn the difference between using roles and resource-based policies for cross-account access, see Cross account resource access in IAM in the *IAM User Guide*.

AWS managed policies for Amazon WorkMail

To add permissions to users, groups, and roles, it is easier to use AWS managed policies than to write policies yourself. It takes time and expertise to create IAM customer managed policies 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 AWS managed policies 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 AWS managed policies for job functions in the *IAM User Guide*.

AWS managed policy: AmazonWorkMailFullAccess

You can attach the `AmazonWorkMailFullAccess` policy to your IAM identities. This policy grants permissions that allow full access to Amazon WorkMail.

To view the permissions for this policy, see AmazonWorkMailFullAccess in the AWS Management Console.

AWS managed policy: AmazonWorkMailReadOnlyAccess

You can attach the `AmazonWorkMailReadOnlyAccess` policy to your IAM identities. This policy grants permissions that allow read-only access to Amazon WorkMail.

To view the permissions for this policy, see AmazonWorkMailReadOnlyAccess in the AWS Management Console.

AWS managed policy: AmazonWorkMailEventsServiceRolePolicy

This policy is attached to the service-linked role named **AmazonWorkMailEvents** to allow access to AWS services and resources used or managed by Amazon WorkMail events. For more information, see Using service-linked roles for Amazon WorkMail.

Amazon WorkMail updates to AWS managed policies

View details about updates to AWS managed policies for Amazon WorkMail since this service began tracking these changes.

Change	Description	Date
AWS managed policy updates - Update to an existing policy	The `AmazonWorkMailReadOnlyAccess` and `AmazonWorkMailFullAccess` permissions were updated for Amazon WorkMail to support *audit logging*. For more information on the updated permissions, see Amazon WorkMail identity-based policy examples and for information on audit logging, see Enabling audit logging.	February 14, 2024
Amazon WorkMail started tracking changes	Amazon WorkMail started tracking changes for its AWS managed policies.	March 1, 2021

Using service-linked roles for Amazon WorkMail

Amazon WorkMail uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that is linked directly to Amazon WorkMail. Service-linked roles are predefined by Amazon WorkMail and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up Amazon WorkMail easier because you don’t have to manually add the necessary permissions. Amazon WorkMail defines the permissions of its service-linked roles, and unless defined otherwise, only Amazon WorkMail can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy can't be attached to any other IAM entity.

You can delete a service-linked role only after first deleting the related resources. This protects your Amazon WorkMail resources because you can't inadvertently remove permission to access the resources.

For information about other services that support service-linked roles, see AWS services that work with IAM and look for the services that have **Yes** in the **Service-linked role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Service-linked role permissions for Amazon WorkMail

Amazon WorkMail uses the service-linked role named **AmazonWorkMailEvents** – Amazon WorkMail uses this service-linked role to enable access to AWS services and resources used or managed by Amazon WorkMail events, such as monitoring email events logged by CloudWatch. For more information about enabling email event logging for Amazon WorkMail, see Enabling email event logging.

The AmazonWorkMailEvents service-linked role trusts the following services to assume the role:

- `events.workmail.amazonaws.com`

The role permissions policy allows Amazon WorkMail to complete the following actions on the specified resources:

- Action: `logs:CreateLogGroup` on `all AWS resources`
- Action: `logs:CreateLogStream` on `all AWS resources`
- Action: `logs:PutLogEvents` on `all AWS resources`

You must configure permissions to allow an IAM entity (such as a user, group, or role) 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 Amazon WorkMail

You don't need to manually create a service-linked role. When you turn on Amazon WorkMail event logging and use the default settings in the Amazon WorkMail console, Amazon WorkMail creates the service-linked role for you.

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 turn on Amazon WorkMail event logging and use the default settings, Amazon WorkMail creates the service-linked role for you again.

Editing a service-linked role for Amazon WorkMail

Amazon WorkMail does not allow you to edit the AmazonWorkMailEvents service-linked role. After you create a service-linked role, you can't 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 Amazon WorkMail

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 the resources for your service-linked role before you can manually delete it.

> **Note**
>
> If the Amazon WorkMail 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 Amazon WorkMail resources used by AmazonWorkMailEvents

1. Turn off Amazon WorkMail event logging.
   a. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

      If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
   b. In the navigation pane, choose **Organizations**, then choose the name of your organization.
   c. In the navigation pane, choose **Organization settings**, then choose **Monitoring**.
   d. For **Log settings**, choose **Edit**.
   e. Move the **Enable mail events** slider to the off position.
   f. Choose **Save**.
2. Delete the Amazon CloudWatch log group.
   a. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.

b. Choose **Logs**.

c. For **Log Groups**, select the log group to delete.

d. For **Actions**, choose **Delete log group**.

e. Choose **Yes, Delete**.

To manually delete the service-linked role using IAM

Use the IAM console, the AWS CLI, or the AWS API to delete the AmazonWorkMailEvents service-linked role. For more information, see Deleting a service-linked role in the *IAM User Guide*.

Supported Regions for Amazon WorkMail service-linked roles

Amazon WorkMail supports using service-linked roles in all of the Regions where the service is available. For more information, see Amazon WorkMail Regions and Endpoints.

Logging and monitoring in Amazon WorkMail

Monitoring and auditing your email and logs is important for maintaining the health of your Amazon WorkMail organization. Amazon WorkMail supports two types of monitoring:

- **Event logging** – Monitoring the email sending activity for your organization helps protect your domain reputation. Monitoring can also help you track emails that are sent and received. For more information about how to enable email event logging, see Enabling email event logging.
- **Audit logging** – You can use audit logs to capture detailed information about your Amazon WorkMail organization usage such as monitor user's access to mailboxes, audit for suspicious activity, and debug access control and availability provider configurations. For more information, see Enabling audit logging.

AWS provides the following monitoring tools to watch Amazon WorkMail, report when something is wrong, and take automatic actions when appropriate:

- *Amazon CloudWatch* monitors your AWS resources and the applications that you run on AWS in real time. For example, when you enable email event logging for Amazon WorkMail, CloudWatch can track emails sent and received for your organization. For more information about monitoring Amazon WorkMail with CloudWatch, see Monitoring Amazon WorkMail with CloudWatch metrics. For more information about CloudWatch, see the Amazon CloudWatch User Guide.

- *Amazon CloudWatch Logs* enables you to monitor, store, and access your email events and audit logs for Amazon WorkMail when email and audit logging is enabled in the Amazon WorkMail console. CloudWatch Logs can monitor information in the log files, and you can archive your log data in highly durable storage. For more information about tracking Amazon WorkMail messages using CloudWatch Logs, see Enabling email event logging and Enabling audit logging. For more information about CloudWatch Logs, see the Amazon CloudWatch Logs User Guide.
- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account, and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see Logging Amazon WorkMail API calls with AWS CloudTrail.
- *Amazon S3* enables you to store and access your Amazon WorkMail events in a cost-effective way. Amazon S3 provides mechanisms for managing the event data lifecycle, enabling you to configure automatic deletion of old events, or configure automatic archival to Amazon S3 Glacier. Note, delivery Amazon S3 is only available for audit logging events. For more information about Amazon S3, see the Amazon S3 User Guide.
- *Amazon Data Firehose* enables you to stream your event data to other AWS services such as Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon OpenSearch Service, Amazon OpenSearch Serverless, Splunk, and any custom HTTP endpoint or HTTP endpoints owned by supported third-party service providers, including Datadog, Dynatrace, LogicMonitor, MongoDB, New Relic, Coralogix, and Elastic. Delivery to Firehose is only available for audit logging events. For more information about Firehose, see the Amazon Data Firehose developer guide.

Topics

- Monitoring Amazon WorkMail with CloudWatch metrics
- Monitoring Amazon WorkMail email event logs
- Monitoring Amazon WorkMail audit logs
- Using CloudWatch Insights with Amazon WorkMail
- Logging Amazon WorkMail API calls with AWS CloudTrail
- Enabling email event logging
- Enabling audit logging

Monitoring Amazon WorkMail with CloudWatch metrics

You can monitor Amazon WorkMail using CloudWatch, which collects raw data and processes it into readable, near real-time metrics. The no charge metrics are stored for 15 months so that you can access historical information to see how your web application or service is performing. You can also set alarms that watch for certain thresholds, and send notifications or take actions when those thresholds are met. For more information, see the Amazon CloudWatch User Guide.

CloudWatch metrics for Amazon WorkMail

Amazon WorkMail sends the following metrics and dimension information to CloudWatch.

The `AWS/WorkMail` namespace includes the following metrics.

Metric	Description
`OrganizationEmailReceived`	The number of emails received by your Amazon WorkMail organization. If one email is addressed to 10 recipients in your organization, the `OrganizationEmailReceived` count is one. Units: Count
`MailboxEmailDelivered`	The number of emails delivered to individual mailboxes in your Amazon WorkMail organization. If one email is successfully delivered to 10 recipients in your organization, the `MailboxEmailDelivered` count is 10. Units: Count
`IncomingEmailBounced`	The number of incoming emails that bounced due to full mailboxes. This metric is counted for each intended recipient. For example, if one email is sent to 10 recipients in your organization, and two of the recipients have full mailboxes resulting in a bounce response, the `IncomingEmailBounced` count is two.

Metric	Description
	Units: Count
`OutgoingEmailBounced`	The number of outgoing emails that couldn't be delivered. This metric is counted for each intended recipient. For example, if one email is sent to 10 recipients, and two emails could not be delivered, the `OutgoingEmailBounced` count is 2. Units: Count
`OutgoingEmailSent`	The number of emails successfully sent from your Amazon WorkMail organization. This metric is counted for each recipient of a successfully sent email. For example, if 1 email is sent to 10 recipients, and the email was successfully delivered to 8 of the recipients, the `OutgoingEmailSent` count is 8. Units: Count
`AuthenticationFailure`	This metric counts the number of authentic ation attempts. When authentication is successful, the count is 0 and when authentic ation is unsuccessful, the count is 1. Use the `Sum` statistic to monitor the amount of failed authentication attempts. Use the `Sample count` statistic to monitor the total number of authentication events. Use the `Average` statistic to monitor the ratio of failed and successful authentication events. Units: Count

Metric	Description
AccessDenied	This metric counts the number of access control evaluations. When action is denied by access control, the count is 1 and when action is granted, the count is 0. Use the Sum statistic to monitor the volume of denied actions, the Sample count statistic to monitor the total number of attempted actions, and the Average statistic to monitor the ratio of allowed and denied actions. Units: Count
ActionDenied	This metric is counted when there's action on mailbox data. When action is denied, the count is 1 and if action is granted, the count is 0. Use the Sum statistic to monitor the volume of denied mailbox actions, the Sample count statistic to monitor the total number of attempted mailbox actions, and the Average statistic to monitor the ratio of allowed and denied actions. Units: Count
AvailabilityProviderFailure	This metric is counted for every *availabil ity provider* request that Amazon WorkMail executes to retrieve calendar availability from an external source. For more information about Availability Providers, see the Amazon WorkMail Administrator Guide.

Monitoring Amazon WorkMail email event logs

When you turn on email event logging for your Amazon WorkMail organization, Amazon WorkMail logs email events with CloudWatch. For more information about turning on email event logging, see Enabling email event logging.

The following tables describe the events that Amazon WorkMail logs with CloudWatch, when the events are transmitted, and what the event fields contain.

`ORGANIZATION_EMAIL_RECEIVED`

This event is logged when your Amazon WorkMail organization receives an email message.

Field	Description
recipients	The intended recipients of the message.
sender	The email address of the user who sent the email message on behalf of another user. This field is set only when an email is sent on behalf of another user.
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
subject	The email message subject.
messageId	The SMTP message ID.
spamVerdict	Indicates whether the message is marked as spam by Amazon SES. For more information, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.

Field	Description
dkimVerdict	Indicates whether the DomainKeys Identified Mail (DKIM) check passed. For more informati on, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
dmarcVerdict	Indicates whether the Domain-based Message Authentication, Reporting and Conformance (DMARC) check passed. For more information, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
dmarcPolicy	Appears only when the dmarcVerdict field contains "FAIL". Indicates the action to take on the email when the DMARC check fails (NONE, QUARANTINE, or REJECT). This is set by the owner of the sending email domain.
spfVerdict	Indicates whether the Sender Policy Framework (SPF) checks passed. For more information, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
messageTimestamp	Indicates when the message is received.

`MAILBOX_EMAIL_DELIVERED`

This event is logged when a message is delivered to a mailbox in your organization.
This is logged once for each mailbox to which a message is delivered, so a single `ORGANIZATION_EMAIL_RECEIVED` event can result in multiple `MAILBOX_EMAIL_DELIVERED` events.

Field	Description
recipient	The mailbox to which the message is delivered .
folder	The mailbox folder where the message is placed.

RULE_APPLIED

This event is logged when an incoming or outgoing message starts an email flow rule.

Field	Description
ruleName	The name of the rule.
ruleType	The type of rule applied (INBOUND_RULE, OUTBOUND_RULE, or MAILBOX_RULE). Inbound and outbound rules apply to your Amazon WorkMail organization. Mailbox rules apply only to specified mailboxes. For more information, see Managing email flows.
ruleActions	Actions taken based on the rule. Different recipients of the message might have different actions, such as a bounced email or a successfully delivered email.
targetFolder	Intended destination folder for a `Move` or `Copy` MAILBOX_RULE.
targetRecipient	Intended recipient of a `Forward` or `Redirect` MAILBOX_RULE.

JOURNALING_INITIATED

This event is logged when Amazon WorkMail sends an email to the journaling address specified by your organization administrator. This is only transmitted if journaling is configured for your organization. For more information, see Using email journaling with Amazon WorkMail.

Field	Description
journalingAddress	The email address to which the journaling message is sent.

INCOMING_EMAIL_BOUNCED

This event is logged when an incoming message can't be delivered to a target recipient. Emails can bounce for a number of reasons, such as a full target mailbox. The system logs this event once for each recipient that results in a bounced email. For example, if an incoming message is addressed to three recipients and two of them have full mailboxes, two INCOMING_EMAIL_BOUNCED events are logged.

Field	Description
bouncedRecipient	The intended recipient for which Amazon WorkMail bounced the message.

OUTGOING_EMAIL_SUBMITTED

This event is logged when a user in your organization submits an email message for sending. This is logged before the message leaves Amazon WorkMail, so this event doesn't indicate whether the email is successfully delivered.

Field	Description
recipients	The recipients of the message as specified by the sender. Includes all recipients on the To, CC, and BCC lines.
sender	The email address of the user who sent the email message on behalf of another user.

Field	Description
	This field is set only when an email is sent on behalf of another user.
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
subject	The email message subject.

`OUTGOING_EMAIL_SENT`

This event is logged when an outgoing email is successfully delivered to a target recipient. This is logged once for each successful recipient, so a single `OUTGOING_EMAIL_SUBMITTED` can result in multiple `OUTGOING_EMAIL_SENT` entries.

Field	Description
recipient	The recipient of the successfully delivered email.
sender	The email address of the user who sent the email message on behalf of another user. This field is set only when an email is sent on behalf of another user.
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.

Field	Description
messageId	The SMTP message ID.

OUTGOING_EMAIL_BOUNCED

This event is logged when an outgoing message can't be delivered to a target recipient. Emails can bounce for a number of reasons, such as a full target mailbox. The system logs a bounce for each recipient that results in a bounced email. For example, if an outgoing message is addressed to three recipients and two of them have full mailboxes, two `OUTGOING_EMAIL_BOUNCED` events are logged.

Field	Description
bouncedRecipient	The intended recipient for which the destinati on mail server bounced the message.

DMARC_POLICY_APPLIED

This event is logged when a DMARC policy is applied to an email sent to your organization.

Field	Description
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
recipients	The intended recipients of the message.
policy	The applied DMARC policy, indicating the action to take on the email when the DMARC check fails (NONE, QUARANTINE, or REJECT).

Field	Description
	This is the same as the dmarcPolicy field in the ORGANIZATION_EMAIL_RECEIVED event.

Monitoring Amazon WorkMail audit logs

You can use audit logs to monitor access to your Amazon WorkMail Organization's mailboxes. Amazon WorkMail logs five types of audit events and these events can be published to CloudWatch Logs, Amazon S3, or Amazon Firehouse. You can use audit logs to monitor user interaction with your Organization's mailboxes, authentication attempts, access control rule evaluation, and perform availability provider calls to external systems and monitor events with personal access tokens. For information about configuring audit logging, see Enabling audit logging.

The following sections describe the audit events logged by Amazon WorkMail, when the events are transmitted, and information about the event fields.

Mailbox access logs

Mailbox access events provide information about what action was taken (or attempted) on which mailbox object. A mailbox access event is generated for every operation that you attempt to run on an item or folder in a mailbox. These events are useful for auditing access to mailbox data.

Field	Description
event_timestamp	When the event happened, in milliseconds since Unix epoch.
request_id	The ID that uniquely identifies the request.
organization_arn	The ARN of the & Amazon WorkMail Organizat ion to which the authenticated user belongs.
user_id	The ID of the authenticated user.
impersonator_id	The ID of the impersonator. Present only if the impersonation feature was used for the request.

Field	Description
protocol	The protocol used. The protocol can be: `AutoDiscover` , `EWS`, `IMAP`, `WindowsOutlook` , `ActiveSync` , `SMTP`, `WebMail`, `IncomingEmail` , or `OutgoingEmail` .
source_ip	The source IP address of the request.
user_agent	The user agent that made the request.
action	The action taken on the object, which can be: `read`, `read_hierarchy` , `read_summary` , `read_attachment` , `read_permissions` , `create`, `update`, `update_permissions` , `update_read_state` , `delete`, `submit_email_for_sending` , `abort_sending_email` , `move`, `move_to`, `copy`, or `copy_to`.
owner_id	The ID of the user that owns the object being acted upon.
object_type	The object type, which can be: Folder, Message, or Attachment.
item_id	The ID that uniquely identifies the message that's the subject of the event or that contains the attachment that's the subject of the event.
folder_path	The path of the folder being acted upon or the path of the folder containing the item being acted upon.
folder_id	The ID that uniquely identifies the folder that's the subject of the event or contains the object that's the subject of the event.

Field	Description
attachment_path	The path of display names to the affected attachment.
action_allowed	Whether the action was allowed. Can be true or false.

Access control logs

Access control events are generated whenever an access control rule is evaluated. These logs are useful for auditing forbidden access, or debugging access control configurations.

Field	Description
event_timestamp	When the event happened, in milliseconds since Unix epoch.
request_id	The ID that uniquely identifies the request.
organization_arn	The ARN of the WorkMail Organization to which the authenticated user belongs.
user_id	The ID of the authenticated user.
impersonator_id	The ID of the impersonator. Present only if the impersonation feature was used for the request.
protocol	The protocol used, which can be: `AutoDiscover` , `EWS`, `IMAP`, `WindowsOutlook` , `ActiveSync` , `SMTP`, `WebMail`, `IncomingEmail` , or `OutgoingEmail` .
source_ip	The source IP address of the request.

Field	Description
scope	The scope of the rule, which can be: `AccessControl` , `DeviceAccessControl` , or `ImpersonationAccessControl` .
rule_id	The ID of the matched access control rule. When there are no rules matched, *rule_id* is not available.
access_granted	Whether access was allowed. Can be true or false.

Authentication logs

Authentication events contain information about authentication attempts.

> **Note**
>
> Authentication events are not generated for authentication events through the Amazon WorkMail WebMail application.

Field	Description
event_timestamp	When the event happened, in milliseconds since Unix epoch.
request_id	The ID that uniquely identifies the request.
organization_arn	The ARN of the WorkMail Organization to which the authenticated user belongs.
user_id	The ID of the authenticated user.
user	The username that the authentication was attempted with.

Field	Description
protocol	The protocol used, which can be: `AutoDiscover` , `EWS`, `IMAP`, `WindowsOutlook` , `ActiveSync` , `SMTP`, `WebMail`, `IncomingEmail` , or `OutgoingEmail` .
source_ip	The source IP address of the request.
user_agent	The user agent that made the request.
method	The auth method. Currently, only basic is supported.
auth_successful	Whether the auth attempt was successful. Can be true or false.
auth_failed_reason	The reason for auth failure. Present only if auth failed.
personal_access_token_id	The ID of the personal access token used for authentication.

Personal access token logs

A personal access token (PAT) event is generated for every attempt in creating or deleting a personal access token. Personal access token events provide information about whether users successfully create personal access tokens. The personal access token logs are useful for auditing end users creating and deleting their own PATs. User login with personal access tokens will generate events in the existing Authentication logs. For more information, see Authentication logs .

Field	Description
event_timestamp	When the event happened, in milliseconds since Unix epoch.
request_id	The ID that uniquely identifies the request.

Field	Description
organization_arn	The ARN of the WorkMail Organization to which the authenticated user belongs.
user_id	The ID of the authenticated user.
user	The username of the user who took this action.
protocol	The protocol used through the action took place, which can be: webapp
source_ip	The source IP address of the request.
user_agent	The user agent that made the request.
action	The action of the personal access token, which can be: create or delete.
name	The name of the personal access token.
expires_time	The date when the personal access token expires.
scopes	The scopes of the personal access token permissions on mailbox.

Availability provider logs

Availability provider events are generated for every availability request Amazon WorkMail does on your behalf to your configured availability provider. These events are useful for debugging your availability provider configuration.

Field	Description
event_timestamp	When the event happened, in milliseconds since Unix epoch.

Field	Description
request_id	The ID that uniquely identifies the request.
organization_arn	The ARN of the WorkMail Organization to which the authenticated user belongs.
user_id	The ID of the authenticated user.
type	The type of availability provider being invoked, which can be: `EWS` or `LAMBDA`.
domain	The domain for which availability is obtained.
function_arn	The ARN of the invoked Lambda, if type is LAMBDA. Otherwise, this field is not present.
ews_endpoint	The EWS endpoint is type is EWS. Otherwise, this field is not present.
error_message	The message describing the cause of the failure. If the request was successful, this field is not present.
availability_event_successful	Whether the availability request was served successfully.

Using CloudWatch Insights with Amazon WorkMail

If you turned on email event logging in the Amazon WorkMail console or enabled audit logs delivery to CloudWatch Logs, you can use Amazon CloudWatch Logs Insights to query your event logs. For more information about turning on email event logging, see Enabling email event logging. For more information about CloudWatch Logs Insights, see Analyze log data with CloudWatch Logs Insights in the *Amazon CloudWatch Logs User Guide*.

The following examples demonstrate how to query CloudWatch Logs for common email events. You run these queries in the CloudWatch console. For instructions about how to run these queries, see Tutorial: Run and modify a sample query in the *Amazon CloudWatch Logs User Guide*.

Example See why User B did not receive an email sent by User A.

The following code example demonstrates how to query for an outgoing email sent by User A to User B, sorted by timestamp.

```
fields @timestamp, traceId

| sort @timestamp asc
| filter (event.from like /(?i)userA@example.com/
and event.eventName = "OUTGOING_EMAIL_SUBMITTED"
and event.recipients.0 like /(?i)userB@example.com/)
```

This returns the sent message and trace ID. Use the trace ID in the following code example to query the event logs for the sent message.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter traceId = "$TRACEID"
```

This returns the email message ID and the email events. `OUTGOING_EMAIL_SENT` indicates that the email was sent. `OUTGOING_EMAIL_BOUNCED` indicates that the email bounced. To see whether the email was received, query using the message ID in the following code example.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter event.messageId like "$MESSAGEID"
```

This should also return the received message, because it has the same message ID. Use the trace ID in the following code example to query for delivery.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter traceId = "$TRACEID"
```

This returns the delivery action and any applicable rule actions.

Example See all mail received from a user or domain

The following code example demonstrates how to query for all mail received from a specified user.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter (event.from like /(?i)user@example.com/ and event.eventName =
 "ORGANIZATION_EMAIL_RECEIVED")
```

The following code example demonstrates how to query for all mail received from a specified domain.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter (event.from like "example.com" and event.eventName =
 "ORGANIZATION_EMAIL_RECEIVED")
```

Example See who sent bounced emails

The following code example demonstrates how to query for outgoing emails that bounced, and also returns the reasons for bouncing.

```
fields @timestamp, event.destination, event.reason
| sort @timestamp desc
| filter event.eventName = "OUTGOING_EMAIL_BOUNCED"
```

The following code example demonstrates how to query for incoming emails that bounced. It also returns the bounced recipients' email addresses and the reasons for bouncing.

```
fields @timestamp, event.bouncedRecipient.emailAddress, event.bouncedRecipient.reason,
 event.bouncedRecipient.status
| sort @timestamp desc
| filter event.eventName = "INCOMING_EMAIL_BOUNCED"
```

Example See which domains are sending spam

The following code example demonstrates how to query for recipients in your organization that are receiving spam.

```
stats count(*) as c by event.recipients.0
| filter (event.eventName = "ORGANIZATION_EMAIL_RECEIVED" and event.spamVerdict =
 "FAIL")
| sort c desc
```

The following code example demonstrates how to query for the sender of the spam emails.

```
fields @timestamp, event.recipients.0, event.sender, event.from
| sort @timestamp asc
| filter (event.spamVerdict = "FAIL")
```

Example See why an email was sent to a recipient's spam folder

The following code example demonstrates how to query for emails identified as spam, filtered by subject.

```
fields @timestamp, event.recipients.0, event.spamVerdict, event.spfVerdict,
 event.dkimVerdict, event.dmarcVerdict
| sort @timestamp asc
| filter event.subject like /(?i)$SUBJECT/ and event.eventName =
 "ORGANIZATION_EMAIL_RECEIVED"
```

You can also query by the email trace ID to see all events for the email.

Example See emails that match email flow rules

The following code example demonstrates how to query for emails that matched outbound email flow rules.

```
fields @timestamp, event.ruleName, event.ruleActions.0.action
| sort @timestamp desc
| filter event.ruleType = "OUTBOUND_RULE"
```

The following code example demonstrates how to query for emails that matched inbound email flow rules.

```
fields @timestamp, event.ruleName, event.ruleActions.0.action,
 event.ruleActions.0.recipients.0
| sort @timestamp desc
| filter event.ruleType = "INBOUND_RULE"
```

Example See how many emails are received or sent by your organization

The following code example demonstrates how to query for the number of emails received by each recipient in your organization.

```
stats count(*) as c by event.recipient
| filter event.eventName = "MAILBOX_EMAIL_DELIVERED"
| sort c desc
```

The following code example demonstrates how to query for the number of emails sent by each sender in your organization.

```
stats count(*) as c by event.from
| filter event.eventName = "OUTGOING_EMAIL_SUBMITTED"
| sort c desc
```

Logging Amazon WorkMail API calls with AWS CloudTrail

Amazon WorkMail is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon WorkMail. CloudTrail captures all API calls for Amazon WorkMail as events, including calls from the Amazon WorkMail console and from code calls to the Amazon WorkMail APIs. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon WorkMail. 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 Amazon WorkMail, 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.

Amazon WorkMail information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in Amazon WorkMail, 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.

For an ongoing record of events in your AWS account, including events for Amazon WorkMail, you must 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 Regions in the 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:

- 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 Amazon WorkMail actions are logged by CloudTrail and are documented in the Amazon WorkMail API Reference. For example, calls to the `CreateUser`, `CreateAlias`, and `GetRawMessageContent` API 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 IAM user 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 the CloudTrail userIdentity element.

Understanding Amazon WorkMail 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, 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 `CreateUser` action from the Amazon WorkMail API.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111111111111:user/WMSDK",
    "accountId": "111111111111",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
```

```
    "userName": "WMSDK"
  },
  "eventTime": "2017-12-12T17:49:59Z",
  "eventSource": "workmail.amazonaws.com",
  "eventName": "CreateUser",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-sdk-java/1.11.205 Mac_OS_X/10.11.6 Java_HotSpot(TM)_64-
Bit_Server_VM/25.151-b12 java/1.8.0_151",
  "requestParameters": {
    "name": "janedoe",
    "displayName": "Jane Doe",
    "organizationId": "m-5b1c980000EXAMPLE"
  },
  "responseElements": {
    "userId": "a3a9176d-EXAMPLE"
  },
  "requestID": "dec81e4a-EXAMPLE",
  "eventID": "9f2f09c5-EXAMPLE",
  "eventType": "AwsApiCall",
  "recipientAccountId": "111111111111"
}
```

The following example shows a CloudTrail log entry that demonstrates the `CreateAlias` action from the Amazon WorkMail API.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111111111111:user/WMSDK",
    "accountId": "111111111111",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "WMSDK"
  },
  "eventTime": "2017-12-12T18:13:44Z",
  "eventSource": "workmail.amazonaws.com",
  "eventName": "CreateAlias",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-sdk-java/1.11.205 Mac_OS_X/10.11.6 Java_HotSpot(TM)_64-
Bit_Server_VM/25.151-b12 java/1.8.0_151",
```

```
  "requestParameters": {
    "alias": "aliasjamesdoe@testofconsole.awsapps.com",
    "organizationId": "m-5b1c980000EXAMPLE"
    "entityId": "a3a9176d-EXAMPLE"
  },
  "responseElements": null,
  "requestID": "dec81e4a-EXAMPLE",
  "eventID": "9f2f09c5-EXAMPLE",
  "eventType": "AwsApiCall",
  "recipientAccountId": "111111111111"
}
```

The following example shows a CloudTrail log entry that demonstrates the `GetRawMessageContent` action from the Amazon WorkMail Message Flow API.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111111111111:user/WMSDK",
    "accountId": "111111111111",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "WMSDK"
  },
  "eventTime": "2017-12-12T18:13:44Z",
  "eventSource": "workmailMessageFlow.amazonaws.com",
  "eventName": "GetRawMessageContent",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-sdk-java/1.11.205 Mac_OS_X/10.11.6 Java_HotSpot(TM)_64-
Bit_Server_VM/25.151-b12 java/1.8.0_151",
  "requestParameters": {
    "messageId": "123A4A5A-67B8-90C1-D23E-45FG67H890J1"
  },
  "responseElements": null,
  "requestID": "dec81e4a-EXAMPLE",
  "eventID": "9f2f09c5-EXAMPLE",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "recipientAccountId": "111111111111"
}
```

Enabling email event logging

You enable email event logging in the Amazon WorkMail console in order to track email messages for your organization. Email event logging uses an AWS Identity and Access Management service-linked role (SLR) to grant permissions to publish the email event logs to Amazon CloudWatch. For more information about IAM service-linked roles, see Using service-linked roles for Amazon WorkMail.

In the CloudWatch event logs, you can use CloudWatch search tools and metrics to track messages and troubleshoot email issues. For more information about the event logs that Amazon WorkMail sends to CloudWatch, see Monitoring Amazon WorkMail email event logs. For more information about CloudWatch Logs, see the Amazon CloudWatch Logs User Guide.

Topics

- Turning on email event logging
- Creating a custom log group and IAM role for email event logging
- Turning off email event logging
- Cross-service confused deputy prevention

Turning on email event logging

The following occurs when you turn on email event logging using the default settings, Amazon WorkMail:

- Creates an AWS Identity and Access Management service-linked role – `AmazonWorkMailEvents`.
- Creates a CloudWatch log group – `/aws/workmail/emailevents/`*`organization-alias`*.
- Sets CloudWatch log retention to 30 days.

To turn on email event logging

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Logging settings**.
4. Choose the **Email flow log settings** tab.
5. In the **Email flow log settings** section, choose **Edit**.
6. Move the **Enable mail events** slider to the **on** position.
7. Do one of the following:

   - (Recommended) Choose **Use default settings**.
   - (Optional) Clear the **Use default settings**, and select a **Destination Log Group** and **IAM Role** from the lists that appear.

   > **Note**
   >
   > Choose this option only if you have already created a log group and custom IAM role using the AWS CLI. For more information, see Creating a custom log group and IAM role for email event logging.

8. Select **I authorize Amazon WorkMail to publish logs in my account using this configuration**.
9. Choose **Save**.

Creating a custom log group and IAM role for email event logging

We recommend using the default settings when enabling email event logging for Amazon WorkMail. If you require a custom monitoring configuration, you can use the AWS CLI to create a dedicated log group and custom IAM role for email event logging.

To create a custom log group and IAM role for email event logging

1. Use the following AWS CLI command to create a log group in the same AWS Region as your Amazon WorkMail organization. For more information, see create-log-group in the *AWS CLI Command Reference*.

   ```
   aws --region us-east-1 logs create-log-group --log-group-name workmail-monitoring
   ```

2. Create a file containing the following policy:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "events.workmail.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

3. Use the following AWS CLI command to create an IAM role and attach this file as the role policy document. For more information, see create-role in the *AWS CLI Command Reference*.

```
aws iam create-role --role-name workmail-monitoring-role --assume-role-policy-
document file://trustpolicyforworkmail.json
```

> **Note**
>
> If you're a `WorkMailFullAccess` managed policy user, you must include the term `workmail` in the role name. This managed policy only allows you to configure email event logging using roles with `workmail` in the name. For more information, see Granting a user permissions to pass a role to an AWS service in the *IAM User Guide*.

4. Create a file containing the policy for the IAM role that you created in the previous step. At minimum, the policy must grant permissions to the role to create log streams and put log events into the log group that you created in step 1.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:us-east-1:111122223333:log-
group:workmail-monitoring*"
        }
    ]
}
```

5. Use the following AWS CLI command to attach the policy file to the IAM role. For more information, see put-role-policy in the *AWS CLI Command Reference*.

```
aws iam put-role-policy --role-name workmail-monitoring-role --policy-
name workmail-permissions --policy-document file://rolepolicy.json
```

Turning off email event logging

Turn off email event logging from the Amazon WorkMail console. If you no longer need to use email event logging, we recommend that you delete the related CloudWatch log group and service-linked role as well. For more information, see Deleting a service-linked role for Amazon WorkMail.

To turn off email event logging

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Monitoring**.
4. In the **Log settings** section, choose **Edit**.
5. Move the **Enable mail events** slider to the off position.
6. Choose **Save**.

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 it wouldn't 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 `aws:SourceArn` and `aws:SourceAccount` global condition context keys in resource policies to limit the permissions that CloudWatch Logs and Amazon S3 give to the services that are generating logs. If you use both global condition context keys, the values must use the same account ID when used in the same policy statement.

The values of `aws:SourceArn` must be the ARNs of the delivery sources that are generating logs.

The most effective way to protect against the confused deputy problem is to use the `aws:SourceArn` global condition context key with the full ARN of the resource. If you don't know the full ARN of the resource or if you're specifying multiple resources, use the `aws:SourceArn` global context condition key with wildcards (*) for the unknown portions of the ARN.

Enabling audit logging

You can use audit logs to capture detailed information about your Amazon WorkMail organization usage. The audit logs can be used to monitor user's access to mailboxes, audit for suspicious activity, and debug access control and availability provider configurations.

> **Note**
>
> The *AmazonWorkMailFullAccess* managed policy does not include all the required permissions to manage log deliveries. If you are using this policy to manage WorkMail, make sure the principal (for example, the assumed role) used to configure log deliveries also has all the required permissions.

Amazon WorkMail supports three delivery destinations for audit logs: CloudWatch Logs, Amazon S3, and Amazon Data Firehose. For more information, see Logging that requires additional permissions [V2] in the *Amazon CloudWatch Logs User Guide*.

In addition to the permissions listed under Logging that requires additional permissions [V2], Amazon WorkMail requires an additional permission to configure log delivery: `workmail:AllowVendedLogDeliveryForResource`.

A working log delivery consists of three elements:

- *DeliverySource*, a logical object that represents the resource or resources that send the logs. For Amazon WorkMail, it's the Amazon WorkMail Organization.
- A *DeliveryDestination*, which is a logical object that represents the actual delivery destination.
- A *Delivery*, which connects a delivery source to delivery destination.

To configure log delivery between Amazon WorkMail and a destination, you can do the following:

- Create a delivery source with PutDeliverySource.
- Create a delivery destination with PutDeliveryDestination.
- If you're delivering logs cross-account, you must use PutDeliveryDestinationPolicy in the destination account to assign an IAM policy to the destination. This policy authorizes creating a delivery from the delivery source in account A to the delivery destination in account B.
- Create a delivery by pairing exactly one delivery source and one delivery destination by using CreateDelivery.

The following sections provide the details of the permissions that you must have when you're signed in to set up log delivery to each type of destination. These permissions can be granted to an IAM role that you're signed in with.

> **Important**
>
> It's your responsibility to remove log delivery resources after deleting the log-generating resource.

To remove log delivery resources after deleting the log-generating resource, follow these steps.

1. Delete the *Delivery* by using the DeleteDelivery operation.
2. Delete the *DeliverySource* by using the DeleteDeliverySource operation.
3. If the *DeliveryDestination* associated with the *DeliverySource* that you just deleted is used only for this specific *DeliverySource*, then you can remove it by using the DeleteDeliveryDestinations operation.

Configuring audit logging using the Amazon WorkMail console

You can configure audit logging in the Amazon WorkMail console:

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and select a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. Choose **Logging settings**.
4. Choose the **Audit log settings** tab.
5. Configure deliveries for the required log type using the appropriate widget.
6. Choose **Save**.

Logs sent to CloudWatch Logs

User permissions

To enable sending logs to CloudWatch Logs, you must be signed in with the following permissions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ReadWriteAccessForLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:GetDelivery",
                "logs:GetDeliverySource",
                "logs:PutDeliveryDestination",
                "logs:GetDeliveryDestinationPolicy",
```

```
                "logs:DeleteDeliverySource",
                "logs:PutDeliveryDestinationPolicy",
                "logs:CreateDelivery",
                "logs:GetDeliveryDestination",
                "logs:PutDeliverySource",
                "logs:DeleteDeliveryDestination",
                "logs:DeleteDeliveryDestinationPolicy",
                "logs:DeleteDelivery"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:delivery:*",
                "arn:aws:logs:region:account-id:delivery-source:*",
                "arn:aws:logs:region:account-id:delivery-destination:*"
            ]
        },
        {
            "Sid": "ListAccessForLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeDeliveryDestinations",
                "logs:DescribeDeliverySources",
                "logs:DescribeDeliveries",
                "logs:DescribeLogGroups"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowUpdatesToResourcePolicyCWL",
            "Effect": "Allow",
            "Action": [
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:*"
            ]
        }
        {
            "Sid":"AllowLogDeliveryForWorkMail",
            "Effect":"Allow",
            "Action":[
                "workmail:AllowVendedLogDeliveryForResource"
            ],
```

```
            "Resource":[
                "arn:aws:workmail:region:account-id:organization/organization-id"
            ]
        }
    ]
}
```

Log group resource policy

The log group where the logs are being sent must have a resource policy that includes certain permissions. If the log group currently does not have a resource policy, and the user setting up the logging has the `logs:PutResourcePolicy`, `logs:DescribeResourcePolicies`, and `logs:DescribeLogGroups` permissions for the log group, then AWS automatically creates the following policy for it when you begin sending the logs to CloudWatch Logs.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"AWSLogDeliveryWrite20150319",
            "Effect":"Allow",
            "Principal":{
                "Service":[
                    "delivery.logs.amazonaws.com"
                ]
            },
            "Action":[
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource":[
                "arn:aws:logs:region:account-id:log-group:my-log-group:log-
stream:*"
            ],
            "Condition":{
                "StringEquals":{
                    "aws:SourceAccount":[
                        "account-id"
                    ]
                },
```

```
                "ArnLike":{
                    "aws:SourceArn":[
                        "arn:aws:logs:region:account-id:*"
                    ]
                }
            }
        }
    ]
}
```

Log group resource policy size limit considerations

These services must list each log group that they're sending logs to in the resource policy. CloudWatch Logs resource policies are limited to 5,120 characters. A service that sends logs to a large number of log groups might run into this limit.

To mitigate this, CloudWatch Logs monitors the size of resource policies used by the service that's sending logs. When it detects that a policy approaches the size limit of 5,120 characters, CloudWatch Logs automatically enables `/aws/vendedlogs/*` in the resource policy for that service. You can then start using log groups with names that start with `/aws/vendedlogs/` as the destinations for logs from these services.

Logs sent to Amazon S3

User permissions

To enable sending logs to Amazon S3, you must be signed in with the following permissions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ReadWriteAccessForLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:GetDelivery",
                "logs:GetDeliverySource",
                "logs:PutDeliveryDestination",
                "logs:GetDeliveryDestinationPolicy",
                "logs:DeleteDeliverySource",
                "logs:PutDeliveryDestinationPolicy",
                "logs:CreateDelivery",
```

```
                "logs:GetDeliveryDestination",
                "logs:PutDeliverySource",
                "logs:DeleteDeliveryDestination",
                "logs:DeleteDeliveryDestinationPolicy",
                "logs:DeleteDelivery"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:delivery:*",
                "arn:aws:logs:region:account-id:delivery-source:*",
                "arn:aws:logs:region:account-id:delivery-destination:*"
            ]
        },
        {
            "Sid": "ListAccessForLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeDeliveryDestinations",
                "logs:DescribeDeliverySources",
                "logs:DescribeDeliveries",
                "logs:DescribeLogGroups"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowUpdatesToResourcePolicyS3",
            "Effect": "Allow",
            "Action": [
                "s3:PutBucketPolicy",
                "s3:GetBucketPolicy"
            ],
            "Resource": "arn:aws:s3:::bucket-name"
        }
          {
            "Sid":"AllowLogDeliveryForWorkMail",
            "Effect":"Allow",
            "Action":[
                "workmail:AllowVendedLogDeliveryForResource"
            ],
            "Resource":[
                "arn:aws:workmail:region:account-id:organization/organization-id"
            ]
        }
    ]
```

```
}
```

The S3 bucket where the logs are being sent must have a resource policy that includes certain permissions. If the bucket currently doesn't have a resource policy and the user setting up the logging has the `S3:GetBucketPolicy` and `S3:PutBucketPolicy` permissions for the bucket, then AWS automatically creates the following policy for it when you begin sending the logs to Amazon S3.

JSON

```
{
    "Version":"2012-10-17",
    "Id":"AWSLogDeliveryWrite20150319",
    "Statement":[
        {
            "Sid":"AWSLogDeliveryAclCheck",
            "Effect":"Allow",
            "Principal":{
                "Service":"delivery.logs.amazonaws.com"
            },
            "Action":"s3:GetBucketAcl",
            "Resource":"arn:aws:s3:::my-bucket",
            "Condition":{
                "StringEquals":{
                    "aws:SourceAccount":[
                        "account-id"
                    ]
                },
                "ArnLike":{
                    "aws:SourceArn":[
                        "arn:aws:logs:region:account-id:delivery-source:*"
                    ]
                }
            }
        },
        {
            "Sid":"AWSLogDeliveryWrite",
            "Effect":"Allow",
            "Principal":{
                "Service":"delivery.logs.amazonaws.com"
            },
            "Action":"s3:PutObject",
```

```
            "Resource":"arn:aws:s3:::my-bucket/AWSLogs/account-id/*",
            "Condition":{
                "StringEquals":{
                    "s3:x-amz-acl":"bucket-owner-full-control",
                    "aws:SourceAccount":[
                        "account-id"
                    ]
                },
                "ArnLike":{
                    "aws:SourceArn":[
                        "arn:aws:logs:region:account-id:delivery-source:*"
                    ]
                }
            }
        }
    ]
}
```

In the previous policy, for `aws:SourceAccount`, specify the list of account IDs for which logs are being delivered to this bucket. For `aws:SourceArn`, specify the list of ARNs of the resource that generates the logs, in the form `arn:aws:logs:`*`source-region`*`:`*`source-account-id`*`:*`.

If the bucket has a resource policy, but that policy doesn't contain the statement shown in the previous policy, and the user setting up the logging has the `S3:GetBucketPolicy` and `S3:PutBucketPolicy` permissions for the bucket, that statement is appended to the bucket's resource policy.

Note

In some cases, you might see `AccessDenied` errors in AWS CloudTrail if the `s3:ListBucket` permission hasn't been granted to `delivery.logs.amazonaws.com`. To avoid these errors in your CloudTrail logs, you must grant the `s3:ListBucket` permission to `delivery.logs.amazonaws.com`. You must also include the `Condition` parameters shown with the `s3:GetBucketAcl` permission set in the preceding bucket policy. To streamline this, instead of creating a new `Statement`, you can directly update the `AWSLogDeliveryAclCheck` to be `“Action”: [“s3:GetBucketAcl”, “s3:ListBucket”]`.

Amazon S3 bucket server-side encryption

You can protect the data in your Amazon S3 bucket by enabling either server-side encryption with Amazon S3-managed keys (SSE-S3) or server-side encryption with an AWS KMS key stored in AWS Key Management Service (SSE-KMS). For more information, see Protecting data using server-side encryption.

If you choose SSE-S3, no additional configuration is required. Amazon S3 handles the encryption key.

> **Warning**
>
> If you choose SSE-KMS, you must use a customer managed key, because using an AWS managed key isn't supported for this scenario. If you set up encryption using an AWS managed key, the logs will be delivered in an unreadable format.

When you use a customer managed AWS KMS key, you can specify the Amazon Resource Name (ARN) of the customer managed key when you enable bucket encryption. Add the following to the key policy for your customer managed key (not to the bucket policy for your S3 bucket), so that the log delivery account can write to your S3 bucket.

If you choose SSE-KMS, you must use a customer managed key, because using an AWS managed key isn't supported for this scenario. When you use a customer managed AWS KMS key, you can specify the Amazon Resource Name (ARN) of the customer managed key when you enable bucket encryption. Add the following to the key policy for your customer managed key (not to the bucket policy for your S3 bucket), so that the log delivery account can write to your S3 bucket.

```
{
    "Sid":"Allow Logs Delivery to use the key",
    "Effect":"Allow",
    "Principal":{
        "Service":[
            "delivery.logs.amazonaws.com"
        ]
    },
    "Action":[
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
```

```
            "kms:DescribeKey"
        ],
        "Resource":"*",
        "Condition":{
            "StringEquals":{
                "aws:SourceAccount":[
                    "account-id"
                ]
            },
            "ArnLike":{
                "aws:SourceArn":[
                    "arn:aws:logs:region:account-id:delivery-source:*"
                ]
            }
        }
}
```

For `aws:SourceAccount`, specify the list of account IDs for which logs are being delivered to this bucket. For `aws:SourceArn`, specify the list of ARNs of the resource that generates the logs, in the form `arn:aws:logs:`*`source-region`*`:`*`source-account-id`*`:*`.

Logs sent to Firehose

User permissions

To enable sending logs to Firehose, you must be signed in with the following permissions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ReadWriteAccessForLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:GetDelivery",
                "logs:GetDeliverySource",
                "logs:PutDeliveryDestination",
                "logs:GetDeliveryDestinationPolicy",
                "logs:DeleteDeliverySource",
                "logs:PutDeliveryDestinationPolicy",
                "logs:CreateDelivery",
                "logs:GetDeliveryDestination",
                "logs:PutDeliverySource",
```

```
                "logs:DeleteDeliveryDestination",
                "logs:DeleteDeliveryDestinationPolicy",
                "logs:DeleteDelivery"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:delivery:*",
                "arn:aws:logs:region:account-id:delivery-source:*",
                "arn:aws:logs:region:account-id:delivery-destination:*"
            ]
        },
        {
            "Sid": "ListAccessForLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeDeliveryDestinations",
                "logs:DescribeDeliverySources",
                "logs:DescribeDeliveries",
                "logs:DescribeLogGroups"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowUpdatesToResourcePolicyFH",
            "Effect": "Allow",
            "Action": [
                "firehose:TagDeliveryStream"
            ],
            "Resource": [
                "arn:aws:firehose:region:account-id:deliverystream/*"
            ]
        },
        {
            "Sid": "CreateServiceLinkedRole",
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
delivery.logs.amazonaws.com/AWSServiceRoleForLogDelivery"
        }
           {
            "Sid":"AllowLogDeliveryForWorkMail",
            "Effect":"Allow",
            "Action":[
```

```
                "workmail:AllowVendedLogDeliveryForResource"
            ],
            "Resource":[
                "arn:aws:workmail:region:account-id:organization/organization-id"
            ]
        }
    ]
}
```

IAM roles used for resource permissions

Because Firehose doesn't use resource policies, AWS uses IAM roles when setting up these logs to be sent to Firehose. AWS creates a service-linked role named **AWSServiceRoleForLogDelivery**. This service-linked role includes the following permissions.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "firehose:PutRecord",
                "firehose:PutRecordBatch",
                "firehose:ListTagsForDeliveryStream"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/LogDeliveryEnabled": "true"
                }
            },
            "Effect": "Allow"
        }
    ]
}
```

This service-linked role grants permission for all Firehose delivery streams that have the `LogDeliveryEnabled` tag set to `true`. AWS gives this tag to the destination delivery stream when you set up the logging.

This service-linked role also has a trust policy that allows the `delivery.logs.amazonaws.com` service principal to assume the needed service-linked role. That trust policy is as follows:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Console-specific permissions

In addition to the permissions listed in the previous sections, if you're setting up log delivery using the console instead of the APIs, you also need the following permissions:

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"AllowLogDeliveryActions",
            "Effect":"Allow",
            "Action":[
                "firehose:DescribeDeliveryStream",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource":[
                "arn:aws:logs:region:account-id:log-group:*",
                "arn:aws:firehose:region:account-id:deliverystream/*",
                "arn:aws:s3:::*"
```

```
            ]
        },
        {
            "Sid":"ListAccessForDeliveryDestinations",
            "Effect":"Allow",
            "Action":[
                "logs:DescribeLogGroups",
                "firehose:ListDeliveryStreams",
                "s3:ListAllMyBuckets"
            ],
            "Resource":"*"
        }
    ]
}
```

Compliance validation for Amazon WorkMail

Third-party auditors assess the security and compliance of Amazon WorkMail as part of multiple AWS compliance programs. These include SOC, ISO, and C5.

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading reports in AWS Artifact.

Your compliance responsibility when using Amazon WorkMail 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:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Config – This AWS service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.

- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in Amazon WorkMail

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.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

In addition to the AWS global infrastructure, Amazon WorkMail offers several features to help support your data resiliency and backup needs.

Infrastructure security in Amazon WorkMail

> **Note**
>
> Amazon WorkMail discontinued support for Transport Layer Security (TLS) 1.0 and 1.1. If you are using TLS 1.0 or 1.1, you must upgrade the TLS version to 1.2. For more information, see TLS 1.2 to become the minimum TLS protocol level for all AWS API endpoints.

As a managed service, Amazon WorkMail 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 Amazon WorkMail 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.

Getting started with Amazon WorkMail

After you complete the Prerequisites, you're ready to get started with Amazon WorkMail. For more information, see Getting started with Amazon WorkMail.

You can learn more about migrating existing mailboxes to Amazon WorkMail, interoperability with Microsoft Exchange, and Amazon WorkMail quotas in the following sections.

Topics

- Getting started with Amazon WorkMail
- Migrating to Amazon WorkMail
- Interoperability between Amazon WorkMail and Microsoft Exchange
- Configure availability settings on Amazon WorkMail
- Configure availability settings in Microsoft Exchange
- Enable email routing between Microsoft Exchange and Amazon WorkMail users
- Enable email routing for a user
- Post setup configuration
- Mail client configuration
- Disabling interoperability mode and decommissioning your mail server
- Troubleshooting
- Amazon WorkMail quotas

Getting started with Amazon WorkMail

Whether you're a new Amazon WorkMail user or an existing user of Amazon WorkSpaces, get started with Amazon WorkMail by completing the following steps.

> **Note**
>
> Complete the Prerequisites before getting started.

Topics

- Step 1: Sign in to the Amazon WorkMail console
- Step 2: Set up your Amazon WorkMail site
- Step 3: Set up Amazon WorkMail user access
- More resources

Step 1: Sign in to the Amazon WorkMail console

You must sign in to the Amazon WorkMail console before you can add users and manage their accounts and mailboxes.

To sign in to the Amazon WorkMail console

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information about regions, see Regions and endpoints in the *Amazon Web Services General Reference*.

Step 2: Set up your Amazon WorkMail site

1. After you sign in to the Amazon WorkMail console, you set up your organization and add a domain. We recommend using a dedicated domain for your Amazon WorkMail organization. For more information, see Creating an organization and Adding a domain.
2. (Optional) You can choose to use a free testing domain provided by Amazon WorkMail. If you choose to do this, skip to step 4.

   > **Note**
   >
   > Test domains use this format: *alias*.awsapps.com. As you go, remember that you should only use test domains for testing. Don't use a test domain for a production environment. Also, you must have at least one enabled user in your Amazon WorkMail organization. If you don't have an enabled user, the domain can become available for registration and use by other customers.

3. If you use an external domain, verify that domain by adding the appropriate text (TXT) and mail exchange (MX) records to your Domain Name System (DNS) service. TXT records allow you to enter notes in the DNS. MX records specify the incoming mail servers. Make sure to set your

domain as the default for your organization. For more information, see Verifying domains and Choosing the default domain.

4. Create new users or enable your existing directory users for Amazon WorkMail. For more information, see Adding a user.
5. (Optional) If you have existing Microsoft Exchange mailboxes, migrate them to Amazon WorkMail. For more information, see Migrating to Amazon WorkMail.

After you've finished setting up your Amazon WorkMail site, you can access Amazon WorkMail using the web application URL.

To locate your Amazon WorkMail web application URL

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. To do so, open the **Select a region** list, located to the right of the search box, then choose the desired Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.

   The **Organization settings** page appears and displays the URL under **User login**. The URLs take this form: https://*alias*.awsapps.com/mail.

Step 3: Set up Amazon WorkMail user access

Choose from the following options to set up Amazon WorkMail user access:

- Set up user access from an existing desktop client using the Microsoft Outlook client. For more information, see Connect Microsoft Outlook to your Amazon WorkMail account.
- Set up user access from a mobile device, such as a Kindle, Android, iPad, or iPhone. For more information, see Getting started with a mobile device.
- To set up user access, use any client software compatible with the Internet Mail Access Protocol (IMAP) protocol. For more information, see Connect IMAP clients to Your Amazon WorkMail account.

More resources

- Migrating to Amazon WorkMail
- Interoperability between Amazon WorkMail and Microsoft Exchange
- Amazon WorkMail quotas

Migrating to Amazon WorkMail

You can migrate to Amazon WorkMail from Microsoft Exchange, Microsoft Office 365, G Suite Basic (formerly Google Apps for Work), and other platforms by working with one of our partners. For more information about our partners, see Amazon WorkMail Features.

Topics

- Step 1: Create or enable users in Amazon WorkMail
- Step 2: Migrate to Amazon WorkMail
- Step 3: Complete the migration to Amazon WorkMail

Step 1: Create or enable users in Amazon WorkMail

Before you migrate your users, you must add those users in Amazon WorkMail to provision their mailbox. For more information, see Adding a user.

Step 2: Migrate to Amazon WorkMail

You can work with any AWS migration partners to migrate to Amazon WorkMail. For information about these providers, see Amazon WorkMail features.

To migrate your mailboxes, create a dedicated Amazon WorkMail user to act as a migration administrator. The following procedure grants permission to that user to access all of the mailboxes in your organization.

To create a migration administrator

1. Do one of the following:

   - In the Amazon WorkMail console, create a new user to act as migration administrator. For more information, see Adding a user.

- In your Active Directory, create a new user to act as migration administrator, and then enable the user for Amazon WorkMail. For more information, see Enabling users.

2. In the Amazon WorkMail console navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Organization settings**, choose **Migration**, and then **Edit**.
4. Move the **Migration enabled** slider to the on position.
5. Open the **Migration administrator** and select a user.
6. Choose **Save**.

Step 3: Complete the migration to Amazon WorkMail

After you migrate your email accounts to Amazon WorkMail, you can verify your DNS records and configure your desktop and mobile clients.

To complete migration to Amazon WorkMail

1. Verify that all DNS records are updated and that they point to Amazon WorkMail. For more information about the required DNS records, see Adding a domain.

   > **Note**
   >
   > The DNS record update process can take several hours. If any new items appear in a source mailbox while the MX records are being changed, run the migration tool again to migrate new items after the DNS records are updated.

2. For more information about configuring your desktop or mobile clients to use Amazon WorkMail, see Connect Microsoft Outlook to your Amazon WorkMail account in the *Amazon WorkMail User Guide*.

Interoperability between Amazon WorkMail and Microsoft Exchange

Interoperability between Amazon WorkMail and Microsoft Exchange Server allows you to minimize disruption to your users as you migrate mailboxes to Amazon WorkMail, or use Amazon WorkMail for a subset of your corporate mailboxes.

This interoperability allows you to use the same corporate domain for mailboxes across both environments. This way, your users can schedule meetings with bidirectional sharing of calendar free/busy status information.

Prerequisites

Before you enable interoperability with Microsoft Exchange, do the following:

- Make sure you have at least one user enabled for Amazon WorkMail This is required to configure availability settings for Microsoft Exchange. To enable a user, follow the steps in Enable email routing for a user.
- Set up an Active Directory (AD) Connector. Setting up an AD Connector with your on-premises directory allows users to continue using their existing corporate credentials. For more information, see Create an AD Connector and Integrate Amazon WorkMail with your on-premises directory.
- Set up your Amazon WorkMail organization. Create an Amazon WorkMail organization that uses the AD Connector that you set up.
- Add your corporate domains to your Amazon WorkMail organization and then verify them in the Amazon WorkMail console. Otherwise, emails sent to this alias will bounce. For more information, see Working with domains.
- Migrate mailboxes to Amazon WorkMail.Enable users to provision and migrate mailboxes from your on-premises environment to Amazon WorkMail. For more information, see Enable existing users and see Migrating to Amazon WorkMail.

  > **Note**
  >
  > Do not update DNS records to point to Amazon WorkMail. This ensures that Microsoft Exchange remains the primary server for incoming email for as long as you want interoperability between the two environments.

- Make sure that the User Principal Names (UPNs) in Active Directory match the users' primary SMTP addresses.

Amazon WorkMail makes HTTPS requests to the Exchange Web Services (EWS) URL on Microsoft Exchange to obtain calendar free/busy information.

For EWS-based availability providers, Amazon WorkMail makes HTTPS requests to the Exchange Web Services (EWS) URL on Microsoft Exchange to obtain calendar free/busy information. Hence, the following prerequisites only apply to EWS-based availability providers.

- Ensure that the relevant firewall settings are set up to allow access from the internet. The default port for HTTPS requests is port 443.
- Amazon WorkMail can only make successful HTTPS requests to the EWS URL on Microsoft Exchange when a certificate signed by a valid certificate authority (CA) is available in your Microsoft Exchange environment. For more information, see Create an Exchange Server certificate request for a certification authority on the Microsoft Exchange Documentation website.
- You must enable **Basic Authentication** for EWS in Microsoft Exchange. For more information, see Virtual directories: Exchange 2013 on the Microsoft MVP Award Program Blog.

Add domains and enable mailboxes

Add your corporate domains to Amazon WorkMail so that they can be used in email addresses. Ensure that the domains added to Amazon WorkMail are verified, and then enable users and groups to provision mailboxes on Amazon WorkMail. Resources can't be enabled in Amazon WorkMail while in interoperability mode, and should be re-created in Amazon WorkMail after you disable interoperability mode. However, you can still use them to schedule meetings while in interoperability mode. Resources from Microsoft Exchange are always shown in the **Users** tab in Amazon WorkMail.

- For more information, see Add domains, Enable existing users, and Enable an existing group.

> **Note**
>
> To ensure interoperability with Microsoft Exchange, don't update the DNS records to point to Amazon WorkMail records. Microsoft Exchange remains the primary server for incoming email as long as you want interoperability between the two environments.

Enable interoperability

If you have not created an Amazon WorkMail organization, you can use the public API to create a new WorkMail Organization with interoperability mode enabled.

If you already have an Amazon WorkMail organization with an AD Connector linked to Active Directory, and you also have Microsoft Exchange, contact AWS Support for assistance with enabling Microsoft Exchange interoperability for an existing Amazon WorkMail organization.

Create service accounts in Microsoft Exchange and Amazon WorkMail

> **Note**
>
> Creating a service account in Exchange is not required when Exchange is not used as a back-end for custom availability provider.

To access calendar free/busy information, create a service account on both Microsoft Exchange and Amazon WorkMail. The Microsoft Exchange service account is any user on Microsoft Exchange that has access to the calendar free/busy information of other Exchange users. Access is granted by default; so no special permissions are required.

Similarly, the Amazon WorkMail service account is any user on Amazon WorkMail that has access to calendar free/busy information of other Amazon WorkMail users. This is also granted by default. You must create the Amazon WorkMail user in your on-premises directory, and then enable that user for Amazon WorkMail, to integrate Amazon WorkMail with AD Connector into your directory.

Limitations in interoperability mode

When your organization is in interoperability mode, you must use the Exchange admin center to manage all users, groups, and resources. To enable Amazon WorkMail users and groups, use the AWS Management Console. For more information, see Enable existing users and Enable an existing group.

When enabling a user or group for Amazon WorkMail, you can't edit the email addresses or aliases of those users and groups. Those must also be configured via the Exchange admincenter. Amazon WorkMail synchronizes changes in your directory every four hours.

Resources can't be created or enabled in Amazon WorkMail while in interoperability mode. However, all of your Exchange resources are available in the Amazon WorkMail address book and can be used for scheduling meetings as usual.

Configure availability settings on Amazon WorkMail

Configure availability settings on Amazon WorkMail to enable querying external systems, offering calendaring functionality, and to get calendar free/busy information. Amazon WorkMail supports two modes of obtaining free/busy information from a remote system:

- **Exchange Web Services (EWS)** — In this configuration, Amazon WorkMail will query an Exchange server or another WorkMail organization for availability information using the EWS protocol. This is the simplest configuration but requires the Exchange server's EWS endpoint to be accessible through the public internet.
- **Custom Availability Provider (CAP)** — In this configuration, an administrator can configure an AWS Lambda function to obtain user availability information for a given email domain. Depending on your email server platform, using CAP with Amazon WorkMail offers the following benefits:
  - Get user availability from internal EWS without the need to open up their firewall for WorkMail.
  - Get user availability from non-Exchange or non-EWS systems, like Google Workspace (formerly known as G Suite).

Topics

- Configure an EWS-based availability provider
- Configuring a Custom Availability Provider
- Building a Custom Availability Provider Lambda function

Configure an EWS-based availability provider

To configure an EWS-based availability settings on the console, complete the following procedure:

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. To do so, open the **Select a region** list, located to the right of the search box, then choose the desired Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, and then choose the name of an organization.
3. In the navigation pane, choose **Organization settings**, and then choose the **Interoperability** tab.
4. Choose **Add availability configuration**, and then enter the following information:
   - **Type** — Select EWS.
   - **Domain** — The domain for which WorkMail will attempt to query availability information using this configuration.
   - **EWS URL** — Amazon WorkMail will query this URL to the EWS endpoint. See the Getting the EWS URL section of this guide.
   - **User email address** — The email address of the user that WorkMail will use to authenticate with to the EWS endpoint.
   - **Password** — The password that WorkMail will use to authenticate with to the EWS endpoint.
5. Choose **Save**.

Getting the EWS URL

To get the EWS URL for Exchange using Microsoft Outlook, complete the following procedure:

1. Log in to Microsoft Outlook on Windows for any user on your Exchange environment.
2. Hold the **Ctrl** key and open the context (right-click) menu on the Microsoft Outlook icon in the task bar.
3. Choose **Test E-mail AutoConfiguration**.
4. Enter the Microsoft Exchange user's email address and password, and choose **Test**.
5. From the Results window, copy the value for the **Availability Service URL**.

To get the EWS URL for exchange using PowerShell, at the PowerShell prompt, execute the following command:

```
Get-WebServicesVirtualDirectory |Select name, *url* | fl
```

To get the EWS URL for Amazon WorkMail, first, find the EWS domain under Amazon WorkMail endpoints and quotas. Enter the EWS URL — `https://"EWS domain"/EWS/Exchange.asmx` and replace "EWS domain" with your EWS domain.

Configuring a Custom Availability Provider

To configure a Custom Availability Provider (CAP), complete the following procedure:

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. To do so, open the **Select a Region** list, located to the right of the search box, then choose the desired Region.
2. In the navigation pane, choose **Organizations**, and then choose the name of an organization.
3. In the navigation panel, choose **Organization settings** and then choose **Interoperability**.
4. Choose **Add availability configuration**, and then enter the following information:

   - **Type** — Select **CAP Lambda**.
   - **Domain** — The domain for which WorkMail will attempt to query availability information using this configuration.
   - **ARN** — The ARN of the Lambda function that will provide the availability information.

To build a CAP Lambda function, see Building a Custom Availability Provider Lambda function.

Building a Custom Availability Provider Lambda function

Custom Availability Providers (CAPs) are configured with a JSON-based request and response protocol that is written in a well defined JSON schema. A Lambda function will parse the request and provide a valid response.

Topics

- Request and response elements
- Granting access
- Example of Amazon WorkMail using a CAP Lambda function

Request and response elements

Request elements

The following is a sample request used to configure a CAP for an Amazon WorkMail user:

```
{
    "requester": {
        "email": "user1@internal.example.com",
        "userName": "user1",
        "organization": "m-0123456789abcdef0123456789abcdef",
        "userId": "S-1-5-18",
        "origin": "127.0.0.1"
    },
    "mailboxes": [
        "user2@external.example.com",
        "unknown@internal.example.com"
    ],
    "window": {
        "startDate": "2021-05-04T00:00:00.000Z",
        "endDate": "2021-05-06T00:00:00.000Z"
    }
}
```

A request is composed of three sections: **requester**, **mailboxes**, and **window**. These are described in the following Requester, Mailboxes, and Window sections of this guide.

Requester

The *requester* section provides information about the user who made the original request to Amazon WorkMail. CAPs use this information to change the behavior of the provider. For instance, this data can be used to impersonate the same user on the backend availability provider or certain details can be omitted from the response.

Field	Description	Required
`Email`	The primary email address of the requester.	Yes
`Username`	The user name of the requester.	Yes

Field	Description	Required
`Organization`	The organization ID of the requester.	Yes
`UserID`	The requester ID.	Yes
`Origin`	The remote address of the request.	No
`Bearer`	Reserved for future use.	No

Mailboxes

The *mailboxes* section contains a comma separated list of email addresses of users for which availability information is requested.

Window

The *window* section contains the time window which the availability information is requested for. Both `startDate` and `endDate` are specified in UTC and are formatted according to RFC 3339. Events aren't expected to be truncated. In other words, if an event starts before the defined `StartDate`, the original start will be used.

Response elements

Amazon WorkMail will wait for 25 seconds to get a response from the CAP Lambda function. After 25 seconds, Amazon WorkMail will assume the function has failed and generate failures for the associated mailboxes in the EWS GetUserAvailability response. This will not cause the entire GetUserAvailability operation to fail.

The following is a sample response from the configuration defined at the beginning of this section:

```
{
    "mailboxes": [{
        "mailbox": "user2@external.example.com",
        "events": [{
            "startTime": "2021-05-03T23:00:00.000Z",
            "endTime": "2021-05-04T03:00:00.000Z",
            "busyType": "BUSY"|"FREE"|"TENTATIVE",
            "details": {  // optional
```

```
                "subject": "Late meeting",
                "location": "Chime",
                "instanceType": "SINGLE_INSTANCE"|"RECURRING_INSTANCE"|"EXCEPTION",
                "isMeeting": true,
                "isReminderSet": true,
                "isPrivate": false
            }
        }],
        "workingHours": {
            "timezone": {
                "name": "W. Europe Standard Time"
                "bias": 60,
                "standardTime": {  // optional (not needed for fixed offsets)
                    "offset": 60,
                    "time": "02:00:00",
                    "month":
 "JAN"|"FEB"|"MAR"|"APR"|"JUN"|"JUL"|"AUG"|"SEP"|"OCT"|"NOV"|"DEC",
                    "week": "FIRST"|"SECOND"|"THIRD"|"FOURTH"|"LAST",
                    "dayOfWeek": "SUN"|"MON"|"TUE"|"WED"|"THU"|"FRI"|"SAT"
                },
                "daylightTime": {  // optional (not needed for fixed offsets)
                    "offset": 0,
                    "time": "03:00:00",
                    "month":
 "JAN"|"FEB"|"MAR"|"APR"|"JUN"|"JUL"|"AUG"|"SEP"|"OCT"|"NOV"|"DEC",
                    "week": "FIRST"|"SECOND"|"THIRD"|"FOURTH"|"LAST",
                    "dayOfWeek": "SUN"|"MON"|"TUE"|"WED"|"THU"|"FRI"|"SAT"
                },
            },
            "workingPeriods":[{
                "startMinutes": 480,
                "endMinutes": 1040,
                "days": ["SUN"|"MON"|"TUE"|"WED"|"THU"|"FRI"|"SAT"]
            }]
        }
    },{
        "mailbox": "unknown@internal.example.com",
        "error": "MailboxNotFound"
    }]
}
```

A response is composed of a single *mailboxes* section which consists of a list of mailboxes. Each mailbox for which availability is successfully obtained is composed of three sections: *mailbox*,

events, and *workinghours*. If the availability provider has failed to obtain availability information for a mailbox, the section is composed of two sections: *mailbox* and *error*. These are described in the following Mailbox, Events, Working Hours, Timezone, Working Periods, and Error sections of this guide.

Mailbox

The *mailbox* section is the email address of the user found in the *mailboxes* section of the request.

Events

The *events* section is a list of events that occur in the requested window. Each event is defined with the following parameters:

Field	Description	Required
`startTime`	The start time of the event in UTC and formatted according to RFC 3339.	Yes
`endTime`	The end time of the event in UTC and formatted according to RFC 3339.	Yes
`busyType`	The busy type of the event. Can be `Busy`, `Free`, or `Tentative` .	Yes
`details`	The details of the event.	No
`details.subject`	The subject of the event.	Yes
`details.location`	The location of the event.	Yes
`details.instanceType`	The instance type of the event. Can be `Single_Instance` , `Recurring_Instance` , or `Exception` .	Yes

Field	Description	Required
`details.isMeeting`	A Boolean to indicate if the event has attendees.	Yes
`details.isReminderSet`	A Boolean to indicate if the event has a reminder set.	Yes
`details.isPrivate`	A Boolean to indicate if the event is set to private.	Yes

Working Hours

The *workingHours* section contains information about the the mailbox owner's working hours. It contains two sections: *timezone* and *workingPeriods*.

Timezone

The *timezone* subsection describes the mailbox owner's time zone. It's important to correctly render the user's working hours when the requester works in a different time zone. The availability provider is required to explicitly describe the time zone, rather than using a name. Using the standarized time zone description helps avoid time zone mismatches.

Field	Description	Required
`name`	The time zone's name.	Yes
`bias`	The default offset from GMT in minutes.	Yes
`standardTime`	The start of standard time for the specified time zone.	No
`daylightTime`	The start of daylight savings time for the specified time zone.	No

You must either define both `standardTime` and `daylightTime`, or omit both. Fields in the `standardTime` and `daylightTime` object are:

Field	Description	Allowed Values
`offset`	The offset relative to the default offset in minutes.	NA
`time`	The time at which the transition between standard time and daylight savings time happens, specified as `hh:mm:ss`.	NA
`month`	The month that the transitio n between standard time and daylight savings time happens.	`JAN`,`FEB`, `MAR`, `APR`, `JUN`, `JUL`, `AUG`, `SEP`, `OCT`, `NOV`, `DEC`
`week`	The week within the specified month that the transitio n between standard time and daylight savings time happens.	`FIRST`, `SECOND`, `THIRD`, `FOURTH`, `LAST`
`dayOfWeek`	The day within the specified week that the transition between standard time and daylight savings time happens.	`SUN`, `MON`, `TUE`, `WED`, `THU`, `FRI`, `SAT`

Working Periods

The *workingPeriods* section contains one or more working period objects. Each period defines a start and end of working day for one or more days.

Field	Description	Allowed Values
`startMinutes`	The start of the working day in minutes from midnight.	NA
`endMinutes`	The end of the working day in minutes from midnight.	NA
`days`	The days that this period applies to.	`SUN`, `MON`, `TUE`, `WED`, `THU`, `FRI`, `SAT`

Error

The *error* field can contain arbitrary error messages. The following table lists a mapping of well known codes to EWS error codes. All other messages will be mapped to `ERROR_FREE_BUSY_GENERATION_FAILED`.

Value	EWS Error code
`MailboxNotFound`	`ERROR_MAIL_RECIPIENT_NOT_FOUND`
`ErrorAvailabilityConfigNotFound`	`ERROR_AVAILABILITY_CONFIG_NOT_FOUND`
`ErrorServerBusy`	`ERROR_SERVER_BUSY`
`ErrorTimeoutExpired`	`ERROR_TIMEOUT_EXPIRED`
`ErrorFreeBusyGenerationFailed`	`ERROR_FREE_BUSY_GENERATION_FAILED`
`ErrorResponseSchemaValidation`	`ERROR_RESPONSE_SCHEMA_VALIDATION`

Granting access

Run the following Lambda command from the AWS Command Line Interface (AWS CLI). This command adds a resource policy to the Lambda function that parses the CAP. This function allows the Amazon WorkMail availability service to invoke your Lambda function.

```
aws lambda add-permission \
    --region LAMBDA_REGION \
    --function-name CAP_FUNCTION_NAME \
    --statement-id AllowWorkMail \
    --action "lambda:InvokeFunction" \
    --principal availability.workmail.WM_REGION.amazonaws.com \
    --source-account WM_ACCOUNT_ID \
    --source-arn arn:aws:workmail:WM_REGION:WM_ACCOUNT_ID:organization/ORGANIZATION_ID
```

In the command, add the following parameters where indicated:

- *LAMBDA_REGION* — Name of the region where the CAP Lambda is deployed. For example, `us-east-1`.
- *CAP_FUNCTION_NAME* — Name of the CAP Lambda function.

> **Note**
>
> This can be the name, alias, or either partial or full ARN of the CAP Lambda function.

- *WM_REGION* — Name of the region where the Amazon WorkMail organization invokes the Lambda function.

> **Note**
>
> Only the following Regions are available for use with CAP:
>
> - US East (N. Virginia)
> - US West (Oregon)
> - Europe (Ireland)

- *WM_ACCOUNT_ID* — The ID of the Organization account.
- *ORGANIZATION_ID* — The ID of the Organization that invokes the CAP Lambda. For example, Org ID: m-934ebb9eb57145d0a6cab566ca81a21f.

> **Note**
>
> *LAMBDA_REGION* and *WM_REGION* will be different only if cross-Region calls are necessary. If cross-Region calls are not necessary, they will be the same.

Example of Amazon WorkMail using a CAP Lambda function

For an example of Amazon WorkMail using a CAP Lambda function to query an EWS endpoint, see this AWS sample application on the *Serverless applications for Amazon WorkMail GitHub repository*.

Configure availability settings in Microsoft Exchange

To redirect all calendar free/busy information requests for enabled users to Amazon WorkMail, set up an availability address space in Microsoft Exchange.

Use the following PowerShell command to create the address space:

```
$credentials = Get-Credential
```

At the prompt, enter the credentials of the Amazon WorkMail service account. The username should be entered as **domain\username** (that is, ***orgname*.awsapps.com\workmail_service_account_*username***. Here, ***orgname*** represents the name of the Amazon WorkMail organization. For more information, see Create service accounts in Microsoft Exchange and Amazon WorkMail.

```
Add-AvailabilityAddressSpace -ForestName orgname.awsapps.com -AccessMethod OrgWideFB -
Credentials $credentials
```

For more information, see Add-AvailabilityAddressSpace on Microsoft Docs.

Enable email routing between Microsoft Exchange and Amazon WorkMail users

With email routing between Microsoft Exchange Server and Amazon WorkMail, users can keep their existing email addresses after they migrate to Amazon WorkMail. Email routing lets you keep

Microsoft Exchange Server as the primary Simple Mail Transfer Protocol (SMTP) server for incoming email for your organization.

Before using email routing, you'll need to complete the following prerequisites:

- Enable interoperability mode for your organization. For more information, see Enable interoperability.
- Ensure that you see your domain in the Amazon WorkMail console.
- Verify that our Microsoft Exchange Server can send email to the internet. You might need to configure a Send connector. For more information about Send connectors, see Create a Send connector in Exchange Server to send mail to the internet in the Microsoft documentation.

Enable email routing for a user

We recommend that you complete the following steps first for test users before applying any changes to your organization.

1. Enable the user account that you are migrating to Amazon WorkMail. For more information, see Enable existing users.
2. In the Amazon WorkMail console, ensure that there are at least two email addresses associated with the enabled user.
   - <*workmailuser*@*orgname*.awsapps.com> (this is added automatically and can be used for tests without your Microsoft Exchange.)
   - <*workmailuser*@*yourdomain*.com> (this is added automatically and is the primary Microsoft Exchange address.)

   For more information, see Edit user email addresses.
3. Ensure that you migrate all data from the mailbox in Microsoft Exchange to the mailbox in Amazon WorkMail. For more information, see Migrating to Amazon WorkMail.
4. After all of the data is migrated, disable the mailbox for the user on Microsoft Exchange. Then, create a mail user (or mail-enabled user) that has the external SMTP address pointed to Amazon WorkMail. To do this, use the following commands in the Exchange Management Shell:

> **Important**
>
> The following steps erase the contents of the mailbox. Ensure that your data has been migrated to Amazon WorkMail before you attempt to enable email routing. Some mail

clients don't seamlessly switch to Amazon WorkMail when you run this command. For more information, see Mail client configuration.

```
$old_mailbox = Get-Mailbox exchangeuser
```

```
Disable-Mailbox $old_mailbox
```

```
$new_mailuser = Enable-MailUser $old_mailbox.Identity -
ExternalEmailAddress workmailuser@orgname.awsapps.com -PrimarySmtpAddress
 $old_mailbox.PrimarySmtpAddress
```

```
Set-MailUser $new_mailuser -EmailAddresses $old_mailbox.EmailAddresses -
HiddenFromAddressListsEnabled $old_mailbox.HiddenFromAddressListsEnabled
```

In the above commands, *orgname* represents the name of your Amazon WorkMail organization. For more information, see Disabling mailbox and Enabling mail users on Microsoft TechNet.

5. Send a test email to the user (in the example above, **workmailuser@yourdomain.com**). If email routing has been enabled correctly, the user should be able to log in to their Amazon WorkMail mailbox and receive the email.

Note

Microsoft Exchange remains the primary server for incoming email as long as you would like to have interoperability between the two environments. To ensure interoperability with Microsoft Exchange, the DNS records shouldn't be updated to point to Amazon WorkMail until later.

Post setup configuration

The above steps move a user mailbox from Microsoft Exchange Server to Amazon WorkMail, while keeping the user in Microsoft Exchange as a contact. Because the migrated user is now an external mail user, Microsoft Exchange Server imposes additional constraints. There may also be additional configuration requirements to complete the migration.

- The user might not be able to send emails to groups by default. To enable this functionality, you must add the user to a safe sender list for all groups. For more information, see Delivery management on Microsoft TechNet.
- The user might not be able to book resources. To enable this functionality, you must set the `ProcessExternalMeetingMessages` of all of the resources that the user needs to access. For more information, see Set-CalendarProcessing on Microsoft TechNet.

Mail client configuration

Some mail clients don't switch seamlessly to Amazon WorkMail. These clients require the user to perform additional setup steps. Different mail clients require different actions to be taken.

- Microsoft Outlook on Windows – Requires Outlook to be restarted. At startup, you are required to choose whether to keep using the old mailbox or use a temporary mailbox. Choose the temporary mailbox option. Then, reconfigure the Microsoft Exchange mailbox.
- Microsoft Outlook on MacOS – When Outlook is restarted, it will prompt the following message: **Outlook was redirected to server *`orgname`*.awsapps.com. Do you want this server to configure your settings?** Accept the suggestion.
- Mail on iOS – The mail app stops receiving emails and generates a **can't get mail** error. Recreate and reconfigure the Microsoft Exchange mailbox.

Disabling interoperability mode and decommissioning your mail server

After you configure your Microsoft Exchange mailboxes for Amazon WorkMail, you can disable interoperability mode. If you haven't migrated any users or records, disabling interoperability mode does not affect any of your configurations.

> **Warning**
>
> Before disabling interoperability mode, ensure that you complete all the required steps. Failure to do so can result in bounced emails or unintended behavior. If you have not completed migration, disabling interoperability may cause disruptions to your organization. You can't undo this operation.

To disable interoperability mode support

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the organization for which you want to disable interoperability mode.
3. Under **Organization settings**, choose **Disable interoperability mode**.
4. In the **Disable interoperability mode** dialog box, enter the name of the organization and choose **Disable interoperability mode**.

After disabling interoperability support, users and groups that are not enabled for Amazon WorkMail are removed from the address book. You can still enable any missing user or group using the Amazon WorkMail console, and they are added to the address book. Resources from Microsoft Exchange can't be enabled and do not appear in the address book until you complete the step below.

- **Create resources in Amazon WorkMail** – You can create resources in Amazon WorkMail and then configure delegates and booking options for these resources. For more information, see Working with resources.
- **Create an AutoDiscover DNS record** – Configure an AutoDiscover DNS record for all mail domains in the organization. This enables users to connect to their Amazon WorkMail mailboxes from their Microsoft Outlook and mobile clients. For more information, see Use AutoDiscover to configure endpoints.
- **Switch your MX DNS record to Amazon WorkMail** – To deliver all incoming emails to Amazon WorkMail, you must switch your MX DNS record to Amazon WorkMail. Changes to DNS records can take up to 72 hours to propagate to all DNS servers.
- **Decommission your mail server** – After you've verified that all email is being routed directly to Amazon WorkMail, you can decommission your mail server if you don't intend to use it going forward.

Troubleshooting

Solutions to the most commonly encountered Amazon WorkMail interoperability and migration errors are listed below.

Exchange Web Services (EWS) URL is invalid or unreachable – Check that you have the correct EWS URL. For more information, see Configure availability settings on Amazon WorkMail.

Connection failure during EWS validation – This is a general error and can be caused by:

- No internet connection in Microsoft Exchange.
- Your firewall is not configured to allow access from the internet. Ensure that port 443 (the default port for HTTPS requests) is open.

If you've confirmed the internet connection and firewall settings, but the error persists, contact AWS Support.

Invalid username and password when configuring Microsoft Exchange interoperability – This is a general error and can be caused by:

- The username is not in the expected form. Use the following pattern:

```
DOMAIN\username
```

- Your Microsoft Exchange server is not configured for Basic Authentication for EWS. For more information, see Virtual directories: Exchange 2013 on the Microsoft MVP Award Program Blog.

User receives emails with winmail.dat attachment – This might happen when encrypted S/MIME email is sent from Exchange to Amazon WorkMail and received in Outlook 2016 for Mac or an IMAP client. The solution is to run the following command in the Exchange Management Shell.

Set-RemoteDomain -Identity "Default" -TNEFEnabled $false

If you've confirmed the points above but the error persists, contact AWS Support.

Amazon WorkMail quotas

Amazon WorkMail can be used by both enterprise customers and small business owners. Although we support most use cases without the need to configure any changes in quotas, we also protect

our users and the internet against abuse of the product. Therefore, some customers may run into quotas that we have set. This section describes these quotas and how to change them.

Some quota values can be changed, and some are hard quotas that can't be changed. For more information about requesting a quota increase, see AWS Service quotas in the *Amazon Web Services General Reference*.

Amazon WorkMail organization and user quotas

You can add up to 25 users to your Amazon WorkMail organization for a 30-day free trial. After this period ends, you are charged for all active users unless you remove them or close your Amazon WorkMail account.

All messages that are sent to another user are considered when evaluating these quotas. These include emails, meeting requests, meeting responses, task requests, and messages that are forwarded or redirected automatically as the result of a rule.

> **Note**
>
> When requesting a quota increase for a specific organization, you must include the organization name in your request.

Resource	Default quota	Upper bound for change requests
Amazon WorkMail organizations per AWS account	100	Can be increased based on an organization's directory type. You can view AWS Directory Service quotas and request increases from the AWS Directory Service console. For more information, see Service quotas in the *AWS General Reference*.
Users per Amazon WorkMail organization	1,000	Can be increased depending on the organization's directory type, as follows:

Resource	Default quota	Upper bound for change requests
		• Amazon WorkMail directory : up to 10 million users • Simple AD or AD Connector , large: up to 5,000 users* • Simple AD or AD Connector , small: up to 500 users* • Microsoft AD, hosted by AWS Directory Service: up to 10 million users depending on your setup and configuration, *If you are using Simple AD or AD Connector, see AWS Directory Service for additional information.
Free trial users	Up to 25 users in the first 30 days	The free trial period is only applicable for the first 25 users in any organization. Any additional users are not included in the free trial offer.
Recipients addressed per AWS account per day	100,000 recipients external to the organization, with no hard quota on recipients internal to the organization	There is no upper bound. However, Amazon WorkMail is a business email service and not intended to be used for bulk email services. For bulk email services, see Amazon SES or Amazon Pinpoint.

Resource	Default quota	Upper bound for change requests
Recipients addressed per AWS account per day using any of the test domains	200 recipients, regardless of destination	The test mail domain is not intended for long-term usage. We recommend that you add your own domain and use it as the default domain.

Quotas for groups are set by the underlying directory.

WorkMail organization setting quotas

Resource	Default quota
Number of domains per Amazon WorkMail organization	1,000 This is a hard quota and can't be changed.
Number of sender patterns in email flow rules per rule	250 This is a hard quota and can't be changed.
Number of sender patterns in email flow rules per organization	1,000 This is a hard quota and can't be changed.

Per-user quotas

All messages that are sent to another user are considered when evaluating these quotas. These include emails, meeting requests, meeting responses, task requests, and messages that are forwarded or redirected automatically as the result of a rule.

Resource	Default quota	Upper quota for change requests
Maximum size of mailbox	50 GB	Not applicable

Resource	Default quota	Upper quota for change requests
	This is a hard quota and can't be changed.	
Maximum number of aliases per user	100 This is a hard quota and can't be changed.	Not applicable
Recipients addressed per user per day using the domain that you own	10,000 recipients external to the organization, with no hard quota on recipients internal to the organization.	There is no upper bound. However, Amazon WorkMail is a business email service and not intended to be used for bulk email services. For bulk email services, see Amazon SES or Amazon Pinpoint.

Message quotas

All messages that are sent to another user are considered when evaluating these quotas. These include emails, meeting requests, meeting responses, task requests, and messages that are forwarded or redirected automatically as the result of a rule.

Resource	Default quota
Maximum size of incoming message	29 MB of unencoded data. Messages are received in a MIME format. Maximum size of incoming MIME message is 40 MB. This is a hard quota and can't be changed.
Maximum size of outgoing message	29 MB of unencoded data.

Resource	Default quota
	Messages are sent in a MIME format. Maximum size of outgoing MIME message is 40 MB. This is a hard quota and can't be changed.
Maximum number of recipients per message	500 This is a hard quota and can't be changed.
Maximum number of attachments per message	500 This is a hard quota and can't be changed.

Working with organizations

In Amazon WorkMail, your organization represents the users in your company. In the Amazon WorkMail console, you see a list of your available organizations. If you don't have any available, you must create an organization in order to use Amazon WorkMail.

Topics

- Creating an organization
- Deleting an organization
- Finding an email address
- Working with organization settings
- Tagging an organization
- Working with access control rules
- Setting mailbox retention policies

Creating an organization

To use Amazon WorkMail, you must first create an organization. One AWS account can have multiple Amazon WorkMail organizations. When you create an organization, you also select a domain for the organization and set up user directory and encryption settings.

You can create a new user directory, or integrate Amazon WorkMail with an existing directory. You can use Amazon WorkMail with an on-premises Microsoft Active Directory, AWS Managed Active Directory, or Simple AD. By integrating with your on-premises directory, you can use your existing users and groups in Amazon WorkMail and users can sign in with their existing credentials. If you're using an on-premises directory, you must first set up an AD Connector in AWS Directory Service. The AD Connector synchronizes your users and groups with the Amazon WorkMail address book and performs user authentication requests. For more information, see Active Directory Connector in the *AWS Directory Service Administration Guide*.

You also have the option of selecting a AWS KMS key that Amazon WorkMail uses to encrypt the mailbox content. You can either select the default AWS managed master key for Amazon WorkMail, or use an existing KMS key in AWS Key Management Service (AWS KMS). For information about creating a new KMS key, see Creating keys in the *AWS Key Management Service Developer Guide*. If you are signed in as an AWS Identity and Access Management (IAM) user, make yourself a key

administrator on the KMS key. For more information, see Enabling and disabling keys in the *AWS Key Management Service Developer Guide*.

Considerations

Remember the following when creating an Amazon WorkMail organization:

- Amazon WorkMail doesn't currently support managed Microsoft Active Directory services that you share with multiple accounts.
- If you have an on-premises Active Directory with Microsoft Exchange and an AD Connector, we recommend configuring interoperability settings for your organization. This allows you to minimize disruption to your users as you migrate mailboxes to Amazon WorkMail, or use Amazon WorkMail for a subset of your corporate mailboxes. For more information, see Interoperability between Amazon WorkMail and Microsoft Exchange.
- If you select the **Free test domain** option, you can start using your Amazon WorkMail organization with the provided test domain. The test domain uses this format: *example*.awsapps.com. You can use the test mail domain with Amazon WorkMail and other supported AWS services as long as you maintain enabled users in your Amazon WorkMail organization. However, you can't use the test domain for other purposes. The test domain might become available for registration and use by other customers if your Amazon WorkMail organization does not maintain at least one enabled user.
- Amazon WorkMail does not support multi-Region directories.

Topics

- Creating an organization
- Viewing an organization's details
- Integrating a WorkSpaces directory
- Organization states and descriptions

Creating an organization

Create a new organization in the Amazon WorkMail console.

To create an organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation bar, select **Organization**.

   The **Organizations** page appears and displays your organizations, if any.

3. Choose **Create organization**.
4. Under **Email domain**, select the domain to use for the email addresses in your organization:

   - **Existing Route 53 domain** – Select an existing domain that you manage with an Amazon Route 53 (Route 53) hosted zone.
   - **New Route 53 domain** – Register a new Route 53 domain name to use with Amazon WorkMail.
   - **External domain** – Enter an existing domain that you manage with an external domain name system (DNS) provider.
   - **Free test domain** – Use a free test domain provided by Amazon WorkMail. You can explore Amazon WorkMail using a test domain, and then add a domain to your organization later.

5. (Optional) If your domain is managed through Amazon Route 53, for **Route 53 hosted zone**, select your Route 53 domain.
6. For **Alias**, enter a unique alias for your organization.
7. Choose **Advanced settings**, and for **User directory**, select one of the following options:

   - **Create new Amazon WorkMail directory** – Creates a new directory for adding and managing your users.
   - **Use existing directory** – Uses an existing directory to manage your users, such as an on-premises Microsoft Active Directory, AWS Managed Active Directory, or Simple AD.

8. For **Encryption**, select one of the following options:

   - **Use an Amazon WorkMail managed key** – Creates a new encryption key in your account.
   - **Use existing KMS key** – Uses an existing KMS key that you have already created in AWS KMS.

9. Choose **Create organization**.

If you use an external domain, verify it by adding the appropriate text (TXT) and mail exchanger (MX) records to your DNS service. TXT records allow you to enter notes about the DNS service. MX records specify the incoming mail server.

Be sure to set your domain as the default for your organization. For more information, see Verifying domains and Choosing the default domain.

When your organization is **Active**, you can add users to it and set up their email clients. For more information, see Adding a user and Setting up email clients for Amazon WorkMail.

Viewing an organization's details

Each of your Amazon WorkMail organizations can display an the organization details page. The page shows you information about their organization, including IDs that you can use with the AWS Command Line Interface. Messages on the page can also show you any steps needed to finish setting up and organization, such as an unverified domain or a lack of users. The messages also provide the first step that you follow to set up a given email client.

To view organization details

1. In the navigation bar, choose **Organization**.

   The **Organizations** page appears and displays your organizations.
2. Choose the organization that you want to view.

Integrating a WorkSpaces directory

To use Amazon WorkMail with WorkSpaces, create a compatible directory by using the following steps.

To add a compatible WorkSpaces directory

1. Create a compatible directory using WorkSpaces. For WorkSpaces instructions, see Get started with Amazon WorkSpaces Quick Setup in the *Amazon WorkSpaces Administration Guide*.
2. In the Amazon WorkMail console, create your Amazon WorkMail organization and choose to use your existing directory for it. For more information, see Creating an organization.

Organization states and descriptions

After you create an organization, it can have one of the following states.

State	Description
Active	Your organization is healthy and ready for use.
Creating	A workflow is running to create your organization.
Failed	Your organization could not be created.
Impaired	Your organization is malfunctioning or an issue has been detected.
Inactive	Your organization is inactive.
Requested	Your organization creation request is in the queue and waiting to be created.
Validating	All settings for the organization are being health-checked.

Deleting an organization

If you no longer want to use Amazon WorkMail for your organization's email, you can delete your organization from Amazon WorkMail.

> **Note**
>
> This operation can't be undone. You won't be able to recover your mailbox data after an organization is deleted.

To delete an organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. On the **Organizations** screen, in the list of organizations, select the organization to delete and choose **Delete**.
3. For **Delete organization**, choose whether to delete or keep the existing user directory, and then enter the name of the organization.
4. Choose **Delete organization**.

> **Note**
>
> If you didn't provide your own directory for Amazon WorkMail, we'll create one for you. If you keep this existing directory when you delete the organization, you will be charged for it unless it is being used by Amazon WorkMail, WorkDocs, or WorkSpaces. For pricing information, see Other directory types pricing.
>
> In order to delete the directory, it can't have any other AWS applications enabled. For more information, see Deleting a Simple AD directory or Deleting an AD Connector directory in the *AWS Directory Service Administration Guide*.

You may get an invalid Amazon Simple Email Service (Amazon SES) rule set error message when you attempt to delete an organization. If you receive this error, edit the Amazon SES rule in the Amazon SES console and remove the invalid rule set. The rule that you edit should have your Amazon WorkMail organization ID in the rule name. For more information about editing Amazon SES rules, see Creating receipt rules in the *Amazon Simple Email Service Developer Guide*.

If you need to figure out which rule set is not valid, save the rule first. An error message appears for the rule set.

Finding an email address

You can find if an email address is used in your Organization by user, resource, or group.

To find an email address

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the name of an organization.
3. In the **Organization** page, choose **Find email address**.
4. Choose **Search**.

Working with organization settings

The following sections explain how to use the settings available for Amazon WorkMail organizations. Settings that you choose will apply to the entire organization.

Topics

- Enabling mailbox migration
- Enabling journaling
- Enabling interoperability
- Enabling SMTP gateways
- Managing email flows
- Enforcing DMARC policies on incoming email

Enabling mailbox migration

You enable mailbox migration when you want to transfer mailboxes from a source, such as Microsoft Exchange or G Suite Basic, to Amazon WorkMail. You enable migration as part of a larger migration process. For more information, including how-to steps, see Migrating to Amazon WorkMail in the *Getting started* section of this guide.

Enabling journaling

You enable journaling to record your email communication. When using journaling, you typically use integrated third-party archiving and eDiscovery tools. Journaling helps ensure that you meet compliance regulations for data storage, privacy protection, and information protection.

For more information, including how-to steps, see Using email journaling with Amazon WorkMail in the *Getting started* section of this guide.

Enabling interoperability

Interoperability allows you to migrate from Microsoft Exchange and to use Amazon WorkMail as a subset of your corporate mailboxes. For more information, including how-to steps, see Configure availability settings on Amazon WorkMail in the *Getting started* section of this guide.

Enabling SMTP gateways

You enable Simple Mail Transfer Protocol (SMTP) gateways for use with outbound email flow rules. Outbound email flow rules let you route email messages sent from your Amazon WorkMail organization through an SMTP gateway. For more information, see Outbound email rule actions.

> **Note**
>
> SMTP gateways configured for outbound email flow rules must support Transport Layer Security (TLS) v1.2 using certificates from major certificate authorities. Only basic authentication is supported.

To configure an SMTP gateway

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of an organization.
3. In the navigation pane, choose **Organization settings**.

   The **Organization settings** page appears and displays a set of tabs.
4. Choose the **SMTP gateways** tab, then choose **Create gateway**.
5. Enter the following:

   - **Gateway name** — Enter a unique name.
   - **Gateway address** — Enter the gateway's host name or IP address.
   - **Port number** — Enter the gateway's port number.
   - **User name** — Enter a user name.

- **Password** — Enter a strong password.

6. Choose **Create**.

   The SMTP gateway is available for use with outbound email flow rules.

When you configure an SMTP gateway for use with an outbound email flow rule, outbound messages attempt to match the rule with an SMTP gateway. The message that match the rule are routed to the corresponding SMTP gateway, which then handles the rest of the email delivery.

If Amazon WorkMail is unable to reach the SMTP gateway, the system bounces the email message back to the sender. If this occurs, follow the previous steps to correct the gateway settings.

Managing email flows

To help manage email, you can set up *email flow rules*. Email flow rules can take one or more actions on email messages based on their addresses or domains. You can use email flow rules on senders' and recipients' email addresses or domains.

When you create an email flow rule, you specify a *rule action* that applies to an email when a specified rule *pattern* is matched.

Topics

- Inbound email rule actions
- Outbound email rule actions
- Sender and recipient patterns
- Creating email flow rules
- Editing email flow rules
- Configuring AWS Lambda for Amazon WorkMail
- Managing access to the Amazon WorkMail Message Flow API
- Testing an email flow rule
- Removing an email flow rule

Inbound email rule actions

Inbound email flow rules help prevent undesirable email from reaching your users' mailboxes. Inbound email flow rules, also called rule actions, automatically apply to all email messages sent

to anyone inside of your Amazon WorkMail organization. This differs from email rules for individual mailboxes.

> **Note**
>
> Optionally, you can use rules with an AWS Lambda function to process incoming email before it is delivered to your users' mailboxes. For more information about using Lambda with Amazon WorkMail, see Configuring AWS Lambda for Amazon WorkMail. For more information about Lambda, see the *AWS Lambda Developer Guide*.

Inbound email flow rules, also called rule actions, automatically apply to all email messages sent to anyone inside of the Amazon WorkMail organization. This differs from email rules for individual mailboxes.

The following rule actions define how inbound email is handled. For each rule, you specify sender and recipient patterns together with one of the following actions.

Action	Description
Drop email	The email message is ignored. It is not delivered, and the sender is not notified of the non-delivery.
Send bounce response	The email message is not delivered, and the sender is notified of the non-delivery in a bounce message.
Deliver to junk folder	The email message is delivered to users' spam or junk folders, even if it is not originally identified as spam by the Amazon WorkMail spam detection system.
Default	The email message is delivered after being checked by the Amazon WorkMail spam detection system. Spam email is delivered to the junk folder. All other email messages are delivered to the inbox.

Action	Description
	Other email flow rules with a less specific sender pattern are ignored. To add exceptions to domain-based email flow rules, configure the Default action with a more specific sender pattern. For more information, see Sender and recipient patterns.
Never deliver to junk folder	The email message is always delivered to users' inboxes, even if it is identified as spam by the Amazon WorkMail spam detection system. **Important** By not using the default spam detection system, you could expose your users to high-risk content from the addresses that you specify.
Run AWS Lambda	Passes the email message to a Lambda function for processing before or while it is delivered to users' inboxes.

Note

Inbound email is first delivered to Amazon SES, and then to Amazon WorkMail. If Amazon SES blocks an incoming email message, then rule actions won't apply. For example, Amazon SES blocks an email message when a known virus is detected or because of explicit IP filtering rules. Specifying a rule action, such as **Default**, **Deliver to junk folder**, or **Never deliver to junk folder** has no effect.

Outbound email rule actions

You use outbound email flow rules to direct email messages via SMTP gateways, or to block senders from sending email messages to specified recipients. For more information about SMTP gateways, see Enabling SMTP gateways.

You can also use outbound email flow rules to pass the email message to an AWS Lambda function for processing after the email is sent. For more information about Lambda, see the *AWS Lambda Developer Guide*.

The following rule actions define how outbound email is handled. For each rule, you specify sender and recipient patterns together with one of the following actions.

Action	Description
Default	The email message is sent via the normal flow.
Drop email	The email message is dropped. It is not sent, and the sender is not notified.
Send bounce response	The email message is not sent, and the sender is notified with a message that the administrator blocked the email message.
Route to SMTP gateway	The email message is sent via a configured SMTP gateway.
Run Lambda	Passes the email message to a Lambda function for processing before or while the email message is sent.

Sender and recipient patterns

An email flow rule can apply to a specific email address, or all email addresses under a specific domain or set of domains. You define a pattern to determine the email addresses that a rule applies to.

Both sender and recipient patterns take one of the following forms:

- An *email address* matches a single email address; for example:

```
mailbox@example.com
```

- A *domain name* matches all email addresses under that domain; for example:

```
example.com
```

- A *wildcard domain* matches all email addresses under that domain and all of its subdomains. A wildcard appears only at the front of a domain; for example:

```
*.example.com
```

- A *star* matches any email addresses under any domain.

```
*
```

> **Note**
>
> The **+** symbol is not valid inside of sender or recipient patterns.

Multiple patterns can be specified for one rule. For more information, see Inbound email rule actions and Outbound email rule actions.

Inbound email flow rules are applied if either the `Sender` or `From` header in an inbound email message matches any patterns. If present, the `Sender` address is matched first. The `From` address is matched if there is no `Sender` header or if the `Sender` header doesn't match any rule. If there are multiple recipients for the email message that match different rules, each rule applies for the matched recipients.

Outbound email flow rules are applied if the recipient and either the `Sender` or `From` header in an outbound email message matches any patterns. If there are multiple recipients for the email message that match different rules, each rule applies for the matched recipients.

If multiple rules match, the action of the most specific rule is applied. An example is when a rule for a specific email address takes precedence over a rule for an entire domain. If multiple rules have the same specificity, the most restrictive action is applied. An example is when a **Drop** action takes precedence over a **Bounce** action. The order of precedence for actions is the same as the order in which they are listed in Inbound email rule actions and Outbound email rule actions.

> **Note**
>
> Take care when creating rules with overlapping sender patterns with **Drop** or **Bounce** actions. Unexpected precedence ordering could result in many inbound email messages not being delivered.

Creating email flow rules

Email flow rules apply rule actions to incoming and outgoing email messages. The actions apply when messages match a specified pattern. New email flow rules take effect immediately.

To create email flow rules

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of an organization.
3. In the navigation pane, choose **Organization settings**.

   The **Organization settings** page appears and displays a set of tabs. From this page, you can create inbound or outbound rules. The following steps explain how to create both types.

   To create inbound rules

   1. Choose the **Inbound rules** tab and then choose **Create**.
   2. In the **Rule name** box, enter a unique name.
   3. Under **Action**, open the list and select an action. Each item in the list contains a description, and some provide **Learn more** links.

      > **Note**
      >
      > If you choose the **Run Lambda** action, additional controls appear: For information about using those controls, see the next section, Configuring AWS Lambda for Amazon WorkMail.

4. Under **Sender domains or addresses**, enter the sender domains or addresses to which you want the rule to apply.
5. Under **Destination domains or addresses**, enter any combination of destination domains and email addresses.
6. Choose **Create**.

To create outbound rules

1. Choose the **Outbound rules** tab and choose **Create**.
2. In the **Rule name** box, enter a unique name.
3. Under **Action**, open the list and select an action. Each item in the list contains a description, and some provide **Learn more** links.

> **Note**
>
> If you choose the **Run Lambda** action, additional controls appear. For information about using those controls, see the next section, Configuring AWS Lambda for Amazon WorkMail.

4. Under **Sender domains or addresses**, enter any combination of valid sender domains and email addresses.
5. Under **Destination domains or addresses**, enter any combination of valid destinations domains and email addresses.
6. Choose **Create**.

You can test the new email flow rule that you created. For more information, see Testing an email flow rule.

Editing email flow rules

You edit email flow rules whenever you need to change one or more rule actions for email messages. The steps in this section apply to incoming and outgoing email messages.

To edit email flow rules

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the name of an organization.
3. In the navigation pane, choose **Organization settings**.

   The **Organization settings** page appears and displays a set of tabs.
4. Choose the **Inbound rules** or **Outbound rules** tabs.
5. Choose the radio button next to the rule that you want to change, then choose **Edit**.
6. Change the action or actions in the rule as needed, then choose **Save**.

Configuring AWS Lambda for Amazon WorkMail

Use the **Run Lambda** action in inbound and outbound email flow rules to pass email messages that match the rules to an AWS Lambda function for processing.

Choose from the following configurations for a **Run Lambda** action in Amazon WorkMail.

Synchronous Run Lambda configuration

Email messages that match the flow rule are passed to a Lambda function for processing before they are sent or delivered. Use this configuration to modify email content. You can also control inbound or outbound email flow for different use cases. For example, a rule passed to a Lambda function can block delivery of sensitive email messages, remove attachments, or add disclaimers.

Asynchronous Run Lambda configuration

Email messages that match the flow rule are passed to a Lambda function for processing while they are sent or delivered. This configuration does not affect email delivery and is used for tasks such as collecting metrics for inbound or outbound email messages.

Whether you choose a synchronous or asynchronous configuration, the event object passed to your Lambda function contains metadata for the inbound or outbound email event. You can also use the message ID in the metadata to access the full content of the email message. For more information, see Retrieving message content with AWS Lambda. For more information about email events, see Lambda event data.

For more information about inbound and outbound email flow rules, see Managing email flows. For more information about Lambda, see the *AWS Lambda Developer Guide*.

> **Note**
>
> Currently, Lambda email flow rules reference only Lambda functions in the same AWS Region and AWS account as the Amazon WorkMail organization being configured.

Getting started with AWS Lambda for Amazon WorkMail

To start using AWS Lambda with Amazon WorkMail, we recommend deploying the WorkMail Hello World Lambda function from the AWS Serverless Application Repository to your account. The function has all the necessary resources, and the permissions configured for you. For more examples, see the amazon-workmail-lambda-templates repository on GitHub.

If you choose to create your own Lambda function, you must configure permissions using the AWS Command Line Interface (AWS CLI). In the following example command, do the following:

- Replace `MY_FUNCTION_NAME` with the name of your Lambda function.
- Replace `REGION` with your Amazon WorkMail AWS Region. Available Amazon WorkMail Regions include `us-east-1` (US East (N. Virginia)), `us-west-2` (US West (Oregon)), and `eu-west-1` (Europe (Ireland)).
- Replace `AWS_ACCOUNT_ID` with your 12-digit AWS account ID.
- Replace `WORKMAIL_ORGANIZATION_ID` with your Amazon WorkMail organization ID. You can find it on the card for your organization on the **Organizations** page.

```
aws --region REGION lambda add-permission --function-name MY_FUNCTION_NAME
--statement-id AllowWorkMail
--action "lambda:InvokeFunction"
--principal workmail.REGION.amazonaws.com
--source-arn
 arn:aws:workmail:REGION:AWS_ACCOUNT_ID:organization/WORKMAIL_ORGANIZATION_ID
```

For more information about using the AWS CLI, see the *AWS Command Line Interface User Guide*.

Configuring synchronous Run Lambda rules

To configure a synchronous **Run Lambda** rule, create an email flow rule with the **Run Lambda** action and select the **Run synchronously** check box. For more information about creating mail flow rules, see Creating email flow rules.

To finish creating the synchronous rule, add the Lambda Amazon Resource Name (ARN) and configure the following options.

Fallback action

The action Amazon WorkMail applies if the Lambda function fails to run. This action also applies to any recipients that are omitted from the Lambda response if the **allRecipients** flag is not set. The **Fallback action** can't be another Lambda action.

Rule timeout (in minutes)

The time period during which the Lambda function is retried if Amazon WorkMail fails to invoke it. The **Fallback action** is applied at the end of this time period.

> **Note**
>
> Synchronous **Run Lambda** rules support the * destination condition only.

Lambda event data

The Lambda function is triggered using the following event data. The presentation of the data varies depending on which programming language is used for the Lambda function.

```
{
    "summaryVersion": "2018-10-10",
    "envelope": {
        "mailFrom" : {
            "address" : "from@example.com"
        },
        "recipients" : [
           { "address" : "recipient1@example.com" },
           { "address" : "recipient2@example.com" }
        ]
    },
```

```
    "sender" : {
        "address" :  "sender@example.com"
    },
    "subject" : "Hello From Amazon WorkMail!",
    "messageId": "00000000-0000-0000-0000-000000000000",
    "invocationId": "00000000000000000000000000000000",
    "flowDirection": "INBOUND",
    "truncated": false
}
```

The event JSON includes the following data.

summaryVersion

The version number for `LambdaEventData`. This only updates when you make a backwards incompatible change in `LambdaEventData`.

envelope

The envelope of the email message, which includes the following: fields.

mailFrom

The **From** address, which is usually the email address of the user who sent the email message. If the user sent the email message as another user or on behalf of another user, the **mailFrom** field returns the email address of the user on whose behalf the email message was sent, not the email address of the actual sender.

recipients

A list of recipient email addresses. Amazon WorkMail doesn't distinguish between **To**, **CC**, or **BCC**.

> **Note**
>
> For inbound email flow rules, this list includes recipients in all the domains in the Amazon WorkMail organization in which you create the rule. The Lambda function is invoked separately for each SMTP conversation from the sender, and the recipients field lists the recipients from that SMTP conversation. Recipients with external domains are not included.

sender

The email address of the user who sent the email message on behalf of another user. This field is set only when an email message is sent on behalf of another user.

subject

The email subject line. Truncated when it exceeds the 256 character limit.

messageId

A unique ID used to access the full content of the email message when using the Amazon WorkMail Message Flow SDK.

invocationId

The ID for a unique Lambda invocation. This ID remains the same when a Lambda function is called more than once for the same **LambdaEventData**. Use to detect retries and avoid duplication.

flowDirection

Indicates the direction of the email flow, either **INBOUND** or **OUTBOUND**.

truncated

Applies to the payload size, not the subject line length. When `true`, the payload size exceeds the 128 KB limit, so the list of recipients is truncated in order to meet the limit.

Synchronous Run Lambda response schema

When an email flow rule with a synchronous **Run Lambda** action matches an inbound or outbound email message, Amazon WorkMail calls the configured Lambda function and waits for the response before taking action on the email message. The Lambda function returns a response according to a pre-defined schema that lists the actions, action types, applicable parameters, and recipients that the action applies to.

The following example shows a synchronous **Run Lambda** response. Responses vary based on the programming language used for the Lambda function.

```
{
    "actions": [
      {
```

```
        "action" : {
          "type": "string",
          "parameters": { various }
        },
        "recipients": [list of strings],
        "allRecipients": boolean
      }
    ]
}
```

The response JSON includes the following data.

action

The action to take for the recipients.

type

The action type. Action types are not returned for asynchronous **Run Lambda** actions.

Inbound rule action types include **BOUNCE**, **DROP**, **DEFAULT**, **BYPASS_SPAM_CHECK**, and **MOVE_TO_JUNK**. For more information, see Inbound email rule actions.

Outbound rule action types include **BOUNCE**, **DROP**, and **DEFAULT**. For more information, see Outbound email rule actions.

parameters

Additional action parameters. Supported for the **BOUNCE** action type as a JSON object with the key **bounceMessage** and value **string**. This bounce message is used to create the bounce email message.

recipients

List of email addresses on which the action should be taken. You can add new recipients to the response even if they were not included in the original recipients list. This field is not required if **allRecipients** is true for an action.

> **Note**
>
> When a Lambda action is called for inbound email, you can only add new recipients that are from your organization. The new recipients are added to the response as **BCC**.

allRecipients

When true, applies the action to all the recipients that are not subject to another specific action in the Lambda response.

Synchronous Run Lambda action limits

The following limits apply when Amazon WorkMail invokes Lambda functions for synchronous **Run Lambda** actions:

- Lambda functions must respond within 15 seconds, or be treated as failed invocations.

> **Note**
>
> The system retries the invocation for the **Rule timeout** interval that you specify.

- Lambda function responses up to 256 KB are allowed.
- Up to 10 unique actions are allowed in the response. Actions greater than 10 are subject to the configured **Fallback action**.
- Up to 500 recipients are allowed for outbound Lambda functions.
- The maximum value for **Rule timeout** is 240 minutes. If the minimum value of 0 is configured, there are no retries before Amazon WorkMail applies the fallback action.

Synchronous Run Lambda action failures

If Amazon WorkMail can't invoke your Lambda function due to an error, invalid response, or Lambda timeout, Amazon WorkMail retries the invocation with exponential backoff that decreases the processing rate until the **Rule timeout** period completes. Then, the **Fallback action** is applied to all recipients of the email message. For more information, see Configuring synchronous **Run Lambda** rules.

Example synchronous Run Lambda responses

The following examples demonstrate the structure of common synchronous **Run Lambda** responses.

Example : Remove specified recipients from an email message

The following example demonstrates the structure of a synchronous **Run Lambda** response for removing recipients from an email message.

```
{
    "actions": [
      {
        "action": {
          "type": "DEFAULT"
        },
        "allRecipients": true
      },
      {
        "action": {
          "type": "DROP"
        },
        "recipients": [
          "drop-recipient@example.com"
        ]
      }
    ]
}
```

Example : Bounce with a custom email message

The following example demonstrates the structure of a synchronous **Run Lambda** response for bouncing with a custom email message.

```
{
    "actions" : [
      {
        "action" : {
          "type": 'BOUNCE',
          "parameters": {
            "bounceMessage" : "Email in breach of company policy."
          }
        },
        "allRecipients": true
      }
    ]
}
```

Example : Add recipients to an email message

The following example demonstrates the structure of a synchronous **Run Lambda** response for adding recipients to the email message. This does not update the **To** or **CC** fields of the email message.

```
{
    "actions": [
      {
        "action": {
          "type": "DEFAULT"
        },
        "recipients": [
          "new-recipient@example.com"
         ]
      },
      {
        "action": {
          "type": "DEFAULT"
        },
        "allRecipients": true
      }
    ]
}
```

For more code examples to use when creating Lambda functions for **Run Lambda** actions, see Amazon WorkMail Lambda templates.

More information about using Lambda with Amazon WorkMail

You can also access the full content of the email message that triggers the Lambda function. For more information, see Retrieving message content with AWS Lambda.

Retrieving message content with AWS Lambda

After you configure an AWS Lambda function to manage email flows for Amazon WorkMail, you can access the full content of the email messages that are processed using Lambda. For more information about getting started with Lambda for Amazon WorkMail, see Configuring AWS Lambda for Amazon WorkMail.

To access the full content of email messages, use the `GetRawMessageContent` action in the Amazon WorkMail Message Flow API. The email message ID that is passed to your Lambda function

upon invocation sends a request to the API. Then, the API responds with the full MIME content of the email message. For more information, see Amazon WorkMail Message Flow in the *Amazon WorkMail API Reference*.

The following example shows how a Lambda function using the Python runtime environment can retrieve the full message content.

Tip

If you start by deploying the Amazon WorkMail Hello World Lambda function from the AWS Serverless Application Repository to your account, the system creates a Lambda function in your account with all the necessary resources and permission. You can then add your business logic to the lambda function based on your use-case.

```
import boto3
import email
import os

def email_handler(event, context):
    workmail = boto3.client('workmailmessageflow',
 region_name=os.environ["AWS_REGION"])
    msg_id = event['messageId']
    raw_msg = workmail.get_raw_message_content(messageId=msg_id)

    parsed_msg = email.message_from_bytes(raw_msg['messageContent'].read())
    print(parsed_msg)
```

For more detailed examples of ways to analyze the content of messages that are in transit, see the amazon-workmail-lambda-templates repository on GitHub.

Note

You only use the Amazon WorkMail Message Flow API to access email messages in transit. You can only access the messages within 24 hours of being sent or received. To programmatically access messages in a user's mailbox, use one of the other protocols supported by Amazon WorkMail, such as IMAP or Exchange Web Services (EWS).

Updating message content with AWS Lambda

After you configure a synchronous AWS Lambda function to manage email flows, you can use the `PutRawMessageContent` action in the Amazon WorkMail Message flow API to update the content of in-transit email messages. For more information about getting started with Lambda functions for Amazon WorkMail, see Configuring synchronous **Run Lambda** rules. For more information about the API, see PutRawMessageContent.

> **Note**
>
> The PutRawMessageContent API requires boto3 1.17.8, or you can add a layer to your Lambda function. To download the correct boto3 version, see the boto page on GitHub. For more information about adding layers, see Configure a function to use layers.
> Here's an example layer: `"LayerArn":"arn:aws:lambda:${AWS::Region}:489970191081:layer:WorkMailLambdaLayer:2"`. In this example, substitute `${AWS::Region}` with an appropriate aws region, such as us-east-1.

> **Tip**
>
> If you start by deploying the Amazon WorkMail Hello World Lambda function from the AWS Serverless Application Repository to your account, the system creates a Lambda function in your account with the necessary resources and permissions. You can then add business logic to the lambda function, based on your use cases.

As you go, remember the following:

- Use the GetRawMessageContent API to retrieve the original message content. For more information see Retrieving message content with AWS Lambda.
- Once you have the original message, change the MIME content. When you finish, upload the message to an Amazon Simple Storage Service (Amazon S3) bucket in your account. Ensure that the S3 bucket uses the same AWS account as your Amazon WorkMail operations, and that it uses the same AWS Region as your API calls.
- For Amazon WorkMail to process requests, your S3 bucket must have the correct policy in order to access the S3 object. For more information, see Example S3 policy.

- Use the PutRawMessageContent API to send the updated the message content back to Amazon WorkMail.

> **Note**
>
> The PutRawMessageContent API ensures that the MIME content of the updated message meets RFC standards, as wells as the criteria mentioned in the RawMessageContent data type. Emails inbound to your Amazon WorkMail organization don't always meet those standards, so the PutRawMessageContent API may reject them. In such cases, you can consult the error message returned for more information on how to fix any issues.

Example Example S3 policy

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {"Service": "workmail.REGION.amazonaws.com"
            },
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::My-Test-S3-Bucket/*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "AWS_ACCOUNT_ID"
                },
                "Bool": {
                    "aws:SecureTransport": "true"
                },
                "ArnLike": {
                    "aws:SourceArn":
 "arn:aws:workmailmessageflow:REGION:AWS_ACCOUNT_ID:message/WORKMAIL_ORGANIZATION_ID/*"
                }
            }
        }
    ]
```

```
}
```

The following example shows how a Lambda function uses the Python runtime to update the subject of an in-transit email message.

```
    import boto3
    import os
    import uuid
    import email

    def email_handler(event, context):
        workmail = boto3.client('workmailmessageflow',
 region_name=os.environ["AWS_REGION"])
        s3 = boto3.client('s3', region_name=os.environ["AWS_REGION"])

        msg_id = event['messageId']
        raw_msg = workmail.get_raw_message_content(messageId=msg_id)
        parsed_msg = email.message_from_bytes(raw_msg['messageContent'].read())

        # Updating subject. For more examples, see https://github.com/aws-samples/
amazon-workmail-lambda-templates.
        parsed_msg.replace_header('Subject', "New Subject Updated From Lambda")

        # Store updated email in S3
        key = str(uuid.uuid4());
        s3.put_object(Body=parsed_msg.as_bytes(), Bucket="amzn-s3-demo-bucket",
 Key=key)

        # Update the email in WorkMail
        s3_reference = {
            'bucket': "amzn-s3-demo-bucket",
            'key': key
        }
        content = {
            's3Reference': s3_reference
        }
        workmail.put_raw_message_content(messageId=msg_id, content=content)
```

For more examples of ways to analyze the content of in-transit messages, see the amazon-workmail-lambda-templates repository on GitHub.

Managing access to the Amazon WorkMail Message Flow API

Use AWS Identity and Access Management (IAM) policies to manage access to the Amazon WorkMail Message Flow API.

The Amazon WorkMail Message Flow API works with a single resource type, an email message in transit. Each email message in transit has a unique Amazon Resource Name (ARN) associated with it.

The following example shows the syntax of an ARN associated with an email message in transit.

```
arn:aws:workmailmessageflow:region:account:message/organization/context/messageID
```

Changeable fields in the preceding example include the following:

- **Region** – The AWS Region for your Amazon WorkMail organization.
- **Account** – The AWS account ID for your Amazon WorkMail organization.
- **Organization** – Your Amazon WorkMail organization ID.
- **Context** – Indicates whether the message is `incoming` to your organization, or `outgoing` from it.
- **Message ID** – The unique email message ID that is passed as input to your Lambda function.

The following example includes example IDs for an ARN associated with an incoming email message in transit.

```
arn:aws:workmailmessageflow:us-east-1:111122223333:message/m-
n1pq2345678r901st2u3vx45x6789yza/incoming/d1234567-8e90-1f23-456g-hjk7lmnop8q9
```

You can use these ARNs as resources in the `Resource` section of your IAM user policies in order to manage access to Amazon WorkMail messages in transit.

Example IAM policies for Amazon WorkMail message flow access

The following example policy grants an IAM entity full read access to all inbound and outbound messages for every Amazon WorkMail organization in your AWS account.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent"
            ],
            "Resource": "arn:aws:workmailmessageflow:region:account:message/*",
            "Effect": "Allow"
        }
    ]
}
```

If you have multiple organizations in your AWS account, you can also limit access to one or more organizations. This is useful if certain Lambda functions should only be used for certain organizations.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent"
            ],
            "Resource":
 "arn:aws:workmailmessageflow:region:account:message/organization/*",
            "Effect": "Allow"
        }
    ]
}
```

You can also choose to grant access to messages depending on whether they are `incoming` to your organization, or `outgoing` from it. To do this, use the qualifier `incoming` or `outgoing` in the ARN.

The following example policy grants access only to messages that are incoming to your organization.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent"
            ],
            "Resource":
 "arn:aws:workmailmessageflow:region:account:message/organization/incoming/*",
            "Effect": "Allow"
        }
    ]
}
```

The following example policy grants an IAM entity full read and update access to all inbound and outbound messages for every Amazon WorkMail organization in your AWS accounts.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent",
                "workmailmessageflow:PutRawMessageContent"
            ],
            "Resource": "arn:aws:workmailmessageflow:region:account:message/*",
            "Effect": "Allow"
        }
    ]
}
```

Testing an email flow rule

To check your current rule configuration, you can test how the configuration behaves against specific email addresses.

To test an email flow rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Inbound/Outbound rules**.
4. Next to **Test configuration**, enter the full email addresses of both the sender and recipient that you want to test.
5. Choose **Test**. The action to be taken for the provided email address is displayed.

Removing an email flow rule

When you remove an email flow rule, the changes are applied immediately.

To remove an email flow rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Inbound/Outbound rules**.
4. Select the rule and choose **Remove**.
5. At the confirmation prompt, choose **Remove**.

Enforcing DMARC policies on incoming email

Email domains use Domain Name System (DNS) records for security. They protect your users from common attacks such as spoofing or phishing. DNS records often include Domain-based Message Authentication, Reporting, and Conformance (DMARC) records, which are set by the domain owner that sends the email. DMARC records include policies that specify actions to take when an email fails a DMARC check. You can choose whether to enforce the DMARC policy on emails being sent to your organization.

New Amazon WorkMail organizations have DMARC enforcement turned on by default.

To turn on DMARC enforcement

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Organization settings**. The **Organization settings** page appears and displays a set of tabs.
4. Choose the **DMARC** tab, then choose **Edit**.
5. Move the **DMARC enforcement** slider to the on position.
6. Select the check box next to **I acknowledge that turning on DMARC enforcement may result in inbound emails being dropped or quarantined based on the sender's domain configuration.**
7. Choose **Save**.

To turn off DMARC enforcement

- Follow the steps in the previous section, but move the **DMARC enforcement** slider to the off position..

Using email event logging to track DMARC enforcement

Turning on DMARC enforcement might result in inbound emails being dropped or marked as spam, depending on how the sender configured their domain. If a sender misconfigures their

email domain, your users might stop receiving legitimate emails. To check for emails that aren't being delivered to your users, you can enable email event logging for your Amazon WorkMail organization. Then, you can query your email event logs for inbound emails that are filtered out based on the sender's DMARC policies.

Before you use email event logging to track DMARC enforcement, enable email event logging in the Amazon WorkMail console. To get the most out of your log data, allow some time to pass while email events are logged. For more information and instructions, see the section called “Turning on email event logging”.

To use email event logging to track DMARC enforcement

1. In the CloudWatch Insights console, under **Logs**, choose **Insights**.
2. For **Select log group(s)**, select your Amazon WorkMail organization's log group. For example, /aws/workmail/events/organization-alias.
3. Select a time period to query.
4. Run the following query: **stats count() by event.dmarcPolicy | filter event.dmarcVerdict == "FAIL"**
5. Choose **Run query**.

You can also set up custom metrics for these events. For more information, see Creating metric filters.

Tagging an organization

Tagging an Amazon WorkMail organization resource allows you to:

- Differentiate between organizations in the AWS Billing and Cost Management console.
- Control access to Amazon WorkMail organization resources by adding them to the `Resource` element of AWS Identity and Access Management (IAM) permission policy statements.

For more information about Amazon WorkMail resource-level permissions, see Resources. For more information about controlling access based on tags, see Authorization based on Amazon WorkMail tags.

Amazon WorkMail administrators can tag organizations using the Amazon WorkMail console.

To add tags to an Amazon WorkMail organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Tags**.
4. For **Organization tags**, choose **Add new tag**.
5. For **Key**, enter a name that identifies the tag.
6. (Optional) For **Value**, enter a value for the tag.
7. (Optional) Repeat steps 4-6 to add more tags to your organization. You can add up to 50 tags.
8. Choose **Save** to save your changes.

You can view your organization tags in the Amazon WorkMail console.

Developers can also tag organizations using the AWS SDK or AWS Command Line Interface (AWS CLI). For more information, see the `TagResource`, `ListTagsForResource`, and `UntagResource` commands in the Amazon WorkMail API Reference or the AWS CLI Command Reference.

You can remove tags from an organization at any time, using the Amazon WorkMail console.

To remove tags from an Amazon WorkMail organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Tags**.
4. For **Organization tags**, choose **Remove** next to the tag to remove.
5. Choose **Submit** to save your changes.

Working with access control rules

Access control rules for Amazon WorkMail allow administrators to control how their organization's users and impersonation roles are granted access to Amazon WorkMail. Each Amazon WorkMail organization has a default access control rule that grants mailbox access to all users and impersonation roles added to the organization, no matter which access protocol or IP address they use. Administrators can edit or replace the default rule with one of their own, add a new rule, or delete a rule.

> **Warning**
>
> If an administrator deletes all access control rules for an organization, Amazon WorkMail blocks all access to the organization's mailboxes.

Administrators can apply access control rules that allow or deny access based on the following criteria:

- **Protocols** – The protocol used to access the mailbox. Examples include **Autodiscover**, **EWS**, **IMAP**, **SMTP**, **ActiveSync**, **Outlook for Windows**, and **Webmail**.
- **IP addresses** – The IPv4 CIDR ranges used to access the mailbox.
- **Amazon WorkMail users** – The users in your organization that are used to access the mailbox.
- **Impersonation roles** – The impersonation roles in your organization that are used to access the mailbox. For more information, see Managing impersonation roles.

Administrators apply access control rules in addition to the user's mailbox and folder permissions. For more information, see Working with mailbox permissions and Sharing folders and folder permissions in the *Amazon WorkMail User Guide*.

> **Note**
>
> - When you are enabling access for Outlook for Windows, it is recommended to also enable access for Autodiscover and EWS.
> - Access control rules do not apply to Amazon WorkMail console or SDK access. Use AWS Identity and Access Management (IAM) roles or policies instead. For more information, see Identity and access management for Amazon WorkMail.

Creating access control rules

Create new access control rules from the Amazon WorkMail console.

To create a new access control rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Access control rules**.
4. Choose **Create rule**.
5. For **Description**, enter a description for the rule.
6. For **Effect**, choose **Allow** or **Deny**. This allows or denies access based on the conditions that you select in the following step.
7. For **This rule applies to requests that ...**, select the conditions to apply to the rule, such as whether to include or exclude specific protocols, IP addresses, or users, or impersonation roles.
8. (Optional) If you enter IP address ranges, users, or impersonation roles, choose **Add** to add them to the rule.
9. Choose **Create rule**.

Editing access control rules

Edit new and default access control rules from the Amazon WorkMail console.

To edit an access control rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.

3. Choose **Access control rules**.
4. Select the rule to edit.
5. Choose **Edit rule**.
6. Edit the description, effect, and conditions, as needed.
7. Choose **Save changes**.

> **Important**
>
> When you change an access rule, the affected mailboxes can take five minutes to follow the updated rule. Clients that access the affected mailboxes may show inconsistent behavior during that time. However, you will immediately see correct behavior when you test your rules. For more information about testing rules, see the steps in the next section.

Testing access control rules

To see how your organization's access control rules are applied, test the rules from the Amazon WorkMail console.

To test access control rules for your organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Access control rules**.
4. Choose **Test rules**.
5. For **Request context**, select the protocol to test for.
6. For **Source IP address**, enter the IP address to test for.
7. For **Request performed by**, choose **User** or **Impersonation role** to test for.
8. Select **User** or **Impersonation role** to test for.
9. Choose **Test**.

The test results appear under **Effect**.

Deleting access control rules

Delete access control rules that you no longer require from the Amazon WorkMail console.

> **Warning**
>
> If an administrator deletes all access control rules for an organization, Amazon WorkMail blocks all access to the organization's mailboxes.

To delete an access control rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Access control rules**.
4. Select the rule to delete.
5. Choose **Delete rule**.
6. Choose **Delete**.

Setting mailbox retention policies

You can set mailbox retention policies for your Amazon WorkMail organization. Retention policies automatically delete email messages from user mailboxes after a time period that you choose. You can choose which mailbox folders to apply retention policies to. Also, you can choose whether to set different retention policies for different folders. Mailbox retention policies apply to the selected folders in all of the user mailboxes in your organization. Users can't override the retention policies.

To set a mailbox retention policy

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. Choose **Retention policy**.
4. For **Folder actions**, next to each mailbox folder that you want to include in the policy, select **Delete** or **Permanently delete**.
5. Enter the number of days to keep the email messages in each mailbox folder before deleting them.
6. Choose **Save**.

Allow 48 hours to apply the retention policies for your organization. If you choose the **Delete** folder action, users can recover deleted email messages from the Amazon WorkMail web application and supported clients. If you choose the **Permanently delete** folder action, email messages can't be recovered after they are deleted.

The number of days a retention policy keeps an item is based on when it was created, modified, or moved. For example, if a retention policy deletes items after a year, the policy counts retention days from the date you created or last took action on that item. It is not affected by the date you implemented the retention policy.

Working with domains

You can configure Amazon WorkMail to use a custom domain. You can also make a domain the default for your organization, and enable AutoDiscover for Microsoft Outlook.

Topics

- Adding a domain
- Removing a domain
- Choosing the default domain
- Verifying domains
- Enabling AutoDiscover to configure endpoints
- Editing domain identity policies
- Authenticating email with SPF
- Configuring a custom MAIL FROM domain

Adding a domain

You can add up to 100 domains to your Amazon WorkMail organization. When you add a new domain, an Amazon Simple Email Service (Amazon SES) sending authorization policy is automatically added to the domain identity policy. This provides Amazon WorkMail with access to all Amazon SES sending actions for your domain and allows you to redirect email to your domain. You can also redirect email to external domains.

> **Note**
>
> As a best practice, you should add aliases for `<postmaster@>` and `<abuse@>` to all your domains. You can create distribution groups for these aliases if you want specific users in your organization to receive mail sent to these aliases.

When you configure your Amazon WorkMail organization with a custom domain, remember the following about your domain's DNS records:

- For MX and autodiscover CNAME records, we recommend setting the **Time to Live (TTL)** value to 3600. Reducing the TTL ensures that your mail servers don't use outdated or invalid MX records after you update those records or migrate your mailboxes.
- After you create your users and distribution groups, and then successfully migrate your mailboxes, you should update the MX record to start forwarding emails to Amazon WorkMail. Updates to DNS records can take up to 48 hours to process.
- Some DNS providers automatically append domain names to the ends of DNS records. Adding a record that already contains the domain name, such as _amazonses.example.com, might result in the duplication of the domain name, resulting in _amazonses.example.com.example.com. To avoid duplicating the domain name in the record name, add a period to the end of the domain name in the DNS record. This indicates to your DNS provider that the record name is fully qualified and no longer relative to the domain name. It also prevents the DNS provider from appending an additional domain name.
- Copied record names include the domain name. Depending on which DNS service you use, the domain name might already be added to the domain's DNS record.
- After you create a DNS record, choose the refresh icon on the Amazon WorkMail console to see the verification status and record value. For more information about verifying domains, see Verifying domains.
- We recommend that you configure your domain as the `MAIL FROM` domain. To enable AutoDiscover for iOS devices, you must configure your domain as the `MAIL FROM` domain. You can see the status of your `MAIL FROM` domain in the **Enhance deliverability** section of the console. For more information, see Configuring a custom MAIL FROM domain.

To add a domain

1. Sign in to the AWS Management Console and open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. In the navigation pane, choose **Organizations**, and then choose the name of the organization that you want to add a domain to.

4. In the navigation pane, choose **Domains**, and then choose **Add domain**.
5. On the **Add domain** screen, enter a domain name. Domain names can contain only Basic Latin (ASCII) characters.

> **Note**
>
> If you have a domain that is managed in an Amazon Route 53 public hosted zone, you can choose it from the dropdown menu that appears as you enter a domain name.

6. Choose **Add domain**.

A page appears and lists the DNS records for the new domain. The page groups the records into the following sections:

- **Domain ownership**
- **WorkMail configuration**
- **Improved security**
- **Improved email delivery**

Each of these sections contains one or more DNS records, and each record displays a **Status** value. The following list shows the records and their available status values.

TXT ownership

Verified – Record resolved and verified.

Pending – Record not verified yet.

Failed – Unable to verify ownership. Record mismatched or unreachable.

MX WorkMail configuration

Verified – Record resolved and verified.

Missing – Unable to resolve record.

Inconsistent – Value does not match expected record.

AutoDiscover

Verified – Record resolved and verified.

Missing – Unable to resolve record.

Inconsistent – Value does not match expected record.

> **Note**
>
> The AutoDiscover verification process also checks for correct AutoDiscover setup. The process verifies the configuration settings for each phase. A green check mark appears next to **Verified** in the **Status** column when verification finishes. You can hover over **Verified** and see which of the phases was verified by the process. For more information about the AutoDiscover phases, see Enabling AutoDiscover to configure endpoints.

DKIM CNAME

Verified – Record resolved and verified.

Pending – Record not verified yet

Failed – Unable to verify ownership. Record mismatched or unreachable.

For more information about DKIM signing, see Authenticating email with DKIM in Amazon SES in the *Amazon Simple Email Service Developer Guide*.

SPF TXT

Verified – Record resolved and verified.

Missing – Unable to resolve record.

Inconsistent – Value does not match expected record.

For more information about SPF verification, see Authenticating email with SPF.

DMARC TXT

Verified – Record resolved and verified.

Missing – Unable to resolve record.

Inconsistent – Value does not match expected record

For more information about DMARC records in Amazon WorkMail, see Complying with DMARC using Amazon SES in the *Amazon Simple Email Service Developer Guide*.

TXT MAIL FROM domain

Verified – Record resolved and verified.

Pending – Record not verified yet.

Failed – Unable to verify ownership. Record mismatched or unreachable.

MX MAIL FROM domain

Verified – Record resolved and verified.

Missing – Unable to resolve record.

Inconsistent – Value does not match expected record.

7. For the next step, choose the appropriate action based on the DNS provider you use.

   If you use a Route 53 domain

   - At the top of the page, choose **Update all in Route 53**.

   If you use another DNS provider

   - Copy the records and paste them into your DNS provider. You can copy the records in bulk or one at a time. To copy records in bulk, choose **Copy all**. That creates a file zone that you can import into your DNS provider. To copy records one at a time, choose the overlapping squares next to the record name, and then paste each one into your DNS provider.

8. Choose the refresh icon update the **Status** for each record. This verifies domain ownership and proper configuration of your domain with Amazon WorkMail.

Removing a domain

When you no longer need a domain, you can delete it. However, you must first delete any individuals or groups that use the domain as their email address.

To remove a domain

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region Name and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the list of domains, select the check box next to the domain name and choose **Remove**.
4. In the **Remove domain** dialog box, enter the name of the domain to remove and choose **Remove**.

Choosing the default domain

You can make a domain associated with your organization the default for users and groups in that organization. Making a domain the default does not change existing email addresses.

To make a domain the default

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region Name and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the list of domains, select the check box next to the domain name you want to use and choose **Set as default**.

Verifying domains

You must verify your domain after you add it in the Amazon WorkMail console. Verifying the domain confirms that you own the domain and will use Amazon WorkMail as the email service for the domain.

You verify a domain by adding TXT and MX records to it in your DNS service. TXT records enable you to add notes to your DNS service. MX records specify the incoming mail server.

You use the Amazon SES console to create the TXT and MX records, and then you use the Amazon WorkMail console to add the records to your DNS service. Follow these steps.

To create TXT and MX records

1. Open the Amazon SES console at https://console.aws.amazon.com/ses/.
2. In the navigation pane, choose **Domains**, and then choose **Verify a New Domain**.

   The **Verify a New Domain dialog box appears.**
3. In the **Domain** box, enter the name of the domain that you created in the Adding a domain section.
4. (Optional) If you want to use DomainKeys Identified Mail (DKIM), select the **Generate DKIM Settings** check box.
5. Choose **Verify This Domain**.

   The console displays a list of TXT and MX records.
6. Choose the **Download Record Set as CSV** link, located under the TXT listing.

   The **Save As** dialog box appears. Choose a location for the download, and then choose **Save**.
7. Open the downloaded CSV file and copy all of its contents.

Once your create the TXT and MX records, you then add them to your DNS provider. The following steps use Route 53. If you use a different DNS provider and you don't know how to add records, consult your provider's documentation.

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted Zones**. Then, choose the radio button next to the domain that you want to verify.
3. From the list of DNS records for your domain, choose **Import zone file**.
4. Under **Zone file**, paste the copied records into the text box. A list of the files appears below the text box.
5. Scroll down to the end of the list and choose **Import**.

> **Note**
>
> Allow up to 72 hours to complete the verification process.

Verifying TXT records and MX records with your DNS service

Confirm that the TXT record that verifies that you own the domain is added correctly to your DNS service. This procedure uses the nslookup tool, which is available for Windows and Linux. On Linux, you can also use dig.

To use the `nslookup` tool, you must first find the DNS servers that serve your domain. Then, you query those servers to view the TXT records. You can query the DNS servers for your domain because those servers contain the most up-to-date information for your domain. This information can take time to propagate to other DNS servers.

Use nslookup to verify that your TXT record is added to your DNS service

1. Find your domain's name servers:

   a. Open a command prompt (Windows) or terminal (Linux).

   b. Run the following command to list all of the name servers that serve your domain. Replace *`example.com`* with your domain.

   ```
   nslookup -type=NS example.com
   ```

   You'll query one of these name servers in the next step.

2. Verify that the Amazon WorkMail TXT record is correctly added.

   a. Run the following command, replacing *`example.com`* with your domain, and *`ns1.name-server.net`* with a name server from Step 1.

   ```
   nslookup -type=TXT _amazonses.example.com ns1.name-server.net
   ```

   b. Review the `"text ="` string shown in the output from **nslookup**. Confirm that this string matches the TXT value for your domain in the **Verified Senders** list in the Amazon WorkMail console.

In the following example, you want to see a TXT record for _amazonses.example.com with a value of `fmxqxT/icOYx4aA/bEUrDPMeax9/s3frblS+niixmqk=`. If you update the record correctly, the command has the following output:

```
_amazonses.example.com text = "fmxqxT/icOYx4aA/bEUrDPMeax9/s3frblS+niixmqk="
```

Use dig to verify that your TXT record is added to your DNS service

1. Open a terminal session.
2. Run the following command to list the TXT records for your domain. Replace *example.com* with your domain.

```
dig +short example.com txt
```

3. Verify that the string that follows TXT in the command's output matches the TXT value you see when you select the domain in the **Verified Senders** list of the Amazon WorkMail console.

To use nslookup to verify that your MX record is added to your DNS service

1. Find the name servers for your domain:
   a. Open a command prompt.
   b. Run the following command to list all of the name servers for your domain.

```
nslookup -type=NS example.com
```

   You'll query one of these name servers in the next step.
2. Verify that the MX record is correctly added:
   a. Run the following command, replacing *example.com* with your domain and *ns1.name-server.net* with one of the name servers you identified in the previous step..

```
nslookup -type=MX example.com ns1.name-server.net
```

   b. In the output of the command, verify that the string that follows `mail exchange =` matches one of the following values:

US East (N. Virginia) Region – `10 inbound-smtp.us-east-1.amazonaws.com`

US West (Oregon) Region – `10 inbound-smtp.us-west-2.amazonaws.com`

Europe (Ireland) Region – `10 inbound-smtp.eu-west-1.amazonaws.com`

> **Note**
>
> `10` represents the MX preference number or priority.

Use dig to verify that your MX record is added to your DNS service

1. Open a terminal session.
2. Run the following command to list the MX records for your domain.

```
dig +short example.com mx
```

3. Verify that the string that follows `MX` matches one of the following values:

   US East (N. Virginia) Region – `10 inbound-smtp.us-east-1.amazonaws.com`

   US West (Oregon) Region – `10 inbound-smtp.us-west-2.amazonaws.com`

   Europe (Ireland) Region – `10 inbound-smtp.eu-west-1.amazonaws.com`

   > **Note**
   >
   > `10` represents the MX preference number or priority.

Troubleshooting domain verification

To troubleshoot common issues with domain verification, see the following suggestions:

Your DNS service does not allow underscores in TXT record names

Omit `_amazonses` from the TXT record name.

You want to verify the same domain multiple times but can't have multiple TXT records with the same name

If your DNS service does not allow you to have multiple TXT records with the same name, use either of the following workarounds:

- (Recommended) If your DNS service allows it, assign multiple values to the TXT record. For example, if your DNS is managed by Amazon Route 53, you can set up multiple values for the same TXT record as follows:
  1. In the Route 53 console, choose the `_amazonses` TXT record that you added when you verified your domain in the first Region.
  2. For **Value**, press **Enter** after the first value.
  3. Add the value for the additional Region, and save the record set.
- If you only need to verify your domain twice, you can verify it once by creating a TXT record with `_amazonses` in the name, and then create another record without `_amazonses` in the record name.

The Amazon WorkMail console reports that domain verification has failed

Amazon WorkMail can't find the necessary TXT record for your DNS service. Verify that the required TXT record is correctly added to your DNS service by following the procedure in Verifying TXT records and MX records with your DNS service.

Your DNS provider appended the domain name to the end of the TXT record

Adding a TXT record that already contains the domain name, such as _amazonses.example.com, can result in the duplication of the domain name, such as _amazonses.example.com.example.com. To avoid duplicating the domain name in the record name, add a period to the end of the domain name in the TXT record. This indicates to your DNS provider that the record name is fully qualified and already has the domain name included in the TXT record.

Amazon WorkMail reports that the MX record is **Inconsistent**

When migrating from existing mail servers, the MX record might return a status of **Inconsistent**. Update your MX record to point to Amazon WorkMail instead of pointing to your previous mail server. The MX record is also returned as **Inconsistent** when a third-party email proxy is used along with Amazon WorkMail. If this is the case, it is safe to ignore the **Inconsistent** warning.

Enabling AutoDiscover to configure endpoints

AutoDiscover enables you to configure Microsoft Outlook and mobile clients by using only your email address and password. The service maintains a connection to Amazon WorkMail and updates local settings whenever you change endpoints or settings. In addition, AutoDiscover enables your client to use additional Amazon WorkMail features, such as the Offline Address Book, Out-of-Office Assistant, and the ability to view free/busy time in Calendar.

The client performs the following AutoDiscover phases to detect the server endpoint URLs:

- **Phase 1** – The client performs a Secure Copy Protocol (SCP) lookup against the local Active Directory. If your client isn't domain-joined, AutoDiscover skips this step.
- **Phase 2** – The client sends a request to the following URLs and validates the results. These endpoints are only available using HTTPS.
  - https://*company.tld*/autodiscover/autodiscover.xml
  - https://autodiscover.*company.tld*/autodiscover/autodiscover.xml
- **Phase 3** – The client performs a DNS lookup to autodiscover.company.tld and sends an unauthenticated GET request to the derived endpoint from the user's email address. If the server returns a 302 redirect, the client resends the AutoDiscover request against the returned HTTPS endpoint.

If all of these phases fail, the client can't be configured automatically. For information about manually configuring mobile devices, see Manually connect your device.

You are prompted to add the AutoDiscover DNS record to your provider when you add your domain to Amazon WorkMail. This enables the client to perform phase 3 of the AutoDiscover process. However, these steps don't work for all mobile devices, such as the stock Android email app. As a result, you may need to set up AutoDiscover phase 2 manually.

You can use the following methods to set up AutoDiscover phase 2 for your domain:

(Recommended) Use Route 53 and Amazon CloudFront

> **Note**
>
> The following steps explain how to create a proxy for https://autodiscover.*company.tld*/autodiscover/autodiscover.xml. To create a proxy for https://*company.tld*/autodiscover/

autodiscover.xml, remove the `autodiscover.` prefix from the domains in the following steps.
Using CloudFront and Route 53 may incure charges. For more information about applicable pricing, see Amazon CloudFront pricing and Amazon Route 53 pricing.

To enable AutoDiscover phase 2 with Route 53 and CloudFront

1. Get an SSL certificate for autodiscover.*company.tld* and upload it to AWS Identity and Access Management (IAM) or AWS Certificate Manager. For more information, see Working with server certificates in the *IAM User Guide*, or Getting started in the *AWS Certificate Manager User Guide*.
2. Create a new CloudFront distribution:
   1. Open the CloudFront console at https://console.aws.amazon.com/cloudfront/v4/home.
   2. In the navigation pane, choose **Distributions**.
   3. Choose **Create Distribution**.
   4. Under **Web**, choose **Get Started**.
   5. In **Origin Settings**, enter the following values:
      - **Origin Domain Name** – The appropriate domain name for your Region:
        - US East (N. Virginia) — **`autodiscover-service.mail.us-east-1.awsapps.com`**
        - US West (Oregon) — **`autodiscover-service.mail.us-west-2.awsapps.com`**
        - Europe (Ireland) — **`autodiscover-service.mail.eu-west-1.awsapps.com`**
      - **Origin Protocol Policy** – The desired policy: **`Match Viewer`**

        Note

        Leave **Origin path** blank. Don't change the auto-populated value for **Origin ID**.
   6. In **Default Cache Behavior Settings**, select the following values for the listed settings:
      - **Viewer Protocol Policy**: HTTPS Only
      - **Allowed HTTP Methods**: GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE
      - **Cache Based on Selected Request Headers**: All
      - **Forward Cookies**: All
      - **Query String Forwarding and Caching**: None (Improves Caching)

- **Smooth Streaming**: No
- **Restrict Viewer Access**: No

7. Select the following values for **Distribution Settings**:
   - **Price Class**: Use only US, Canada, and Europe
   - For **Alternate Domain Names (CNAMEs)**, enter **`autodiscover.`*`company.tld`*** or ***`company.tld`***, where *`company.tld`* is your domain name.
   - **SSL Certificate**: Custom SSL Certificate (stored in IAM)
   - **Custom SSL Client Support**: Choose **All Clients** or **Only Clients that Support Server Name Indication (SNI)**. Older versions of Android might not work with the latter option.

     > **Note**
     >
     > If you choose **All Clients**, leave **Default Root Object** blank.

   - **Logging**: Choose **On** or **Off**. **On** enables logging.
   - For **Comment**, enter **`AutoDiscover type2 for autodiscover.`*`company.tld`***
   - **Distribution State**: choose **Enabled**.
8. Choose **Create Distribution**.

3. In the Route 53 console, create a record that routes internet traffic for your domain name to your CloudFront distribution.

   > **Note**
   >
   > These steps assume that the DNS record for example.com is hosted on Route 53. If you don't use Route 53, follow the procedures in your DNS provider's management console.

   1. In the console's navigation pane, choose **Hosted Zones**. and then choose a domain.
   2. In the list of domains, choose the domain name that you want to use.
   3. In **Records**, choose **Create record**.
   4. Under **Quick create record**, set the following parameters:
      - Under **Record Name**, enter a name for the record.
      - Under **Routing policy**, select **Simple routing**.
      - Choose the **Alias** slider to turn it on. The slider turns blue when in the on state.

- In the **Record type** list, choose **A - Routes traffic to an IPv4 address and some AWS resources**.
- In the **Route traffic to** list, choose **Alias to CloudFront distribution**.
- A search box will appear beneath the **Route traffic to** list. Enter your CloudFront distribution's name into the text box. You can also select your distribution from the list that appears when you select the search box.

5. Choose **Create record**.

Use an Apache web server

The following steps explain how to use an Apache web server to create a proxy for https://autodiscover.*company.tld*/autodiscover/autodiscover.xml. To create a proxy for https://*company.tld*/autodiscover/autodiscover.xml, remove the "autodiscover." prefix from the domains in the following steps.

To enable AutoDiscover phase 2 with an Apache web server

1. Run the following directives on an SSL-enabled Apache server:

```
SSLProxyEngine on ProxyPass /autodiscover/autodiscover.xml https://autodiscover-
service.mail.REGION.awsapps.com/autodiscover/autodiscover.xml
```

2. As needed, enable the following Apache modules. If you don't know how, refer to the Apache help:

   - `proxy`
   - `proxy_http`
   - `socache_shmcb`
   - `ssl`

See the following section for information about testing and troubleshooting AutoDiscover.

AutoDiscover phase 2 troubleshooting

Once you've configured your DNS provider for AutoDiscover, you can test your AutoDiscover endpoint configuration. If you've configure your endpoint correctly, it responds with an unauthorized request message.

To make a basic unauthorized request

1. From a terminal, create an unauthenticated POST request to the AutoDiscover endpoint.

```
$ curl -X POST -v https://autodiscover.''company.tld''/autodiscover/
autodiscover.xml
```

   If your endpoint is configured correctly, it should return a `401 unauthorized` message, as shown in the following example:

```
$ curl -X POST -v https://autodiscover.''company.tld''/autodiscover/
autodiscover.xml
...
HTTP/1.1 401 Unauthorized
```

2. Next, test a real AutoDiscover request. Create a `request.xml` file with the following XML content:

```
<?xml version="1.0" encoding="utf-8"?>

<Autodiscover xmlns="http://schemas.microsoft.com/exchange/autodiscover/mobilesync/
requestschema/2006">
    <Request>
        <EMailAddress>testuser@company.tld</EMailAddress>
        <AcceptableResponseSchema>
            http://schemas.microsoft.com/exchange/autodiscover/mobilesync/
responseschema/2006
        </AcceptableResponseSchema>
    </Request>
</Autodiscover>
```

3. Use the `request.xml` file you created and make an authenticated AutoDiscover request to the endpoint. Remember to replace *`testuser@company.tld`* with a valid email address:

```
$ curl -d @request.xml -u testuser@company.tld -v https://autodiscover.company.tld/
autodiscover/autodiscover.xml
```

   The response will look similar to the following example if the endpoint is configured correctly:

```
$ curl -d @request.xml -u testuser@company.tld -v https://autodiscover.company.tld/
autodiscover/autodiscover.xml
```

```
Enter host password for user 'testuser@company.tld':
<?xml version="1.0" encoding="UTF-8"?>
<Autodiscover xmlns="http://schemas.microsoft.com/exchange/autodiscover/
responseschema/2006" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Response xmlns="http://schemas.microsoft.com/exchange/autodiscover/mobilesync/
responseschema/2006">
    <Culture>en:us</Culture>
    <User>
        <DisplayName>User1</DisplayName>
        <EMailAddress>testuser@company.tld</EMailAddress>
    </User>
    <Action>
        <Settings>
            <Server>
                <Type>MobileSync</Type>
                <Url>https://mobile.mail.us-east-1.awsapps.com/Microsoft-Server-
ActiveSync</Url>
                <Name>https://mobile.mail.us-east-1.awsapps.com/Microsoft-Server-
ActiveSync</Name>
            </Server>
        </Settings>
    </Action>
</Response>
```

Editing domain identity policies

Domain identity policies specify permissions for email actions, such as redirecting email messages. For example, you can redirect emails to any email address in your Amazon WorkMail organization.

> **Note**
>
> As of April 1, 2022, Amazon WorkMail began using service principals for authorization instead of AWS account principals. If you added a domain prior to April 1, 2022 you may have an older policy that uses an AWS account principal for authorization. If so, we recommend updating to the latest policy. The steps in this section explain how. Your organization continues to send emails normally during the update.

You only follow these steps if you don't use a custom Amazon SES policy. If you use a custom Amazon SES policy, you must update it yourself. For more information, see Custom Amazon SES service-principal policy, later in this topic.

> **Important**
>
> Don't remove your existing domains. If you do, you'll disrupt mail service. All you need to do is re-enter your existing domains.

To update a domain identity policy

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. To do so, open the **Select a region** list, located to the right of the search box, then choose the desired region. For more information about regions, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In navigation pane, choose **Organizations**, and then choose name of your organization.
3. In the navigation pane, choose **Domains**.
4. Highlight and copy the name of the domain that you want to re-enter, and then choose **Add Domain**.

   The **Add domain** dialog box appears.
5. Paste the copied name into the **Domain name** box, and then choose **Add domain**.
6. Repeat steps 3-5 for the remaining domains in your organization.

Custom Amazon SES service-principal policy

If you use a custom Amazon SES policy, adapt this example for use in your domain.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AuthorizeWorkMail",
```

```
      "Effect": "Allow",
      "Principal": {
        "Service": "workmail.REGION.amazonaws.com"
      },
      "Action": [
        "ses:*"
      ],
      "Resource": "arn:aws:ses:REGION:AWS_ACCOUNT_ID:identity/WORKMAIL-DOMAIN-
NAME",
      "Condition": {
        "ArnEquals": {
          "aws:SourceArn":
 "arn:aws:workmail:REGION:AWS_ACCOUNT_ID:organization/WORKMAIL_ORGANIZATION_ID"
        }
      }
    }
  ]
}
```

Authenticating email with SPF

The Sender Policy Framework (SPF) is an email validation standard designed to combat email spoofing. *Spoofing* is the act of making an email sent by a malicious actor look like one sent by a legitimate user. For information about configuring SPF for your Amazon WorkMail-enabled domain, see Authenticating email with SPF in Amazon SES.

Configuring a custom MAIL FROM domain

By default, Amazon WorkMail uses a subdomain of amazonses.com as the `MAIL FROM` domain for your outgoing email. This can cause delivery failure if the DMARC policy on your domain is only set up for SPF. To resolve this, configure your own domain as the `MAIL FROM` domain. To learn how to set up your email domain as the `MAIL FROM` domain, see Setting up a custom MAIL FROM domain in the *Amazon Simple Email Service Developer Guide*.

> **Important**
>
> A custom MAIL FROM domain is required when you enable AutoDiscover for iOS devices.

For more information about custom `MAIL FROM` domains, see Amazon SES now supports custom MAIL FROM domains.

Working with users

You can create and remove users from Amazon WorkMail. In addition, you can reset their email passwords, manage their mailbox quotas and device access, and control their mailbox permissions.

Topics

- Viewing a list of users
- Adding a user
- Enabling users
- Managing user aliases
- Disabling users
- Editing user details
- Resetting user password
- Troubleshooting Amazon WorkMail password policies
- Working with notifications
- Enabling signed or encrypted email

Viewing a list of users

To view the list of users

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations** and then choose the name of your organization.
3. In the navigation pane, choose **Users**.
4. Additionally, you can filter users by **Username**, **Display name**, or **Primary email address**.

> **Note**
>
> Search is case-sensitive.

Adding a user

When you add a user, Amazon WorkMail automatically creates mailboxes for them. Users can log in and access their mail from the Amazon WorkMail web application, their mobile device, or by using Microsoft Outlook on macOS or PC.

To add a user

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the organization to which you want to add users.
3. In the navigation pane, choose **Users**, and then choose **Add User**.

   The **Add a user** screen appears.
4. Under **User details**, in the **User name** field, enter the user's name. The name also appears in the **Email address** box. If you want the user to have a different email address from their user name, you can edit the **Email address** field.
5. (Optional) Enter the user's first and last name in the **First name** and **Last name** boxes.
6. In the **Display name** box, enter the user's display name.
7. In the **Email address** box, accept the email alias or enter another one.
8. By default, user are displayed in the global address list. To hide the user from the global address list, clear the **Show in global address list** check box.
9. Select **Do not create a mailbox** to add a user as a remote user to the organization.
10. Under **Password setup**, enter the user's password in the **Password** and **Repeat password** boxes.
11. Choose **Add user**.

Enabling users

When you integrate Amazon WorkMail with your corporate Active Directory, or you already have users available in your Simple AD directory, you can enable those users in Amazon WorkMail. You also follow these steps to reenable a user whose account was disabled.

To enable users

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the organization for which you want to enable users.
3. In the navigation pane, choose **Users**.

   A list of users appears. User accounts in the enabled, disabled, and system user states are shown in the list.
4. From the list of users with disabled accounts, select the check boxes for the users that you want to enable, and then choose **Enable**.

   The **Enable users** dialog box appears.
5. As needed, review and change the primary email address for each user, and then choose **Enable**.

Managing user aliases

You can add or remove email aliases to users.

To add an email alias to a user

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, and then choose the name of the organization for which you want to add users.
3. In the navigation pane, choose **Users**, and then select the name of the user to which you want to add an alias.
4. In the **User details** section, choose the **Aliases** tab.
5. Under the **Aliases** tab, choose **Add alias**.
6. In the **Alias** box, enter an alias.
7. Select a domain for an alias.
8. Choose **Add**.

To remove an email alias from a user

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization from which you want to remove users.
3. In the navigation pane, choose **Users**, and then select the name of the user from which you want to remove aliases.
4. In the **User details** section, choose the **Aliases** tab.
5. Under the **Aliases** tab, select the check box against the aliases you want to remove.
6. Verify the aliases that will be removed.
7. On the **Remove aliases** window, choose **Remove**.

Disabling users

You can disable any user in an organization at any time. When you disable a user, it immediately becomes inaccessible. Users that are disabled for longer than 30 days will have their inbox deleted from Amazon WorkMail.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the organization that contains the users that you want to disable.
3. In the navigation pane, choose **Users**.

   A list of all users appears, showing accounts that are in the enabled, disabled, and system user states.
4. From the list of enabled users, select the check boxes for the accounts that you want to disable, and then choose **Disable**.

   The **Disable users** dialog box appears.
5. Choose **Disable**.

Editing user details

When you edit the user details, you can change the following:

- **Personal data** – Names, email address, phone numbers, and other personal details.
- **Mailbox quotas (sizes)** – Quotas can range from 1 MB to 51,200 MB (50 GB). Amazon WorkMail notifies users when they reach 90 percent of their quota. Also, changing a user's mailbox quota won't affect pricing. For more information about pricing, refer to Amazon WorkMail Pricing.
- **Mobile device access** – Remove and wipe devices, and view device details.
- **Mailbox access permissions** – Grant users permission to use a mailbox, and grant users different levels of access to the mailbox.
- **Personal access tokens (when IAM Identity Center is enabled)** – View and delete personal access tokens.

> **Note**
>
> If you integrate Amazon WorkMail with an AD Connector directory, you can't edit these details from the AWS Management Console. Instead, you must edit them using your

Active Directory management tools. Limitations apply when your organization is in interoperability mode. For more information, see Limitations in interoperability mode.

To edit the user details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the organization that you want to use.
3. In the navigation pane, choose **Users**, and then choose the name of the user to edit.

To edit personal data

1. In the **User details** section, choose **Edit**.
2. Under **User details**, enter or change the user's personal information as needed.
3. When finished, choose **Save changes**.

To associate with an IAM Identity Center user

1. Under **User details**, choose **Edit**.
2. Enter the user ID of the IAM Identity Center user you want to associate. You can view this information under the **Assigned Users** table in the IAM Identity Center page or in the IAM Identity Center console.
3. Choose **Save changes**.

To edit mailbox quotas

1. Under **User details**, choose the **Quota** tab, and then choose **Edit**.
2. In the **Update mailbox quota** box, enter a size for the mailbox. You can enter values from **1** to **51200**.
3. Choose **Save changes**.

To manage mobile device data

> **Note**
>
> To manage mobile devices, your users first need to connect their devices to your instance of Amazon WorkMail. For information about connecting mobile devices, refer to Setting up mobile device clients for Amazon WorkMail.

1. Under **User details**, choose the **Mobile devices** tab.
2. To see a current list of devices, choose **Refresh**.
3. To view a device's details, choose the device name from the **Device ID** column.
4. To remove or wipe the device, choose the radio button next to the device name, and then choose **Remove** or **Wipe** as needed.
5. In the dialog box that appears, confirm the removal or wipe operation. Remember that users will reappear when they sync their devices with Amazon WorkMail again.

To edit mailbox permissions

1. Choose the **Permissions** tab.
2. Do one of the following:

    1. To add permissions, choose **Add permissions**. Open the **Add new permissions** list and choose a user or group, choose the permission settings for the user or group, and then choose **Save**.
    2. To edit user permissions, choose the button next to the user's name. Choose **Edit**, select the desired options, and then choose **Save**.

    For more information about the permission options, refer to Working with mailbox permissions.
3. To remove all permissions, choose **Remove**, then confirm the removal.

To delete personal access tokens

> **Note**
>
> Make sure the token you are deleting is not actively used by any email client. Deleting a token when in use will break the authentication for the clients using the token.

1. Choose the **Personal Access Tokens** tab.
2. From the list of personal access tokens, select the personal access token to delete.
3. Choose **Delete token**.
4. Enter **Type** in the confirmation text box.

Resetting user password

If a user forgets their password or has trouble signing in to Amazon WorkMail, you can reset their password.

> **Note**
>
> - If you've integrated Amazon WorkMail with an AD Connector directory, you must reset the user password in Active Directory.
> - If you've integrated Amazon WorkMail with IAM Identity Center, you can choose to reset the user password. For more information, see Reset the IAM Identity Center user password for an end user in the *AWS IAM Identity Center User Guide*.

To reset a user password

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.

3. In the navigation pane, choose **Users**.
4. In the list of users, select the check box next to the name of the user, and then choose **Reset password**.
5. In the **Reset Password** dialog box, enter the new password, and then choose **Reset**.

Troubleshooting Amazon WorkMail password policies

If resetting the password is unsuccessful, verify that the new password meets the password policy requirements.

The password policy requirements depend on which directory type your Amazon WorkMail organization uses.

Amazon WorkMail directory and Simple AD directory password policy

By default, passwords for an Amazon WorkMail directory or Simple AD directory must be:

- Non-empty
- At least eight characters
- Less than 64 characters
- Composed of Basic Latin or Latin-1 supplement characters

Passwords must also contain characters from three out of five of the following groups:

- Uppercase characters
- Lowercase characters
- Numerical digits (0 through 9)
- Special characters (for example, **<**, **~**, or **!**)
- Latin-1 supplement characters (for example, **é**, **ü**, or **ñ**)

Amazon WorkMail directory password policies can't be changed.

To change a Simple AD password policy, use the AD administration tools on an Amazon Elastic Compute Cloud (Amazon EC2) Windows instance of your Simple AD directory. For more

information, see Installing the Active Directory administration tools in the *AWS Directory Service Administration Guide*.

AWS Managed Microsoft AD Directory password policy

For information about the default password policy for an AWS Managed Microsoft AD directory, see Manage Password Policies for AWS Managed Microsoft AD in the *AWS Directory Service Administration Guide*.

AD Connector password policy

AD Connector uses the password policy of the Active Directory domain that it is connected to. See the documentation for your Active Directory domain for more information on password policy settings.

Working with notifications

With the Amazon WorkMail Push Notifications API, you can receive push notifications about changes in your mailbox, including new email and calendar updates. You must register the URLs (or push notification responders) to receive notifications. With this feature, developers can create responsive applications for Amazon WorkMail users, as applications are quickly notified about changes from a user's mailbox.

For more information, see Notification subscriptions, mailbox events, and EWS in Exchange.

You can subscribe to specific folders, such as Inbox or Calendar, or to all folders for mailbox change events (including New Mail, Created, and Modified).

You can use client libraries such as the EWS Java API or the Managed EWS C# API to access this feature. A complete sample application of a push responder, developed using AWS Lambda and API Gateway (using the AWS Serverless framework), is available on the AWS GitHub page. It uses the EWS Java API.

The following is a sample push subscription request:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:t="http://
schemas.microsoft.com/exchange/services/2006/types">
   <soap:Body>
      <m:Subscribe xmlns:m="http://schemas.microsoft.com/exchange/services/2006/
messages">
         <m:PushSubscriptionRequest>
```

```
            <t:FolderIds>
               <t:DistinguishedFolderId Id="inbox" />
            </t:FolderIds>
            <t:EventTypes>
               <t:EventType>NewMailEvent</t:EventType>
               <t:EventType>CopiedEvent</t:EventType>
               <t:EventType>CreatedEvent</t:EventType>
               <t:EventType>DeletedEvent</t:EventType>
               <t:EventType>ModifiedEvent</t:EventType>
               <t:EventType>MovedEvent</t:EventType>
            </t:EventTypes>
            <t:StatusFrequency>1</t:StatusFrequency>
            <t:URL>https://YOUR_PUSH_RESPONDER_URL</t:URL>
         </m:PushSubscriptionRequest>
      </m:Subscribe>
   </soap:Body>
</soap:Envelope>
```

The following is a successful subscription request result:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance">
   <Header xmlns="http://schemas.xmlsoap.org/soap/envelope/">
      <ServerVersionInfo xmlns="http://schemas.microsoft.com/exchange/
services/2006/types" MajorVersion="14" MinorVersion="2" MajorBuildNumber="390"
 Version="Exchange2010_SP2" MinorBuildNumber="3" />
   </Header>
   <soap:Body>
      <m:SubscribeResponse xmlns:m="http://schemas.microsoft.com/exchange/
services/2006/messages" xmlns:t="http://schemas.microsoft.com/exchange/services/2006/
types">
         <m:ResponseMessages>
            <m:SubscribeResponseMessage ResponseClass="Success">
               <m:ResponseCode>NoError</m:ResponseCode>
               <m:SubscriptionId>hKJETtoAdi9PPW0tZDQ4MThmMDoVYB</m:SubscriptionId>
               <m:Watermark>AAAAAAA=</m:Watermark>
            </m:SubscribeResponseMessage>
         </m:ResponseMessages>
      </m:SubscribeResponse>
   </soap:Body>
</soap:Envelope>
```

Afterwards, notifications are sent to the URL specified in the subscription request. The following is a sample notification:

```
<soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Header>
        <t:RequestServerVersion
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
 Version="Exchange2010_SP2">
        </t:RequestServerVersion>
    </soap:Header>
    <soap:Body>
        <m:SendNotification
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
            <m:ResponseMessages>
                <m:SendNotificationResponseMessage ResponseClass="Success">
                    <m:ResponseCode>NoError</m:ResponseCode>
                    <m:Notification>
                        <t:SubscriptionId>hKJETtoAdi9PPW0tZDQ4MThmMDoVYB</
t:SubscriptionId>
                        <t:PreviousWatermark>ygwAAAAAAAA=</t:PreviousWatermark>
                        <t:MoreEvents>false</t:MoreEvents>
                        <t:ModifiedEvent>
                            <t:Watermark>ywwAAAAAAAA=</t:Watermark>
                            <t:TimeStamp>2018-02-02T15:15:14Z</t:TimeStamp>
                            <t:FolderId Id="AAB2L089bS1kNDgxOGYwOGE5OTQ0="></
t:FolderId>
                            <t:ParentFolderId Id="AAB2L089bS1kNDgxOGYwOGE="></
t:ParentFolderId>
                        </t:ModifiedEvent>
                    </m:Notification>
                </m:SendNotificationResponseMessage>
            </m:ResponseMessages>
        </m:SendNotification>
    </soap:Body>
</soap:Envelope>
```

To acknowledge that the push notification responder has received the notification, it must reply with the following:

```
<?xml version="1.0"?>
```

```
  <s:Envelope xmlns:s= "http://schemas.xmlsoap.org/soap/envelope/">
    <s:Body>
      <SendNotificationResult xmlns="http://schemas.microsoft.com/exchange/
services/2006/messages">
        <SubscriptionStatus>OK</SubscriptionStatus>
      </SendNotificationResult>
    </s:Body>
  </s:Envelope>
```

To unsubscribe from receiving push notifications, clients must send an unsubscribe response in the `SubscriptionStatus` field, similar to the following:

```
<?xml version="1.0"?>
  <s:Envelope xmlns:s= "http://schemas.xmlsoap.org/soap/envelope/">
    <s:Body>
      <SendNotificationResult xmlns="http://schemas.microsoft.com/exchange/
services/2006/messages">
        <SubscriptionStatus>Unsubscribe</SubscriptionStatus>
      </SendNotificationResult>
    </s:Body>
  </s:Envelope>
```

To verify the health of your push notification responder, Amazon WorkMail sends a "heartbeat" (also called a `StatusEvent`). The frequency with which they are sent is determined by the `StatusFrequency` parameter provided in the initial subscription request. For example, if `StatusFrequency` equals **1**, a `StatusEvent` is sent every 1 minute. This value can range between 1 and 1440 minutes. This `StatusEvent` looks like the following:

```
<?xml version="1.0 (http://www.w3.org/TR/REC-xml/)" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Header>
    <t:RequestServerVersion xmlns:t="http://schemas.microsoft.com/exchange/
services/2006/types" xmlns:m="http://schemas.microsoft.com/exchange/services/2006/
messages" Version="Exchange2010_SP2"/>
</soap:Header>
<soap:Body>
    <m:SendNotification xmlns:t="http://schemas.microsoft.com/exchange/services/2006/
types" xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
    <m:ResponseMessages>
        <m:SendNotificationResponseMessage ResponseClass="Success">
            <m:ResponseCode>NoError</m:ResponseCode>
            <m:Notification>
```

```
                <t:SubscriptionId>hKJETtoAdi9PPW0tZDQ4MThmMDoVYB</t:SubscriptionId>
                <t:PreviousWatermark>AAAAAAAAAAA=</t:PreviousWatermark>
                <t:MoreEvents>false</t:MoreEvents>
                <t:StatusEvent>
                    <t:Watermark>AAAAAAAAAAA=</t:Watermark>
                </t:StatusEvent>
            </m:Notification>
        </m:SendNotificationResponseMessage>
    </m:ResponseMessages>
</m:SendNotification>
</soap:Body>
</soap:Envelope>
```

If a client push notification responder fails to respond with the same OK status as before, the notification is retried for a maximum of `StatusFrequency` minutes. For example, if `StatusFrequency` equals 5, and the first notification fails, it is retried for a maximum of 5 minutes with an exponential backoff between each retry. If the notification is not delivered after the retry time has expired, the subscription is invalidated and no new notifications are delivered. You must create a new subscription to continue to receive notifications about mailbox events. Currently, you can subscribe for a maximum of three subscriptions per mailbox.

Enabling signed or encrypted email

You can use S/MIME to enable users to send signed or encrypted email both inside and outside of the organization.

> **Note**
>
> User certificates in the Global Address List (GAL) are supported only in a connected Active Directory setup.

To enable users to send signed or encrypted emails

1. Set up an Active Directory (AD) Connector. Setting up an AD Connector with your on-premises directory allows users to continue to use their existing corporate credentials.
2. Configure Certificate Autoenrollment to issue and store user certificates automatically in the Active Directory. Amazon WorkMail receives user certificates from the Active Directory and publishes them to the GAL. For more information, see Configure Certificate Autoenrollment.

3. Distribute the generated certificates to users by exporting the certificates from the server running Microsoft Exchange and mailing them.
4. Each user installs the certificate to their email program (such as Windows Outlook) and mobile devices.

Working with groups

You can use groups as distribution lists in Amazon WorkMail for receiving emails for generic email addresses, such as <sales@example.com> or <support@example.com>. You can create multiple email aliases for a group.

You can also use groups as security groups to share a mailbox or calendar with a certain team.

Groups don't have their own mailboxes, and that affects the mailbox permissions that you can grant to a group. For information about setting up mailbox permissions for a group, see Managing mailbox permissions for groups.

> **Note**
>
> It can take up to 2 hours before newly added groups appear in your Microsoft Outlook offline address book.

Topics

- Viewing a list of groups
- Adding a group
- Enabling groups
- Adding members to a group
- Editing group details
- Removing members from a group
- Managing group aliases
- Disabling groups
- Deleting a group

Viewing a list of groups

To view the list of groups

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations** and then choose the name of your organization.
3. In the navigation pane, choose **Groups**.
4. Additionally, you can filter groups by **Group name** or **Primary email address**.

> **Note**
>
> Search is case-sensitive.

Adding a group

You can add groups from the Amazon WorkMail console.

To add a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Groups**, and then choose **Add group**.

   The **Add group** page appears.
4. Under **Group name**, enter a name for the group.
5. Under **Email address**, enter the primary email address for the group.
6. Verify the group's email address, update as required.
7. By default, the group is displayed in the global address list. To hide the group from the global address list, clear the **Show in global address list** check box.
8. Choose **Add group**.

Enabling groups

When you integrate Amazon WorkMail with your corporate Active Directory, or you already have groups available in your simple Active Directory, you can use those groups as security groups or distribution lists in Amazon WorkMail.

To enable an existing directory group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Groups**.
4. Choose the check box next to the group that you want to enable, and then choose **Enable**.

   The **Enable groups** dialog box appears and asks you to confirm the operation.
5. As needed, review and change the primary email address for each group, and then choose **Enable**.

Adding members to a group

After you create and enable an Amazon WorkMail group, use the Amazon WorkMail console to add members to that group.

> **Note**
>
> If Amazon WorkMail is integrated with a connected Active Directory service or Microsoft Active Directory, you can use Active Directory to manage your group members. However, changes can take longer to propagate to Amazon WorkMail.

To add members to a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Groups**.
4. Select the name of the group.
5. On the **Group details** page, choose the **Members** tab.
6. Choose a group or user to add under **Group or User**.
7. Select the user or group from the drop-down.
8. Choose **Save**.

Your changes can take a few minutes to propagate.

Editing group details

You can edit the details of a group.

To edit group details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Groups**, and then select the group to edit.
4. On the **Group details** page, update the **Email address** as needed.
5. By default, groups are displayed in the global address list. To hide the group from the global address list, clear the **Show in global address list** check box.
6. Choose **Save changes**.

Removing members from a group

Use the Amazon WorkMail console to remove members from a group.

> **Note**
>
> If Amazon WorkMail is integrated with a connected Active Directory or Microsoft Active Directory, you can use Active Directory to manage your group members. However, doing so can create the time needed to propagate your changes to Amazon WorkMail.

To remove members from a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Groups**, then choose the name of the group.
4. On the **Group details** page, choose the **Members** tab.
5. Select the member to remove from the group.
6. Choose **Remove**.

Your changes can take a few minutes to propagate.

Managing group aliases

You can add or remove email aliases to groups.

To add an email alias to a group.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, and then choose the name of the organization for which you want to add an alias.
3. In the navigation pane, choose **Groups**, and then select the name of the group to which you want to add an alias.
4. In the **Group details** section, choose **Aliases**.
5. Under **Aliases**, choose **Add alias**.
6. In the **Alias** box, enter an alias.
7. Select a domain for an alias.
8. Choose **Add**.

To remove an email aliases from a group.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization from which you want to remove an alias.
3. In the navigation pane, choose **Groups**, and then select select the name of the group from which you want to remove aliases.
4. In the **Group details** section, choose **Aliases**.
5. Under **Aliases**, select the check box against the aliases you want to remove.
6. Choose **Remove**.
7. Verify the aliases that will be removed.
8. On the **Remove aliases** window, choose **Remove**.

Disabling groups

When you no longer need a group, you can disable it.

To disable a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Groups**.
4. Under **Group name**, select the groups to disable, and then choose **Disable**.
5. In the **Disable group(s)** dialog box, choose **Disable**.

Deleting a group

Before you can delete a group, you must first disable that group. For information about disabling groups, see Disabling groups.

To delete a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Groups**.
4. Select the check box next to the disabled group that you want to delete and choose **Delete**.

   The **Delete** dialog box appears.
5. In the **Enter the group name to confirm deletion** box, enter the name of the group, then choose **Delete** .

> **Note**
>
> To permanently delete a group, use the `DeleteGroup` API action for Amazon WorkMail. For more information, see DeleteGroup in the *Amazon WorkMail API Reference*.

Working with resources

Amazon WorkMail can help your users reserve resources. For example, users can reserve meeting rooms, or equipment such as projectors, phones, or cars. To book a resource, the user adds the resource to the meeting invitation.

Topics

- Viewing a list of resources
- Adding a resource
- Editing resource details
- Managing resource aliases
- Enabling a resource
- Disabling a resource
- Deleting a resource

Viewing a list of resources

To view the list of resources

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations** and then choose the name of your organization.
3. In the navigation pane, choose **Resources**.
4. Additionally, you can filter resources by **Resource name** or **Primary email address**.

> **Note**
>
> Search is case-sensitive.

Adding a resource

You can add a new resource to your organization and allow your users to reserve it.

To add a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Resources**, and then **Add resource**.

   The **Add resource** page appears.
4. In the **Resource name** box, enter a name for the resource.
5. Optionally, in the **Resource description** box, enter a description for the resource.
6. Under **Resource type**, choose an option.
7. Verify the resource's email address, update as required.
8. By default, the resource is displayed in the global address list. To hide the resource from the global address list, clear the **Show in global address list** check box.
9. Choose **Add resource**.

Editing resource details

You can edit a resource's general details, including name, description, type, and email address, booking options, and delegates.

To edit general resource details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Resources**, and then select the resource to edit.
4. On the **Resource details** page, update the **Resource name**, **Description**, **Resource Type**, or **Email address** as needed.
5. By default, resources are displayed in the global address list. To hide the resource from the global address list, clear the **Show in global address list** check box.
6. Choose **Save changes**.

You can configure a resource to accept or decline booking requests automatically.

You can edit the resource's booking options.

To change a resource's booking options

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Resources**, and then select the resource to edit. A page appears and displays **Resource details**.
4. Under **Booking options** choose **Edit**.
5. As required, select or clear the check box next to an option to enable or disable the option.

> **Note**
>
> When you disable any of the automatic booking options, you must create a delegate to handle the booking requests. The next steps explain how to create delegate.

You can add a delegate to control booking requests for a resource that doesn't have automatic booking options configured. Resource delegates automatically receive copies of all booking requests and have full access to the resource calendar. In addition, they must accept all booking requests for a resource.

To add a resource delegate

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, choose **Resources**, and then select the name of the resource to which you want to add a delegate.
4. (Optional) In the **Booking options** tab, choose **Edit**, clear the **Automatically accept all resource requests** check box, and then choose **Save**.
5. Choose the **Delegates** tab, and then choose **Add delegate**.

   The **Add delegate** dialog box appears.
6. Open the **Search delegates** list and choose a delegate, then choose **Save**.

To remove a resource delegate

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization from which you want to remove delegates.
3. In the navigation pane, choose **Resources**, and then select the name of the resource from which you want to remove a delegate.
4. Choose **Delegates**, and then choose the delegate to remove.
5. Chooose **Remove**.

Managing resource aliases

You can add or remove email aliases to resources.

To add an email alias to a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization to which you want to add an alias.
3. In the navigation pane, choose **Resources**, and then select the name of the resource to which you want to add an alias.
4. In the **Resource details** section, choose **Aliases**.
5. Under **Aliases**, choose **Add alias**.
6. In the **Alias** box, enter an alias.
7. Select a domain for an alias.
8. Choose **Add**.

To remove an email aliases from a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization from which you want to remove aliases.
3. In the navigation pane, choose **Resources**, and then select the name of the resource from which you want to remove aliases.
4. In the **Resource details** section, choose **Aliases**.
5. Under **Aliases**, select the check box against the aliases you want to remove.
6. Choose **Remove**.
7. Verify the aliases that will be removed.
8. On the **Remove aliases** window, choose **Remove**.

Enabling a resource

By default, Amazon WorkMail creates a resource. If you or someone else disables a resource, you can reenable the resource within 30 days.

To enable a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information about regions, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the organization which contains the resource that you want to enable.
3. In the navigation pane, choose **Resources**.
4. In the list of resources, select the button next to the resource that you want to enable, and then choose **Enable**.

   The **Enable resource** dialog box appears.
5. Choose **Enable**.

Disabling a resource

When you disable a resource, you make it unavailable for booking. For example, you can disable a conference room while it's being remodeled, then enable the room once it's available for use.

To disable a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information about regions, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the organization which contains the resource that you want to disable.
3. In the navigation pane, choose **Resources**.

4. In the list of resources, select the button next to the resource that you want to disable, and then choose **Disable**.

   The **Disable resource** dialog box appears.
5. Choose **Disable**.

Deleting a resource

When you no longer need a resource, you can delete it. However, you must first disable the resource. For information about disabling a resource, see the steps in the previous section.

To remove a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information about regions, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the desired organization.
3. In the navigation pane, choose **Resources**.
4. In the list of resources, select the button next to the disabled resource that you want to remove, and then choose **Delete**.

   The **Delete resource** dialog box appears.
5. In the **Enter the resource name to confirm deletion** box, enter the name of the resource that you want to delete, and then choose **Delete resource**.

Working with IAM Identity Center

You can enable multi-factor authentication (MFA) in Amazon WorkMail by associating your Amazon WorkMail users with IAM Identity Center. For more information, see What is IAM Identity Center.

The table below describes the steps to address different scenarios.

Scenario	Steps
Associating Amazon WorkMail users to IAM Identity Center	1. Enabling IAM Identity Center in Amazon WorkMail 2. Assigning IAM Identity Center users and groups to Amazon WorkMail application 3. Associating Amazon WorkMail users with IAM Identity Center users
Existing Amazon WorkMail users	1. Create IAM Identity Center users with the same username, group the users together and assign the group to the Amazon WorkMail application. 2. Associate the Amazon WorkMail users to the IAM Identity Center users.
Existing IAM Identity Center users	1. Create Amazon WorkMail users with the same username as the IAM Identity Center users. 2. Assign the IAM Identity Center users or groups to the Amazon WorkMail applicati on. 3. Associate the Amazon WorkMail users to IAM Identity Center users.
Connecting an external directory to IAM Identity Center	1. Sync the external directory users to the IAM Identity Center group. For more informati on, see IAM Identity Center Identity source tutorials

Scenario	Steps
	2. Assign the IAM Identity Center group to the Amazon WorkMail application. 3. Connect the external directory to Amazon WorkMail and make sure the user names match 4. Associate the Amazon WorkMail users to the IAM Identity Center users.

Once the above steps are completed you can view the IAM Identity Center status, link to the AWS IAM Identity Center to manage users and groups, MFA enabled Amazon WorkMail web application URL, authentication mode, personal access token status and timeline under IAM Identity Center under **Settings** in the Amazon WorkMail console. For more information on managing MFA in the IAM Identity Center console, see Multi-factor authentication for IAM Identity Center users .

> **Note**
>
> Make sure the configuration between Amazon WorkMail and IAM Identity Center is well tested and verified. Users could lose access to their mailboxes when the configuration is not correct and complete.

Topics

- Enabling IAM Identity Center in Amazon WorkMail
- Assigning IAM Identity Center users and groups to Amazon WorkMail application
- Associating Amazon WorkMail users with IAM Identity Center users
- Authentication mode
- Configuring personal access tokens
- Disabling IAM Identity Center

Enabling IAM Identity Center in Amazon WorkMail

When you enable IAM Identity Center, it acts as an authentication layer for the Amazon WorkMail users. IAM Identity Center users are managed separately from the Amazon WorkMail directory. It is recommended to use the same usernames across IAM Identity Center and Amazon WorkMail.

> **Note**
>
> Make sure Amazon WorkMail and IAM Identity Center are setup in the same region.

To enable IAM Identity Center, follow these steps.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Identity Center**.

   The **IAM Identity Center Settings** page appears.
3. Choose **Enable**.

   The **Enable IAM Identity Center** window appears.
4. Choose **Enable**.

   The **Identity Center Settings** page appears with the **Identity Center Status** displayed.
5. To add IAM Identity Center users and groups to your Amazon WorkMail Organization, follow the link under **Identity Center status**. For information on how to add users and groups, see Manage identities in IAM Identity Center..

Assigning IAM Identity Center users and groups to Amazon WorkMail application

When you enable IAM Identity Center in Amazon WorkMail, WorkMail creates an application in IAM Identity Center on your behalf. By default, IAM Identity Center users must be assigned to this

application or belong to a group which is assigned to this application in order to access a mailbox in the Amazon WorkMail organization. For more information, see AWS managed applications in the AWS IAM Identity Center User Guide.

You can assign IAM Identity Center users and groups to Amazon WorkMail in the following ways:

- By IAM Identity Center users – You can assign IAM Identity Center users to Amazon WorkMail.
- By IAM Identity Center group – You can assign IAM Identity Center groups to Amazon WorkMail. By adding a group, all users under a group will have access to Amazon WorkMail.

  For more information on adding users and groups, see Users, groups, and provisioning in IAM Identity Center .

Note

If you are connecting your existing identity source with IAM Identity Center, review the following before changing your directory source.

- Your authentication is being managed by IAM Identity Center.
- Amazon WorkMail will retain all Amazon WorkMail users and groups.
- IAM Identity Center will retain all IAM Identity Center users, groups, and assignments.
- You must manage Amazon WorkMail users and groups in Amazon WorkMail console.
- You must manage IAM Identity Center users and groups in IAM Identity Center.
- Users without an IAM Identity Center assignment or user association cannot access Amazon WorkMail.
- You must manage MFA policy controls in IAM Identity Center.
- When you change the IAM Identity Center source to and from Manage Active Directory in IAM Identity Center you must disable the existing IAM Identity Center configurations in Amazon WorkMail and reconfigure to associate your Amazon WorkMail users with IAM Identity Center.

Users and groups synced with your IAM Identity Center directory are available to assign to your Amazon WorkMail application. For more information about IAM Identity Center user and group management, see Get started with common tasks in IAM Identity Center..

To assign IAM Identity Center users and groups to Amazon WorkMail, follow these steps.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Identity Center**.

   The **IAM Identity Center Settings** page appears.

3. Choose **Assign users and groups**.

   You can add and assign new users or assign existing users and groups.

   - Assign Users – You can assign individual IAM Identity Center users to the Amazon WorkMail. You can either create a new IAM Identity Center user or search for an existing user.
   - Assign Groups – You can also assign an IAM Identity Center group to Amazon WorkMail. All members of the group will then be assigned to Amazon WorkMail.

> **Note**
>
> All new IAM Identity Center users are enabled by default in IAM Identity Center. To grant access to Amazon WorkMail, you must set their password in IAM Identity Center and assign them to Amazon WorkMail. For more information, see Add users to your Identity Center directory .

Associating Amazon WorkMail users with IAM Identity Center users

When a user signs in to the Amazon WorkMail web client with their IAM Identity Center user credentials, the client will open the mailbox of the associated Amazon WorkMail user. If no user in the WorkMail organization is associated with the IAM Identity Center user, WorkMail will create an association between the IAM Identity Center user signing in and the WorkMail user having the same username, if such a WorkMail user exists. Otherwise, the client will display an error message to the user.

> **Note**
>
> You are recommended to use the same username for a user across Amazon WorkMail and IAM Identity Center because WorkMail will create the association automatically when the user first signs in to the Amazon WorkMail web client with their IAM Identity Center user credentials. When the usernames are different, you are responsible to create the association.

To associate users, follow these steps.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Region and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Identity Center**.

   The **IAM Identity Center Settings** page appears.
3. Choose **Associate users**.
4. Under **Select a WorkMail user**, select the Amazon WorkMail user you wish to associate.
5. Under **Enter the IAM Identity Center user ID**, enter the ID of the IAM Identity Center user you wish to associate. You may copy the ID from the **Assigned users** tab on the **Identity Center** page.

   > **Note**
   >
   > The IAM Identity Center user must be authorized to access the Amazon WorkMail application.
6. Choose **Associate users**.

   Once the association is successful, the Amazon WorkMail user can log into Amazon WorkMail using the MFA IAM Identity Center credentials.

> **Note**
>
> You can also associate Amazon WorkMail users with IAM Identity Center users when you edit the Amazon WorkMail user details. For more information, see Editing user details.

Authentication mode

You can use authentication mode to allow users to log in using either their Amazon WorkMail directory credentials, their IAM Identity Center credentials, or restricting login to only IAM Identity Center credentials.

There are two authentication modes available in Amazon WorkMail.

> **Note**
>
> The choice of authentication mode depends on your organization's security requirements and user experience preferences. It is recommended to use *IAM Identity Center only* mode as it provides enhanced security by enforcing IAM Identity Center credentials and MFA. However, before switching from the *Amazon WorkMail Directory and IAM Identity Center* mode, make sure to test the MFA process with all your users to ensure a smooth transition and avoid any impact on existing email client access.

- **Amazon WorkMail Directory and IAM Identity Center (recommended for testing)** – This is the default option for you to test the IAM Identity Center associations before switching to production mode. Test mode allows users to log into the Amazon WorkMail web client using both the Amazon WorkMail directory and IAM Identity Center credentials. When you share the Amazon WorkMail web application URL from the *Organization* settings, your user can log in using their Amazon WorkMail directory credentials. When you share the MFA-enabled URL from the IAM Identity Center settings, you user can log in using their IAM credentials.
- **IAM Identity Center only (recommended for production)** – This authentication mode only allows you to login into the Amazon WorkMail client mailbox using the IAM Identity Center credentials. For any existing Amazon WorkMail users, the Amazon WorkMail directory credentials are no longer valid for both the Amazon WorkMail web application and any existing email clients. You can request a personal access token to access the mailbox using any email clients. To avoid losing access to mailboxes, make sure MFA is enabled for all Amazon WorkMail users.

To enable authentication mode, follow these steps.

1. Under the **Identity Center Settings** page, choose the **Authentication Mode** tab.
2. Choose **Edit**.

   The **Edit authentication mode** page appears.
3. Select one of the following:

   - *IAM Identity Center only*
   - *Amazon WorkMail Directory and IAM Identity Center*
4. Choose **Save**.

Configuring personal access tokens

You can enable personal access token for Amazon WorkMail users to access their mailboxes using desktop and mobile email clients. After IAM Identity Center is enabled, by default, the personal access token status is set to active and is valid for 365 days. After enabling IAM Identity Center, your users' existing credentials will no longer be valid to log into their email clients. Your users can generate the personal access token from the Amazon WorkMail web application and use it to log into any email clients. You can edit the personal access token expiration and when the token expires, your user can generate a new one.

Note

- Your user can only view and copy your personal access token once when you create them in Amazon WorkMail. If you lose your personal access token, you will need to generate a new one for security reasons.
- Amazon WorkMail only allows personal access tokens for mailbox access when the Amazon WorkMail user is associated with an IAM Identity Center user who is authorized to access the Amazon WorkMail application.

The personal access token configurations are listed below:

- Active – When the personal access token status set to *Active*, your user can generate personal access token from Amazon WorkMail and use it to log in to any email client within the token's lifetime.

- Inactive – When the personal access token status is set to *Inactive*, your user will not be able to generate or use personal access tokens to access mailboxes.
- Token lifetime – By default, the personal access token is valid for 365 days. You have the option to change the personal access token lifetime. When you leave the lifetime setting blank, the token will have an indefinite lifetime and never expire.

To configure personal access tokens, follow these steps.

1. Under the **Identity Center Settings** page, choose the **Personal access token configuration** tab.
2. Choose **Edit**.

   The **Edit personal token configuration** page appears.
3. Under **Token status**, slide the **Active** button to enable personal access token.
4. In the **Token lifetime (in days)** text box, enter the number of days the personal access token can be activated.
5. Choose **Save**.

Disabling IAM Identity Center

You can disable IAM Identity Center from the Amazon WorkMail console. Once disabled, you cannot access the mailbox using the IAM Identity Center credentials or personal access tokens. It is recommended to reset all user passwords and the Amazon WorkMail users will revert to using the Amazon WorkMail Directory credentials.

> **Note**
>
> Check the following:
>
> - After disabling IAM Identity Center, your Amazon WorkMail and IAM Identity Center users and groups will remain unchanged.
> - The existing user associations will continue to exist.
> - Your authentication will revert to being managed by Amazon WorkMail directory, instead of IAM Identity Center.

To disable IAM Identity Center, follow these steps.

1. Under the **Identity Center Settings** page, choose **Disable**.

   The **Disable IAM Identity Center** page appears.
2. Choose **Confirm**.

Working with mobile devices

The topics in this section explain how to manage mobile devices connected to Amazon WorkMail.

Topics

- Editing your organization's mobile device policy
- Managing mobile devices
- Managing mobile device access rules
- Managing mobile device access overrides
- Integrating with mobile device management solutions

Editing your organization's mobile device policy

You can edit your organization's mobile device policy to change the way that mobile devices interact with Amazon WorkMail.

To edit your organization's mobile device policy

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a region** list and choose a Region. For more information, see Region Name and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Mobile Policies**, and then on the **Mobile policy** screen, choose **Edit**.
4. Update any of the following as necessary:

   a. **Require encryption on device**: Encrypt email data on the mobile device.

   b. **Require encryption on storage card**: Encrypt email data on the mobile device's removable storage.

   c. **Password required**: Require a password to unlock a mobile device.

   d. **Allow simple password**: Use the device's PIN as the password.

e. **Minimal password length**: Set the number of characters required for a valid password.

f. **Require alphanumeric password:** Require passwords that consist of letters and numbers.

g. **Number of failed attempts allowed**: Specify the number of failed device unlock attempts that are allowed before the user's device is wiped. All data, including personal files will be deleted when the device is wiped.

h. **Password expiration**: Specify the number of days before a password expires and must be changed.

i. **Enable screen lock**: Specify the number of seconds that must elapse without user input to lock the user's screen.

j. **Enforce password history**: Specify the number of passwords that can be entered before repeating the same password.

5. Choose **Save**.

Managing mobile devices

The topics in this section explain how to remotely wipe mobile devices, remove devices from your organization, and view the details for devices. For information about editing your organization's mobile device policy, see Editing your organization's mobile device policy.

Topics

- Remotely wiping mobile devices
- Removing user devices from the devices list
- Viewing mobile device details

Remotely wiping mobile devices

The steps in this section explain how to remotely wipe mobile devices. Remember the following:

- Devices must be online and connected to Amazon WorkMail. If someone disconnects the device, the wipe operation resumes when the user reconnects the device.
- Wipe operations can take five minutes to propagate.

> **Important**
>
> For most mobile devices, a remote wipe resets the device to factory defaults. All data, including personal files, can be removed when you perform this procedure.

To remotely wipe a user's mobile device

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a region** list and choose a Region. For more information, see Region Name and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Users**, and in the list of users, select the name of the user whose device you need to wipe.
4. Choose the **Mobile devices** tab.
5. In the list of devices, choose the button next to the device, and then choose **Wipe**.
6. Check the status in the overview to see whether the wipe is requested.
7. After the device is wiped, remove it from the devices list. The steps in the next section explain how.

   > **Important**
   >
   > To return a wiped device to a user's list of devices, make sure you first remove it from the devices list. Otherwise, the system wipes the device again.

Removing user devices from the devices list

If someone stops using a specific mobile device, or you've remotely wiped the device, you can remove the device from the devices list. When the user configures the device again, it shows up in the list.

To remove a user's mobile devices from the devices list

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Users**, and then select the user's name.
4. Choose the **Mobile devices** tab.
5. In the list of devices, select the button next to the device and choose **Remove**.

Viewing mobile device details

You can you view the details of a user's mobile device.

> **Note**
>
> Some devices don't send all their details to the server. You may not see all available device details.

To view device details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of your organization.
3. In the navigation pane, choose **Users**, and then choose the **Mobile devices** tab.
4. In the list of devices, select the ID of the device for which you want to view details.

   The following table lists the device status codes.

Status	Description
PROVISIONING_REQUIRED	A user or administrator has requested that the device be provisioned for use with Amazon WorkMail. Devices are also set to this status if the current policy for that device is modified in the Amazon WorkMail console.
PROVISIONING_SUCCEEDED	The device has been successfully provision ed. The device has enforced the given policy.
WIPE_REQUIRED	An administrator requested a wipe in the Amazon WorkMail console.
WIPE_SUCCEEDED	The device has been successfully wiped.

Managing mobile device access rules

Mobile device access rules for Amazon WorkMail allow administrators to control mailbox access for certain types of mobile devices. By default, each Amazon WorkMail organization uses a rule that grants mailbox access to any devices, regardless of type, model, operating system, or user agent. You can edit or replace that default rule with one of your own. You can also add, change, and delete rules.

> **Warning**
>
> If you delete all the mobile device access rules for an organization, Amazon WorkMail blocks all mobile device access.

You can create rules that allow or deny access based on the following device properties:

- **Device type**—"iPhone", "iPad", or "Android."
- **Device model**—"iPhone10C1", "iPad5C1", or "HTCOneX."
- **Device operating system**—"iOS 12.3.1 16F203", or "Android 8.1.0."

- **Device user agent**—"iOS/14.2 (18B92) exchangesyncd/1.0," or "Android-Mail/7.7.16.163886392.release."

To view device properties on the AWS Management Console, see Viewing mobile device details.

> **Note**
>
> Some devices and clients may not report properties for all fields. For information about working around those cases, see Dealing with empty fields

> **Important**
>
> Amazon WorkMail mobile device access rules only apply to devices that use the Microsoft Exchange ActiveSync protocol. Mobile clients that use a different protocol, such as IMAP, don't report the device properties listed here, so these rules won't apply.
> If you need to restrict access for devices that use other protocols, you can create access control rules. For more information about them, see Working with access control rules . As an example, you can restrict access to other protocols and webmail to just a range of corporate IP addresses, but allow Microsoft ActiveSync from elsewhere, and then use Mobile Device Access Rules to further limit the types and versions of allowed clients.

Topics

- How mobile device access rules work
- Using mobile device access rules

How mobile device access rules work

Mobile device access rules only apply to devices that use the Microsoft Exchange ActiveSync protocol. Each rule has a set of conditions that specify when the rule applies, plus an access effect of `ALLOW` or `DENY` for the device. A rule applies to an access request only if all of the conditions of the rule match properties of the user's mobile device. Rules with no conditions apply to all requests. Each condition uses a case-insensitive prefix match against the device's reported properties.

Amazon WorkMail evaluates rules as follows:

- If any DENY rule matches a device property, the policy blocks the device. DENY rules take precedence over ALLOW rules.
- If at least one ALLOW rule matches, and no DENY rule matches, the policy allows the device.
- If no rule applies, the device is blocked.

> **Important**
>
> Mobile devices report the properties that the rules use to operate. The devices report their properties during the Microsoft ActiveSync device provisioning process. Amazon WorkMail cannot independently verify that mobile clients report correct or up-to-date information.

Using mobile device access rules

You can use APIs or the AWS Command Line Interface (CLI) to create and manage mobile device access rules. For more information about the AWS CLI, see the AWS Command Line Interface User Guide.

> **Important**
>
> When you change an access rule for an Amazon WorkMail organization, the affected devices can take five minutes to follow the updated rule, and devices may show inconsistent behavior during that time. However, you immediately see correct behavior when you test rules. For more information, see Testing mobile device access rules.

Listing mobile device access rules

The following example shows how to list mobile device access rules.

```
aws workmail list-mobile-device-access-rules --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56
```

Creating mobile device access rules

The following example creates a rule that blocks all Android devices from accessing mailboxes.

```
aws workmail create-mobile-device-access-rule --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --name BlockAllAndroid --effect DENY --device-types
 "android"
```

The following example creates a rule that only allows a specific version of iOS. Be sure to remove the default `ALLOW-all` rule.

```
aws workmail create-mobile-device-access-rule --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --name AllowLatestiOS --effect ALLOW --device-
operating-systems "iOS 14.3"
```

Updating mobile device access rules

The following example updates a device rule by adding an identifier.

```
aws workmail update-mobile-device-access-rule --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --mobile-device-access-rule-id 1a2b3c4d --
name AllowLatestiOS --effect ALLOW --device-operating-systems "iOS 14.4"
```

Deleting a mobile device access rule

The following example deletes the mobile device access rule with the given identifier.

```
aws workmail delete-mobile-device-access-rule --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --mobile-device-access-rule-id 1a2b3c4d
```

Testing mobile device access rules

To test access rules, you can use the GetMobileDeviceAccessEffect API, or the get-mobile-device-access-effect command in the AWS CLI . For more information about the AWS CLI, see the AWS Command Line Interface User Guide.

When you test, you pass in the properties of a simulated mobile device, and the API or CLI returns the access effect—`ALLOW` or `DENY`—that a real mobile device with those properties would receive. For example, this command tests whether an iPhone running iOS 14.2, plus the default mail app, can access a mailbox.

```
aws workmail get-mobile-device-access-effect --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --device-type "iPhone" --device-model "iPhone10C1"
```

```
--device-operating-system "iOS 14.2.1 16F203" --device-user-agent "iOS/14.2 (18B92)
exchangesyncd/1.0"
```

Dealing with empty fields

Some mobile devices or clients may not report information for one or more fields, leaving the values empty. Rules can match against these devices by using the special value `$NONE` in a condition. For example, a rule with `DeviceTypes=["iphone", "ipad", "$NONE"]` will match devices that report a device type of `"iphone"` or `"ipad"`, or don't report a device type at all.

Negative conditions such as `NotDeviceTypes` or `NotDeviceUserAgents` won't match these empty values. For example, a rule with `NotDeviceTypes=["android"]` will match devices that report a device type other than `"android"`. However, the rule won't match devices that don't report a device type at all.

Managing mobile device access overrides

You use mobile device access overrides to override the results of mobile device access rules. The overrides apply to specific users and devices, and it reverses the default access rule. You can also use overrides to create one-off exceptions to access rules and allow or deny specific user and device pairs. In addition, you can use overrides with a `DefaultDenyAll` mobile device access rule. That defers access decisions to a third-party mobile device management (MDM) solution. For more information, see Managing overrides and Integrating with mobile device management solutions

Topics

- How mobile device access overrides work
- Managing overrides

How mobile device access overrides work

You create mobile device access overrides for a specific user and device pair. The override reverses the default access result when evaluating mobile device access rules for a given user and device. For example, if an access rule normally denies access, an access override allows that user and device to synchronize their email. Conversely, if an access rule normally allows access, you can create an override that prevents the user and device from synchronizing their mail. When you delete a mobile device access override, Amazon WorkMail again respects the result of the current mobile device access rules when deciding whether to grant access for that user and device.

> **Important**
>
> When you change a mobile device access override for an Amazon WorkMail organization, the affected devices can take five minutes to follow the updated override.

Managing overrides

Mobile device access overrides can be created, updated, or deleted using the API or AWS Command Line Interface. For more information about the AWS CLI, see the AWS Command Line Interface User Guide.

To find device ID, use the AWS Management Console. For more information, see Viewing mobile device details.

Listing mobile device access overrides

This example shows how to list all mobile device access overrides for a specified Amazon WorkMail organization.

```
aws workmail list-mobile-device-access-overrides --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56
```

Creating and updating mobile device access overrides

This will create a mobile device access override to deny access to the specified Amazon WorkMail organization, user, and device ID.

```
aws workmail put-mobile-device-access-override --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --user-id user1@domain.com --device-
id 6APMEKPHCP2ND42VIJ4BR8ECD0 --effect DENY
```

An existing mobile device access override can be modified to have a different effect. This will update the previously created mobile device access override to allow access instead of denying.

```
aws workmail put-mobile-device-access-override --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --user-id user1@domain.com --device-
id 6APMEKPHCP2ND42VIJ4BR8ECD0 --effect ALLOW
```

Deleting mobile device access overrides

This will delete the mobile device access override for the specified Amazon WorkMail organization, user, and device ID.

```
aws workmail delete-mobile-device-access-override --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --user-id user1@domain.com --device-
id 6APMEKPHCP2ND42VIJ4BR8ECD0
```

Integrating with mobile device management solutions

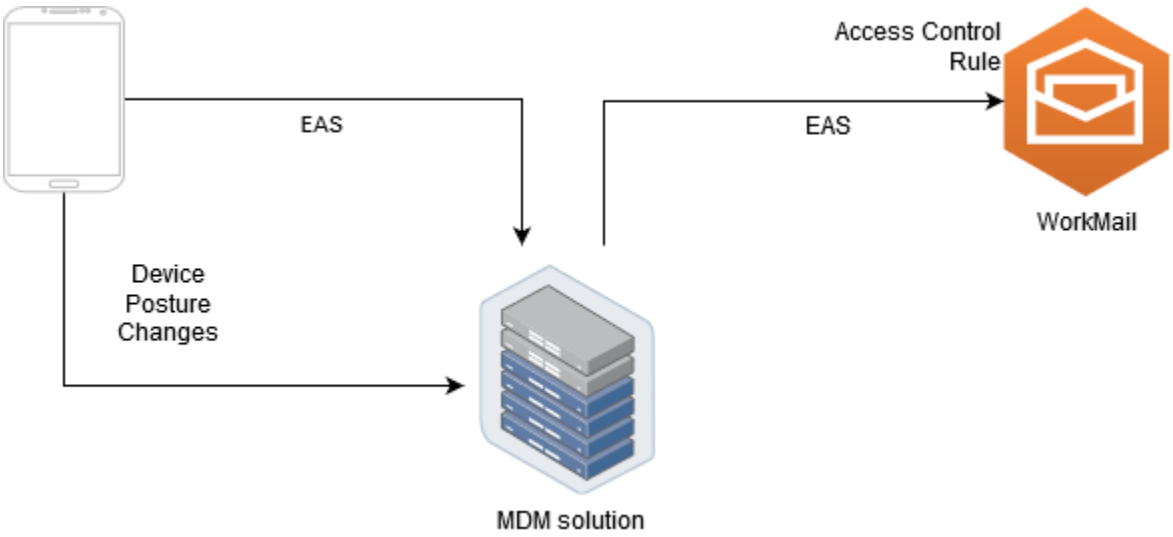
Amazon WorkMail supports some basic mobile device management capabilities through mobile device policies and mobile device access rules. However, those features can only interact with mobile devices through the Microsoft Exchange ActiveSync (EAS) protocol, so they have limited ability to introspect and enforce device security posture. Administrators who need greater control over device security and compliance can use a third-party mobile device management (MDM) solution.

Mobile device management solutions overview

You can configure your MDM solution in two modes, *proxy* or *direct*. Consult your MDM documentation to see which modes your solution supports.

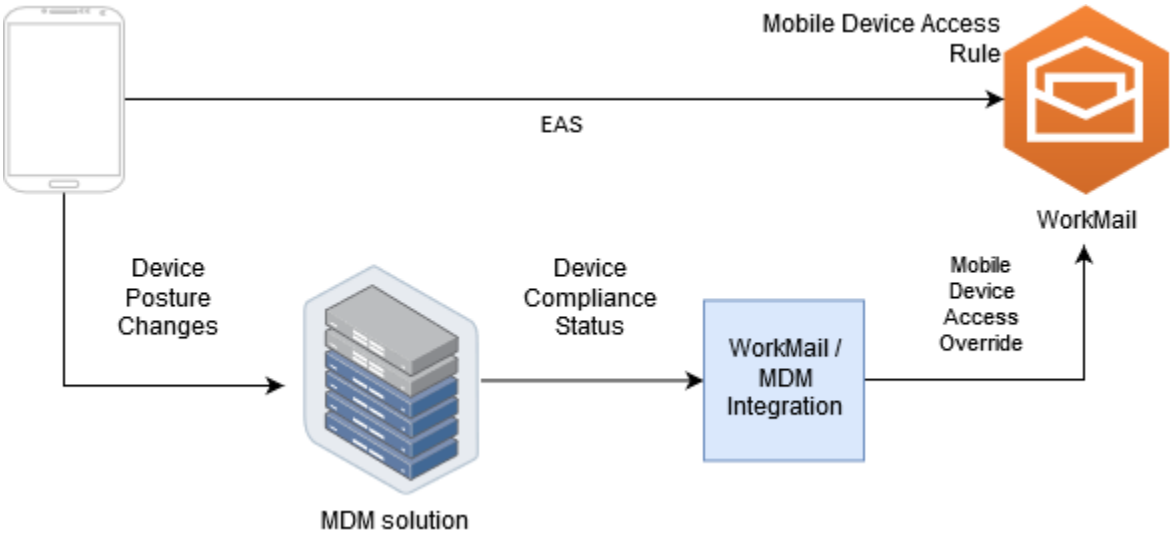
In proxy mode, mobile devices use the Exchange Active Sync (EAS) protocol via your MDM solution to access Amazon WorkMail. The MDM solution uses device posture to allow or deny access to Amazon WorkMail data. On the Amazon WorkMail side, use an Access Control Rule that allows EAS access only from the MDM solution's IP address or addresses. For more information, refer to Working with access control rules.

The following image shows a typical proxy mode configuration.



In direct mode, mobile devices use EAS to access Amazon WorkMail directly. Your MDM solution receives device posture changes and continually assesses whether each device meets those requirements. When the MDM solution detects a posture changes, such as a device going out of compliance, it can take several actions and typically emits notifications or events. An Amazon WorkMail administrator can set up a system to listen to these compliance status events and automatically create mobile device access overrides that allow or deny access to devices when they go in or out of compliance with the MDM device requirements.

The following image shows a typical direct mode configuration.



Configuring a WorkMail organization to integrate with a third-party MDM solution in direct mode

To integrate with a third-party mobile device management (MDM) solution in direct mode, you must meet these requirements:

- Create access control rules that restrict access to user devices to only the ActiveSync protocol.
- Create a default "deny-to-all" mobile device access rule to ensure that all unknown or unmanaged mobile devices are denied by default.
- Adopt a mobile device management solution that emits custom notifications or events when a device changes security posture, meaning it goes in or out of compliance.
- Create a custom software component to listen to those notifications and call the Amazon WorkMail SDK to create mobile device access overrides.

These components ensure that all user devices meet their MDM compliance requirements before being allowed to access their Amazon WorkMail mailboxes.

Use access control rules to restrict mobile device access to ActiveSync

You must ensure that all devices use only the ActiveSync protocol, and you can use access control rules to do so. For example, you can grant access to other mail protocols only from an internal corporate IP address range, and then allow only ActiveSync when accessing eamil from outside the corporate firewall. You must do this because only ActiveSync allows you to identify devices using a device ID. You can't use protocols such as the Internet Message Access Protocol (IMAP) or Exchange Web Services. For more information, see Working with access control rules.

Create a default 'deny to all' access rule

To defer all mobile device access decisions to the third-party mobile device management solution, create an access rule that automatically denies all devices unless overridden on a per-user or per-device basis. For more information, refer to Managing mobile device access rules.

This example shows a 'deny to all' rule.

```
aws workmail create-mobile-device-access-rule --organization-id
 m-a123b4c5de678fg9h0ij1k2lm234no56 --name DefaultDenyAll --effect DENY
```

React to device posture changes and create mobile device access overrides

You must configure your MDM solution to send notifications for device posture changes. These notifications must be consumed by a component that can use the Amazon WorkMail SDK to create or update mobile device access overrides. By default, Amazon WorkMail denies access to unmanaged or newly provisioned devices because of the default “deny to all” mobile device access rule shown earlier in this topic. When the MDM solution determines that the device meets all requirements and emits a notification indicating that the device is compliant, this component can react to this notification by creating a mobile device access override with an effect of `ALLOW` for the specified user and device. If the device later goes out of compliance, the mobile device management solution emits another notification, and the access override can be deleted or modified to deny access for that device. For more information, see Managing mobile device access overrides.

For an example of Amazon WorkMail integrated with MDM, see this AWS sample application.

Working with mailbox permissions

You can use mailbox permissions in Amazon WorkMail to grant users and groups the right to work in other users' mailboxes. Mailbox permissions apply to an entire mailbox. They enable multiple users to access the same mailbox without sharing that mailbox's credentials. Users with mailbox permissions can read and modify mailbox data and send email from the shared mailbox.

> **Note**
>
> Users with permissions to a mailbox belonging to a user hidden from the global address list can still access the hidden user's mailbox.

The following list shows the permissions that you can grant:

- **Full Access** – Enables full read and write access to the mailbox, including permissions to modify folder-level permissions.

  > **Note**
  >
  > This options is only available for users. Groups can't be granted full access rights.

- **Send On Behalf** – Enables a user or group to send email on behalf of another user. The mailbox owner appears in the **From:** header, and the sender appears in the **Sender:** header.
- **Send As** – Enables a user or group to send email as the mailbox owner, without showing the actual sender of the message. The mailbox owner appears in both the **From:** and **Sender:** headers.
- **None** – Prevents a user or group from sending emails.

> **Note**
>
> Granting mailbox permissions to a group extends those permissions to all the members of that group, including members of nested groups.

When you grant mailbox permissions, the Amazon WorkMail AutoDiscover service automatically updates access to those mailboxes for the users or groups you added.

For the Microsoft Outlook client in Windows, users with full access permissions can automatically access the shared mailboxes. Allow up to 60 minutes for the changes to propagate, and then restart Microsoft Outlook.

For the Amazon WorkMail web application and in other email clients, users with full access permissions can manually open the shared mailboxes. Opened mailboxes stay open, even between sessions, unless the user closes them.

Topics

- About mailbox and folder permissions
- Managing mailbox permissions for users
- Managing mailbox permissions for groups

About mailbox and folder permissions

Mailbox permissions apply to all folders within a mailbox. These permissions can only be enabled by the AWS account holder or an IAM user authorized to call the Amazon WorkMail management API. To set and change permissions for mailboxes, or for groups as a whole, use the AWS Management Console or the Amazon WorkMail API. You can manage up to 100 mailbox and group permissions from the console. To manage permissions for more users and groups, use the Amazon WorkMail API.

Folder permissions apply only to a single folder. End users can set folder permissions by using an email client, or by using the Amazon WorkMail web application. For more information about using the Amazon WorkMail web application to share folders, see Sharing folders and folder permissions in the *Amazon WorkMail User Guide*.

Managing mailbox permissions for users

You can use the Amazon WorkMail console to manage mailbox permissions for users, as well as groups. The following sections explain how to manage permissions for users. For information about managing permissions for groups, refer to Managing mailbox permissions for groups.

Topics

- Adding permissions
- Editing mailbox permissions for users

Adding permissions

When you add permissions, you grant one user the right to perform one or more tasks in another user's mailbox. For example, say that Employee A needs to send messages on behalf of his supervisor, Employee B. To grant that permission, you go to Employee B's mailbox settings and grant Employee A permission to do the requested task.

To add mailbox permissions

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization for which you want to manage permissions.
3. In the navigation pane, choose **Users**, and then select the name of the user for whom you want to manage permissions.
4. Choose the **Permissions** tab, and then choose **Add permissions**.

   The **Add permissions** dialog box appears.
5. Open the **Add new permissions** list and select the user or group that needs access to the mailbox.
6. Under **Mailbox permissions** and **Send permissions**, choose the desired options.
7. Choose **Add**.

New permissions can take up to five minutes to propagate to users.

Editing mailbox permissions for users

When you edit mailbox permissions for a user, you change the access that others have to that user's mailbox. Editing mailbox permissions doesn't change access for the mailbox's original user.

To edit mailbox permissions

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

2. In the navigation pane, choose **Organizations**, and then choose the name of the organization for which you want to manage permissions.
3. In the navigation pane, choose **Users**, and then select the name of the user whose permissions you want to edit.
4. Choose the **Permissions** tab.

   A list of the users and groups that have access to the mailbox appears.
5. Select the radio button next to the user or group that you want to change, then do any of the following:

   To remove a user's permissions

   1. Choose **Remove**.

      The **Remove permissions** dialog box appears.
   2. In the **Remove permissions** dialog box, choose **Remove**.

   To edit a user's permissions

   1. Choose **Edit**.

      The **Edit permissions** dialog box appears.
   2. Set the permissions as needed, and then choose **Save**.

   To grant another user permissions to the mailbox

   1. Choose **Add permissions**.

      The **Add permissions** dialog box appears.
   2. Open the **Add new permissions** list and select the user that you want to add.
   3. Set the permissions as needed, and then choose **Add**.

Changes to permissions can take up to five minutes to propagate to users.

Managing mailbox permissions for groups

You can add or remove group permissions for Amazon WorkMail.

> **Note**
>
> You can't apply **Full Access** permissions to a group, because groups don't have a mailbox to access.

To manage group permissions

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of the organization for which you want to manage permissions.
3. In the navigation pane, choose **Groups**, and then select the name of the group for which you want to set permissions.
4. Choose the **Permissions** tab, then choose **Add permissions**.

   The **Add permissions** dialog box appears.
5. Open the **Add new permissions** list and select the user or group to grant permissions for the mailbox.
6. Under **Mailbox permissions** and **Send permissions**, choose the desired options.
7. Choose **Add**.

Changes to permissions can take up to five minutes to propagate to users.

Programmatic access to mailboxes

To programmatically access Amazon WorkMail mailboxes, use the Exchange Web Services (EWS) protocol. With EWS, you can access all item types in a mailbox. Here are some EWS libraries that you can use with Amazon WorkMail:

- **Java** – EWS Java API
- **.Net** – EWS Managed API
- **Python** – Exchangelib

Amazon WorkMail also supports IMAP and SMTP protocols, which you can use to send and receive emails. You can see the URLs supported for Amazon WorkMail protocols under Amazon WorkMail endpoints and quotas.

When using the EWS protocol, Amazon WorkMail supports the following authentication methods:

- **Basic Authentication** – With basic authentication, you enter an email address and password.
- **Impersonation roles** – With impersonation roles, you access users' mailboxes without entering the user's credentials.

Topics

- Managing impersonation roles
- Using impersonation roles

Managing impersonation roles

With impersonation roles, administrators configure programmatic access to user's mailboxes without entering the user's credentials. Services and tools can assume an impersonation role to perform actions in user's mailboxes. Impersonation is only supported with the EWS protocol.

Impersonation roles overview

To allow impersonation, administrators must create an impersonation role with the following properties:

- **Role type** – Choose either **Full access** or **Read only**. The role type limits the kind of operations a role can perform.
- **Rules** – A list of rules that define which users the impersonation role can impersonate.

Amazon WorkMail evaluates the rules on the following conditions:

- If any **DENY** rule matches, the policy denies impersonation. **DENY** rules take precedence over any **ALLOW** rules.
- If at least one **ALLOW** rule matches, and no **DENY** rule matches, the policy allows impersonation.
- If no rule applies, impersonation is denied.

> **Note**
>
> To allow impersonation for all users in an Amazon WorkMail organization, create a rule with the **ALLOW** effect and with no conditions.

> **Warning**
>
> You must create rules to allow an impersonation role to impersonate a user. If you do not specify rules, an impersonation role can't assume a user's access rights.

After the impersonation role is created, you can use it to get access to users' mailboxes. For more information, see Using impersonation roles.

Security considerations

The use of impersonation roles creates the potential for security issues within your Amazon WorkMail organization and AWS account. Here are some of the potential issues to consider when you create an impersonation role:

- **Transitive permissions** – If user A has access to user B's mailbox, and an impersonation role is allowed to impersonate user A, then this impersonation role can impersonate user A's access permissions and access user’s B mailbox.
- **Access control** – You can use access control rules to limit impersonation role access. For more information, see Working with access control rules.

- **IAM policy** – You can assign an `AssumeImpersonationRole` action to a particular Amazon WorkMail organization and impersonation role by using the `workmail:ImpersonationRoleId` condition. To see an IAM policy example, see How Amazon WorkMail works with IAM.

Creating impersonation roles

You can create impersonation roles from the Amazon WorkMail console.

To create an impersonation role

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization.
3. Choose **Impersonation roles**, and then choose **Create role**.
4. The **Create impersonation role** dialog box appears. Under **Role**, enter the following information:

   - **Name** – Enter a unique name for the impersonation role.
   - (Optional) **Description** – Enter a description for the impersonation role.
   - **Role type** – Choose **Read only** or **Full access**.
5. Under **Rules**, choose **Add rule**.
6. The **Add rule** dialog box appears. Enter the following information:

   - **Name** – Enter a unique name for the rule.
   - (Optional) **Description** – Enter a description for the rule.
   - Under **Effect**, choose **Allow** or **Deny**. This allows or denies access based on the conditions you select in the following step.
   - (Optional) Under **This rule:**, choose **Matches requests that impersonate the selected users** to include specific users. Choose **Matches requests that impersonate users other than the selected users** to add users other than the selected users.
7. Choose **Add rule**.

> **Note**
>
> Rules are only saved when you save the corresponding role.

8. Choose **Create role**.

Editing impersonation roles

You can edit impersonation roles from the Amazon WorkMail console.

To edit an impersonation role

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization.
3. Choose **Impersonation roles**.
4. Select the impersonation role name you want to edit, then choose **Edit**.
5. The **Edit impersonation role** dialog box appears. Under **Role**, enter the following information:

   - **Name** – Enter a unique name for the impersonation role.
   - (Optional) **Description** – Enter a description for the impersonation role.
   - **Role type** – To give the impersonation role read only access to a user's mailbox, choose **Read only**. To give the impersonation role rights to read and modify items in a user's mailbox, choose **Full access**.
6. Under **Rules**, select the rule that you want to edit and choose **Edit**.
7. The **Edit rule** dialog box appears. Enter the following information:

   - **Name** – Edit the name of the rule.
   - (Optional) **Description** – Update or enter a description for the rule.
   - Under **Effect**, choose **Allow** to allow access when the conditions set in the rules are met. To deny access, choose **Deny**.

- (Optional) Under **This rule:**, choose **Matches requests that impersonate the selected users** to include specific users. Choose **Matches requests that impersonate users other than the selected users** to add users other than the selected users.

8. Choose **Save**.
9. Choose **Save changes**.

> **Important**
>
> When you change an impersonation rule, the affected mailboxes can take up to five minutes to update. During the rule update process, you may observe inconsistent behavior in your mailbox. However, if you test a role, Amazon WorkMail responds as expected based on the updated rule. For more information, see Testing impersonation roles.

Testing impersonation roles

You can test an impersonation role from the Amazon WorkMail console.

To test an impersonation role

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization.
3. Choose **Impersonation roles**.
4. Select the impersonation role that you want to test.
5. Choose **Test role**.
6. The **Test impersonation role** dialog box appears. Under **Target user**, select the user for which you want to test the impersonation access.
7. Choose **Test**.

Deleting impersonation roles

You can delete an impersonation role from the Amazon WorkMail console.

To delete an impersonation role

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, and then choose the name of the organization.
3. Choose **Impersonation roles**.
4. Select the impersonation role name you want to delete.
5. Choose **Delete**.
6. The **Delete role** dialog box appears. To confirm deletion, enter the role's name into the dialog box and choose **Delete**.

Using impersonation roles

To access mailbox data, use the Amazon WorkMail API action `AssumeImpersonationRole`. For more details on Amazon WorkMail APIs, see API Reference.

`AssumeImpersonationRole` returns a `Token`. This `Token` must be passed within 15 minutes to the EWS protocol through the HTTP header `Authorization`.

The following examples demonstrate how to use impersonation roles with the EWS protocol. The constants used in the examples specify the following details unique to your organization and account:

- *`WORKMAIL_ORGANIZATION_ID`* – Amazon WorkMail organization ID
- *`IMPERSONATION_ROLE_ID`* – Impersonation role ID
- *`WORKMAIL_EWS_URL`* – EWS endpoint available at Amazon WorkMail endpoints and quotas
- *`EMAIL_ADDRESS`* – Email address of the user mailbox

Example Java – EWS Java API

```
import software.amazon.awssdk.services.workmail.WorkMailClient;
import software.amazon.awssdk.services.workmail.model.AssumeImpersonationRoleRequest;
import software.amazon.awssdk.services.workmail.model.AssumeImpersonationRoleResponse;

import microsoft.exchange.webservices.data.core.ExchangeService;
import microsoft.exchange.webservices.data.core.enumeration.misc.ExchangeVersion;
import microsoft.exchange.webservices.data.misc.ImpersonatedUserId;
import microsoft.exchange.webservices.data.core.enumeration.misc.ConnectingIdType;

// ...

AssumeImpersonationRoleResponse response = workMailClient.assumeImpersonationRole(
    AssumeImpersonationRoleRequest.builder()
        .organizationId(WORKMAIL_ORGANIZATION_ID)
        .impersonationRoleId(IMPERSONATION_ROLE_ID)
        .build());

ExchangeService exchangeService = new
 ExchangeService(ExchangeVersion.Exchange2010_SP2);
exchangeService.setUrl(URI.create(WORKMAIL_EWS_URL));
exchangeService.getHttpHeaders().put("Authorization", "Bearer " + response.token());
exchangeService.setImpersonatedUserId(new
 ImpersonatedUserId(ConnectingIdType.SmtpAddress, EMAIL_ADDRESS));
```

Example .Net – EWS Managed API

```
using Amazon.WorkMail;
using Amazon.WorkMail.Model;

using Microsoft.Exchange.WebServices.Data;

// ...

AssumeImpersonationRoleRequest request = new AssumeImpersonationRoleRequest();
request.OrganizationId = WORKMAIL_ORGANIZATION_ID;
request.ImpersonationRoleId = IMPERSONATION_ROLE_ID;
AssumeImpersonationRoleResponse response =
 workMailClient.AssumeImpersonationRole(request);

ExchangeService service = new ExchangeService(ExchangeVersion.Exchange2010_SP2);
service.Url = new Uri(WORKMAIL_EWS_URL);
```

```
service.HttpHeaders.Add("Authorization", "Bearer " + response.Token);
service.ImpersonatedUserId = new
 ImpersonatedUserId(ConnectingIdType.SmtpAddress, EMAIL_ADDRESS);
```

Example Python – Exchangelib

```
import boto3

from requests.auth import AuthBase
from exchangelib.transport import AUTH_TYPE_MAP
from exchangelib import Configuration, Account, Version, IMPERSONATION
from exchangelib.version import EXCHANGE_2010_SP2


work_mail_client = boto3.client("workmail")

class ImpersonationRoleAuth(AuthBase):
    def __init__(self):
         self.token = work_mail_client.assume_impersonation_role(
             OrganizationId=WORKMAIL_ORGANIZATION_ID,
             ImpersonationRoleId=IMPERSONATION_ROLE_ID
        )["Token"]

     def __call__(self, r):
         r.headers["Authorization"] = "Bearer " + self.token
         return r

AUTH_TYPE_MAP["ImpersonationRoleAuth"] = ImpersonationRoleAuth

ews_config = Configuration(
     service_endpoint=WORKMAIL_EWS_URL,
     version=Version(build=EXCHANGE_2010_SP2),
     auth_type="ImpersonationRoleAuth"
)
ews_account = Account(
     config=ews_config,
     primary_smtp_address=EMAIL_ADDRESS,
     access_type=IMPERSONATION
)
```

Exporting mailbox content

Use the StartMailboxExportJob API action in the *Amazon WorkMail API Reference* to export Amazon WorkMail mailbox content to an Amazon Simple Storage Service (Amazon S3) bucket. This action exports all email messages and calendar items from the specified mailbox to a `.zip` file in the Amazon S3 bucket, in MIME format. Other items, such as contacts and tasks, are not exported.

The time it takes for the mailbox export job to finish is dependent on the size and number of items in the mailbox. Because the mailbox export job takes place over a period of time, it does not represent a snapshot of the mailbox content at a single point in time. To see the status of an export job, use the DescribeMailboxExportJob or ListMailboxExportJobs API actions in the *Amazon WorkMail API Reference*.

When a mailbox export job is completed, the `.zip` file in the Amazon S3 bucket is encrypted using the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK) that you provide. Because AWS KMS encryption is integrated with Amazon S3, the decrypted data is visible to the user who downloads it, as long as the user has access to the AWS KMS CMK.

Prerequisites

The following are prerequisites for exporting mailbox content:

- The ability to program.
- An Amazon WorkMail administrator account.
- An Amazon S3 bucket that does not allow public access. For more information, see Using Amazon S3 block public access in the *Amazon Simple Storage Service User Guide* and the *Amazon Simple Storage Service User Guide*.
- A symmetric AWS KMS CMK. For more information, see Getting started in the *AWS Key Management Service Developer Guide*.
- An AWS Identity and Access Management (IAM) role with a policy that grants permission to write to the Amazon S3 bucket and encrypt the sent files with the AWS KMS CMK. For more information, see How Amazon WorkMail works with IAM.

IAM policy examples and role creation

The following example shows an IAM policy that grants permission to write to the Amazon S3 bucket and encrypt the sent files with the AWS KMS CMK. To use this example policy in the following Example: Exporting mailbox content procedure, save the policy as a JSON file with file name `mailbox-export-policy.json`.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:GetBucketPolicyStatus"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": [
                "arn:aws:kms:us-east-1:111122223333:key/KEY-ID"
            ],
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "s3.us-east-1.amazonaws.com"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::amzn-s3-
demo-bucket/S3-PREFIX*"
                }
            }
```

```
        }
    ]
}
```

The following example shows an IAM trust policy that is attached to the IAM role you create. To use this example policy in the following Example: Exporting mailbox content procedure, save the policy as a JSON file with file name `mailbox-export-trust-policy.json`.

You don't have to use the `aws:SourceArn` and `aws:SourceAccount` conditions at the same time. For example, you can remove `aws:SourceArn` from the policy if you need to use the same role to export messages from different Amazon WorkMail organizations under the same AWS account. For more information about condition keys, refer to the AWS global condition context keys in the *AWS Identity and Access Management user guide*.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "export.workmail.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:workmail:us-
east-1:111122223333:organization/m-a123b4c5de678fg9h0ij1k2lm234no56"
        }
      }
    }
  ]
}
```

You can use the AWS CLI to create the IAM role in your account by running the following commands.

```
aws iam create-role --role-name WorkmailMailboxExportRole --assume-role-policy-
document file://mailbox-export-trust-policy.json --region us-east-1
```

```
aws iam put-role-policy --role-name WorkmailMailboxExportRole --policy-
name MailboxExport --policy-document file://mailbox-export-policy.json
```

For more information about the AWS CLI, see the AWS Command Line Interface User Guide.

Example: Exporting mailbox content

After you create the IAM role and policies in the preceding section, complete the following steps to export your mailbox content. You must have your Amazon WorkMail organization ID and user ID (entity ID), which you can access in the Amazon WorkMail console or by using the Amazon WorkMail API.

Example: To export mailbox content

1. Use the AWS CLI to start the mailbox export job.

   ```
   aws workmail start-mailbox-export-job --organization-id m-
   a123b4c5de678fg9h0ij1k2lm234no56 --entity-
   id S-1-1-11-1111111111-2222222222-3333333333-3333 --kms-key-
   arn arn:aws:kms:us-east-1:111122223333:key/KEY-ID --role-arn
    arn:aws:iam::111122223333:role/WorkmailMailboxExportRole --s3-bucket-name amzn-s3-
   demo-bucket --s3-prefix S3-PREFIX
   ```

2. Use the AWS CLI to monitor the state of the mailbox export jobs for your Amazon WorkMail organization.

   ```
   aws workmail list-mailbox-export-jobs --organization-id m-
   a123b4c5de678fg9h0ij1k2lm234no56
   ```

   Alternatively, use the job ID generated by the **`start-mailbox-export-job`** command to monitor the state of that mailbox export job only.

```
aws workmail describe-mailbox-export-job --organization-id m-
a123b4c5de678fg9h0ij1k2lm234no56 --job-id JOB-ID
```

When the mailbox export job state is **COMPLETED**, the exported mailbox items are available in a `.zip` file in the specified Amazon S3 bucket.

The following is an example of the output log from the exported mailbox:

```
{
  "totalNonExportableItems" : "13",
  "totalMessages" : "76",
  "sha384Hash" : "4de93a***96a1dd",
  "totalBytes" : "161892",
  "totalFolders" : "15",
  "startTime" : "168***380",
  "endTime" : "168***384"
}
```

Note

totalNonExportableItems are unsupported items like notes and contacts.

Considerations

The following considerations apply when exporting mailbox jobs for Amazon WorkMail:

- You can run up to 10 concurrent mailbox export jobs for a given Amazon WorkMail organization.
- You can run a mailbox export job for a given mailbox as often as once every 24 hours.
- The following resources must all be in the same AWS Region:
  - Amazon WorkMail organization
  - AWS KMS CMK
  - Amazon S3 bucket

Troubleshooting

The topics in this section explain how to troubleshoot issues in Amazon WorkMail.

Topics

- Viewing email headers
- Mail routing

Viewing email headers

The information in email headers can help you troubleshoot common user email issues. Amazon WorkMail allows you to view the header information for any message.

To view email headers in Amazon WorkMail

1. In the Amazon WorkMail web application, double-click on the email message to open.
2. Choose **Message options** (the gear and envelope icon) located in the upper-right corner of the message, next to the **Sent on** date.

The email headers appear under **Internet Headers**.

Mail routing

If a user stops receiving emails, your Amazon WorkMail organization may be experiencing a mail routing issue. The steps in this section explain common ways to resolve delivery and routing issues.

Inbound mail issues:

- Check the MX record for the domain associated with your Amazon WorkMail organization. `WorkMail` should be the only entry and should have the lowest priority. Multiple MX records can lead to the wrong service receiving messages. For more information about MX records, see Verifying domains.
- Check the Domain-based Message Authentication, Reporting, and Conformance (DMARC) settings for your organization in the Amazon WorkMail console. DMARC records are used to protect against common attacks, such as spoofing or phishing, that can compromise a user's

account credentials. For more information about DMARC, see Enforcing DMARC policies on incoming email.

- Check the Amazon Simple Email Service inbound rule. If the rule contains actions other than Amazon WorkMail, those actions can fail and cause Amazon WorkMail to stop receiving mail. For more information about Amazon SES rules, see Integrate with Amazon WorkMail action in the *Amazon Simple Email Service Developer Guide*.
- Enable message tracking in Amazon WorkMail, and then check the logs for delivery problems. For more information about message tracking, see Enabling email event logging.

Outbound mail issues

- Ensure your SPF record includes Amazon SES. Check the domains page in the Amazon WorkMail console to verify. For more information about SPF, see Authenticating email with SPF.
- Ensure Amazon WorkMail has permissions to use the domain. If not, add the domain again. Adding a domain in this guide provides the how-to steps.

Using email journaling with Amazon WorkMail

You can set up journaling to record your email communication, using integrated third-party archiving and eDiscovery tools. This ensures that email storage compliance regulations for privacy protection, data storage, and information protection are met.

Using journaling

Amazon WorkMail journals all email messages that are sent to any user in the specified organization, as well as all email messages sent by users in that organization. A copy of all email messages is sent to an address specified by the system administrator, in a format called `journal record`. This format is compatible with Microsoft email programs. There is no additional charge for email journaling.

Two email addresses are used for email journaling—a journaling email address and a report email address. The journaling email address is the address of a dedicated mailbox or third-party device that is integrated with your account, where journal reports are sent. The report email address is the address of your system administrator, where notifications of failed journal reports are sent.

All journal records are sent from an email address that is automatically added to your domain and looks like the following.

```
amazonjournaling@yourorganization.awsapps.com
```

There is no mailbox associated with this address, and you won't be able to create one using this name or address.

> **Note**
>
> don't delete the following domain record from the Amazon Simple Email Service (Amazon SES) console, or email journaling stops functioning.
>
> ```
> yourorganization.awsapps.com
> ```

Every incoming or outgoing email message generates one journal record, regardless of the number of recipients or user groups. Email that fails to generate a journal record generates an error notification, which is sent to the report email address.

To enable email journaling

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

   If necessary, change the AWS Region. In the bar at the top of the console window, open the **Select a Region** list and choose a Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
2. In the navigation pane, choose **Organizations**, then choose the name of your organization.
3. In the navigation pane, **Organization settings**, choose the **Journaling** tab, and then choose **Edit**.
4. Move the **Journaling status** slider to the on position.
5. in the **Journaling email address** box, enter the email address given by your email journaling provider.

   > **Note**
   >
   > We recommend using a dedicated journaling provider.
6. In the **Report email address**, enter the email administrator's address.
7. Choose **Save**. The changes apply immediately.

Document history

The following table describes important changes in each release of the *Amazon WorkMail Administrator Guide*. For notification about updates to this documentation, you can subscribe to an RSS feed.

Change	Description	Date
Audit logging support	The audit logs can be used to monitor user's access to mailboxes, audit for suspiciou s activity, and debug access control and availability provider configurations. For more information, see Enabling audit logging and Logging and monitoring in Amazon WorkMail in the *Amazon WorkMail Administr ator Guide*.	March 20, 2024
Transport Layer Security (TLS) support	Amazon WorkMail discontin ued support for Transport Layer Security (TLS) 1.0 and 1.1. If you are using TLS 1.0 or 1.1, you must upgrade the TLS version to 1.2.	November 2, 2023
Remote users	Remote users are Amazon WorkMail users hosted outside the Amazon WorkMail organization or hosted on a different email domain. For more information, see Users in the *Amazon WorkMail Administrator Guide*.	September 18, 2023

Programmatic access to mailboxes	Amazon WorkMail now offers Impersonation Roles to grant programmatic access to mailboxes. For more information, see Programmatic access to mailboxes in the *Amazon WorkMail Administrator Guide*.	October 4, 2022
Configure Custom Availability Providers on Amazon WorkMail	Amazon WorkMail supports the use of Custom Availability Providers (CAPs). For more information, see Configuring a Custom Availability Provider in the *Amazon WorkMail Administrator Guide*.	June 30, 2022
Console changes for creating an organization	The Amazon WorkMail console experience for creating an organization is updated. For more information, see Creating an organization in the *Amazon WorkMail Administrator Guide*.	October 23, 2020
Exporting mailbox content	Use the `StartMailboxExportJob` API action to export Amazon WorkMail mailbox content to an Amazon Simple Storage Service (Amazon S3) bucket. For more information, see Exporting mailbox content in the *Amazon WorkMail Administrator Guide*.	September 22, 2020

Mailbox retention policies	Set mailbox retention policies for your Amazon WorkMail organization that automatically delete email messages after a time period that you choose. For more information, see Setting mailbox retention policies in the *Amazon WorkMail Administrator Guide*.	May 28, 2020
Synchronous and asynchronous **Run Lambda** actions	Choose synchronous or asynchronous configurations for **Run Lambda** actions in Amazon WorkMail email flow rules. For more information, see Configuring AWS Lambda for Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	May 11, 2020
Working with access control rules	Access control rules allow Amazon WorkMail administrators to control how their organization's mailboxes are accessed. For more information, see Working with access control rules in the *Amazon WorkMail Administrator Guide*.	February 12, 2020

Tagging an organization	Tag an Amazon WorkMail organization to different iate between organizations in the AWS Billing and Cost Management console, or to control access to organization resources. For more informati on, see Tagging an organizat ion in the *Amazon WorkMail Administrator Guide*.	January 23, 2020
Enforce DMARC policies on incoming email	For more information, see Enforcing DMARC policies on incoming email in the *Amazon WorkMail Administrator Guide*.	October 17, 2019
Retrieving message content with Lambda	Use the Amazon WorkMail Message Flow API with AWS Lambda to retrieve message content. For more informati on, see Retrieving message content with Lambda in the *Amazon WorkMail Administr ator Guide*.	September 12, 2019
Logging Amazon WorkMail email events	Enable email event logging in the Amazon WorkMail console to track email messages for your organization. For more information, see Tracking messages in the *Amazon WorkMail Administrator Guide*.	May 13, 2019

Route 53 DNS record insertion	When setting up a domain that is managed in a Route 53 public hosted zone, Amazon WorkMail automatically inserts the DNS records for you. For more information, see Adding a domain in the *Amazon WorkMail Administrator Guide*.	February 13, 2019
Configuring Lambda for inbound email rule actions	Amazon WorkMail supports configuring Lambda functions to use with inbound email flow rules. For more information, see Managing email flows in the *Amazon WorkMail Administrator Guide*.	January 24, 2019
Configuring Lambda for Amazon WorkMail	Amazon WorkMail supports configuring Lambda functions to use with outbound email flow rules. For more information, see Configuring Lambda for Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	November 19, 2018
SMTP routing	Amazon WorkMail supports configuring SMTP gateways to use with outbound email flow rules. For more information, see Configuring SMTP gateways in the *Amazon WorkMail Administrator Guide*.	November 1, 2018

Debugging tools for custom domains	Amazon WorkMail has added debugging tools for custom domains. For more information, see Adding a domain in the *Amazon WorkMail Administrator Guide*.	October 15, 2018
Support for Outlook 2019	Amazon WorkMail supports Outlook 2019 for Windows and macOS. For more information, see Amazon WorkMail system requirements in the *Amazon WorkMail Administrator Guide*.	October 1, 2018
Various updates	Various updates to topic layout and organization.	July 12, 2018
Mailbox permissions	You can use mailbox permissions in Amazon WorkMail to grant users or groups the right to work in other users' mailboxes. For more information, see Working with mailbox permissions in the *Amazon WorkMail Administrator Guide*.	April 9, 2018
Support for AWS CloudTrail	Amazon WorkMail is integrated with AWS CloudTrail. For more information, see Logging Amazon WorkMail API calls with AWS CloudTrail in the *Amazon WorkMail Administrator Guide*.	December 12, 2017

Support for email flows	You can set up email flow rules for handling incoming email based on a sender's email address or domain. For more information, see Managing email flows in the *Amazon WorkMail Administrator Guide*.	July 5, 2017
Updates to Quick Setup	Quick Setup now creates an Amazon WorkMail directory for you. For more information, see Set up Amazon WorkMail with Quick Setup in the *Amazon WorkMail Administrator Guide*.	May 10, 2017
Support for a wider range of email clients	You can now use Amazon WorkMail with Microsoft Outlook 2016 for Mac and IMAP email clients. For more information, see System requirements for Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	January 9, 2017
Support for SMTP journaling	You can set up journaling to record your email communication. For more information, see Using email journaling with Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	November 25, 2016

Support for email redirection to external email addresses	You can set up email redirecti on rules by updating the Amazon SES identity policy for your domain. For more information, see Edit domain identity policies in the *Amazon WorkMail Administr ator Guide*.	October 26, 2016
Support for interoperability	You can enable interoper ability between Amazon WorkMail and Microsoft Exchange. For more informati on, see Interoperability between Amazon WorkMail and Microsoft Exchange in the *Amazon WorkMail Administr ator Guide*.	October 25, 2016
General availability	The general availability release of Amazon WorkMail.	January 4, 2016
Support for reserving resources	Support for reserving resources, such as meeting rooms and equipment. For more information, see Working with resources in the *Amazon WorkMail Administr ator Guide*.	October 19, 2015
Support for the email migration tool	Support for the email migration tool. For more information, see Migrating to Amazon WorkMail in the *Amazon WorkMail Administr ator Guide*.	August 16, 2015

Preview release of Amazon WorkMail	The preview release of Amazon WorkMail.	January 28, 2015