Automating large-scale server migrations with Cloud Migration Factory

AWS Prescriptive Guidance

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Prescriptive Guidance: Automating large-scale server migrations with Cloud Migration Factory

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

Table of Contents

Introduction **1**
The Cloud Migration Factory workflow **3**
The migration metadata pipeline tool **5**
Automation scripts **6**
How to get started **8**
Prerequisites 8
Migration process **9**
Step 1. Prepare the CSV file and import the data into Cloud Migration Factory 9
Step 2. Build the servers 10
Verify prerequisites for source servers 10
Install the replication agent 11
Push the post-launch script 11
Step 3. Validate the migration 12
Step 4. Perform migration boot-up testing 12
Launch servers for boot-up testing 12
Verify instance boot-up status 13
Step 5. Cut over 13
Verify the replication status 13
Shut down source servers in preparation for cutover 14
Launch target EC2 instances for cutover 14
Verify instance boot-up status 14
(Optional) Get new IP addresses for target instances 15
Test RDP/SSH access to target servers 15
Reconfigure application and networking settings 15
Test the application 15
Complete the cutover 16
FAQ **17**
How do I know if this solution is suitable for my project? 17
How much does it cost to run Cloud Migration Factory? 17
How can I deploy or get support for Cloud Migration Factory? 17
After I deploy Cloud Migration Factory, can AWS help me migrate my servers using this solution? 17
What permissions do I need to deploy the migration factory solution? 18
What permissions do I need to run the automation scripts? 18

What firewall rules do I need to create for the migration execution server? 18
Can I customize the Cloud Migration Factory schema to capture additional attributes? 18
Why doesn't the CSV file include all Amazon EC2 launch template attributes? 18
Can I bulk-update the Amazon EC2 launch templates for all my servers? 19
Does Cloud Migration Factory support SAML federation? 19
What should I do if using Launch Server or Check Status in Cloud Migration Factory displays an "Unexpected timeout" error? 19
Document history 20

Automating large-scale server migrations with Cloud Migration Factory

Wally Lu, Amazon Web Services (AWS)

September 2022 (document history)

Many companies today want to migrate their servers to Amazon Web Services (AWS) as quickly as possible. This is not an easy job, especially if you want to migrate thousands of servers in a short period of time, such as 6–12 months.

Large-scale migrations present a few challenges:

- Integrating multiple tools is difficult. There are many tools that support migration, such as discovery tools, migration tools, and configuration management database (CMDB) tools. These tools must be connected so that data flows from one tool to another. However, different tools use different data formats, and that makes integration difficult. A migration is more likely to be successful if there is a way to integrate all the tools.
- Manual processes are slow and hard to scale. Migrations involve many small tasks, and each task takes a few minutes to complete. A migration is faster when these tasks are automated.

Cloud Migration Factory was designed to solve these problems for migrations that require rehosting (lift and shift). AWS Transform MGN simplifies, expedites, and reduces the cost of cloud migration by offering a highly automated lift-and-shift solution. *Cloud Migration Factory* is an orchestration platform for rehosting servers to AWS at scale. It helps customers with their medium-scale to large-scale migrations by automating manual processes, which are often slow or complex to scale. Thousands of servers have been migrated to AWS using Cloud Migration Factory. For example, AWS customers used Cloud Migration Factory to migrate 1,200 servers in 5 months, and were able to cut over more than 600 servers in a single cutover window.

This guide describes the Cloud Migration Factory process for rehosting servers at scale, for migration architects, program managers, and technical leads. For more information about migration factories, see Mobilize your organization to accelerate large-scale migrations on the AWS Prescriptive Guidance website.

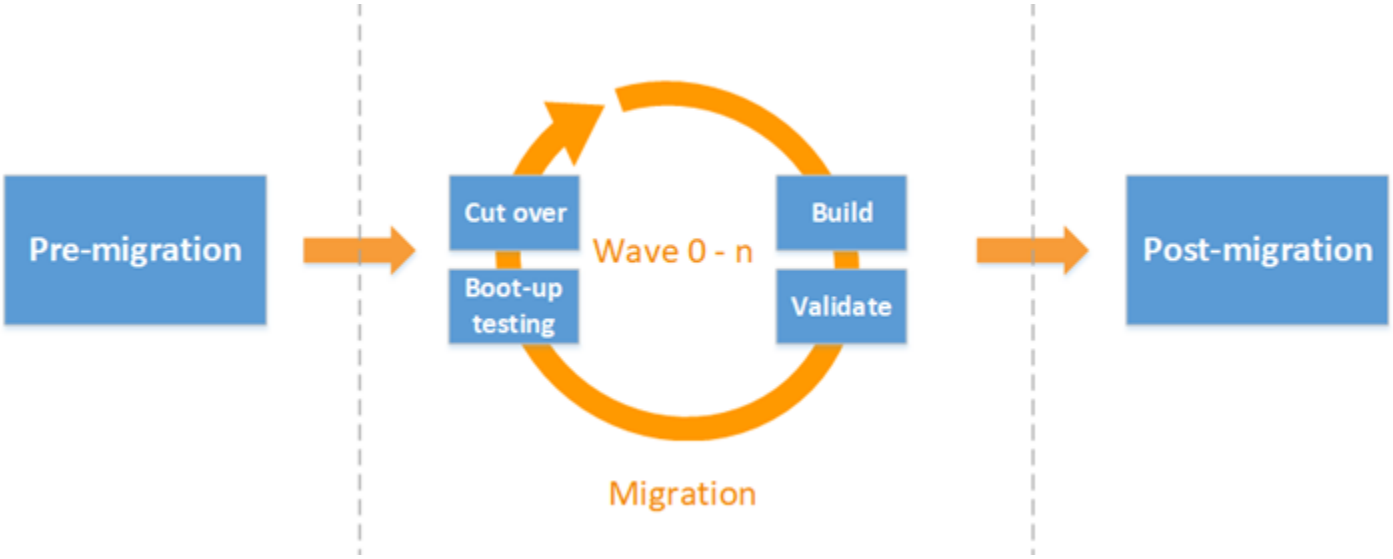
How to get access to Cloud Migration Factory

Cloud Migration Factory is available to all AWS customers and partners. To use Cloud Migration Factory, see the AWS Cloud Migration Factory on the AWS Solutions website. The source code is available in a GitHub repository. If you have any questions, email AWS Professional Services at migration-factory-support@amazon.com.

If you want to get hands-on experience before using Cloud Migration Factory for your production migration, email us at migration-immersion-day@amazon.com to arrange a migration immersion day.

The Cloud Migration Factory workflow

Cloud Migration Factory comes with a predefined process that includes three phases: pre-migration, migration implementation, and post-migration, as shown in the following diagram.

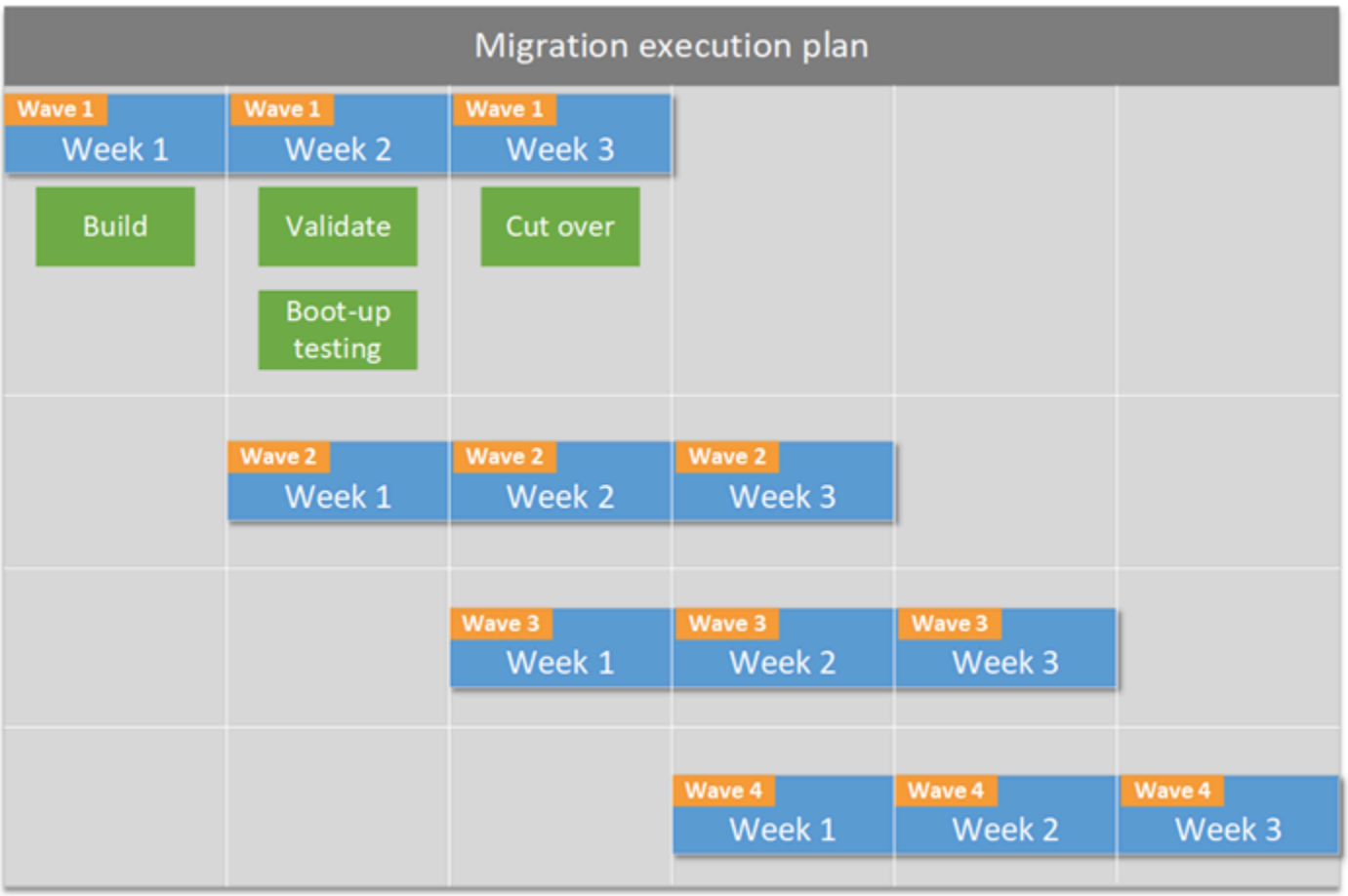


In the **pre-migration phase**, your migration team is responsible for preparing the implementation environment. This includes deploying Cloud Migration Factory, building a migration execution server, and setting up AWS Transform MGN.

During the **migration implementation phase**, the migration team is responsible for running predefined tasks that automate the migration process. These tasks can include:

- Verifying prerequisites
- Pushing the replication agent to the source machines for a given wave
- Verifying replication status
- Launching servers for boot-up testing
- Scheduling a window for application cutover

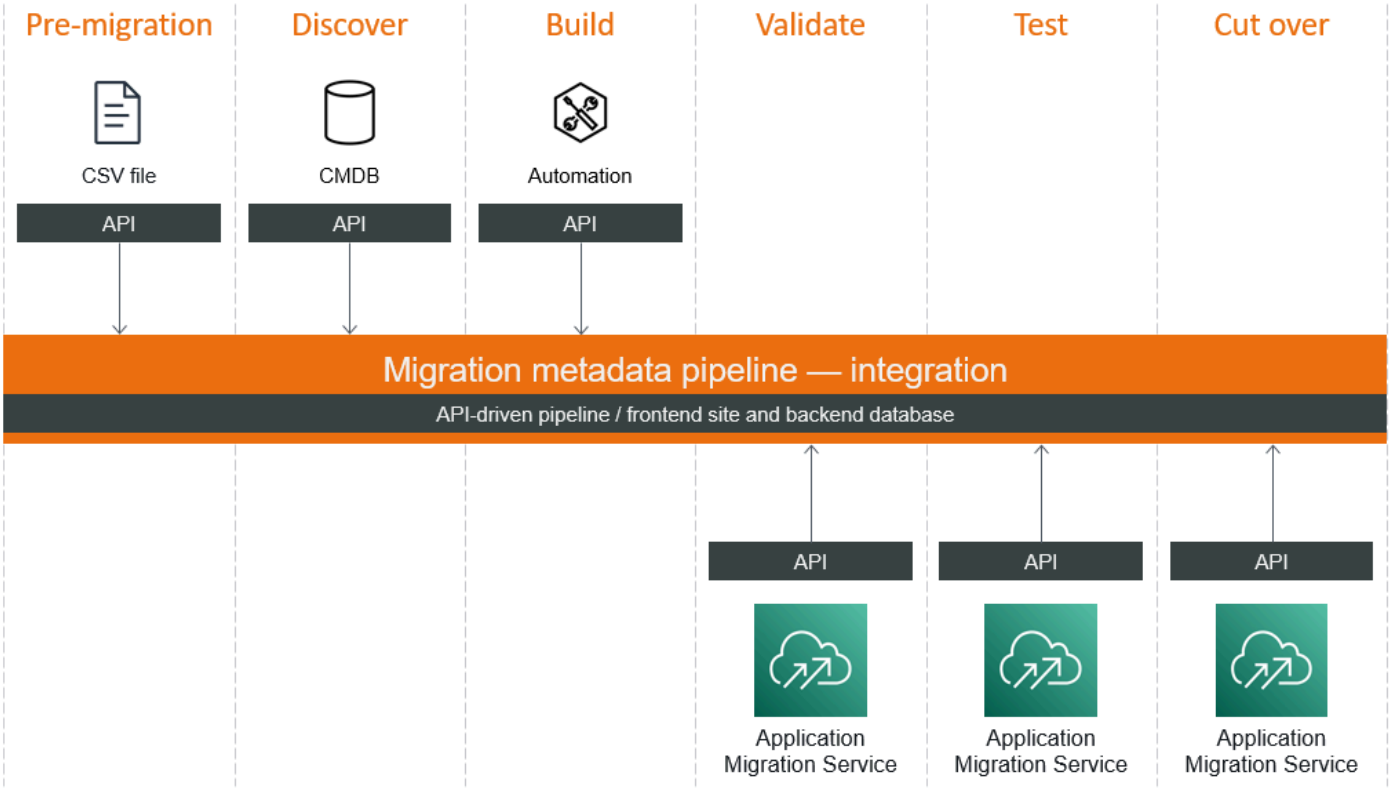
Migration tasks are scheduled in *waves*. Each wave consists of a group of applications and servers that have the same cutover date. As shown in the following diagram, each wave should be completed in a predefined period. For example, in the three-week period shown, week 1 is the build stage, week 2 is the validate and boot-up testing stage, and week 3 is the cutover stage. All the waves run in parallel.



Post-migration tasks depend on the specific migration scenario and your requirements. These tasks might include removing servers from the source CMDB, decommissioning source machines, and optimizing performance for the target Amazon Elastic Compute Cloud (Amazon EC2) instances.

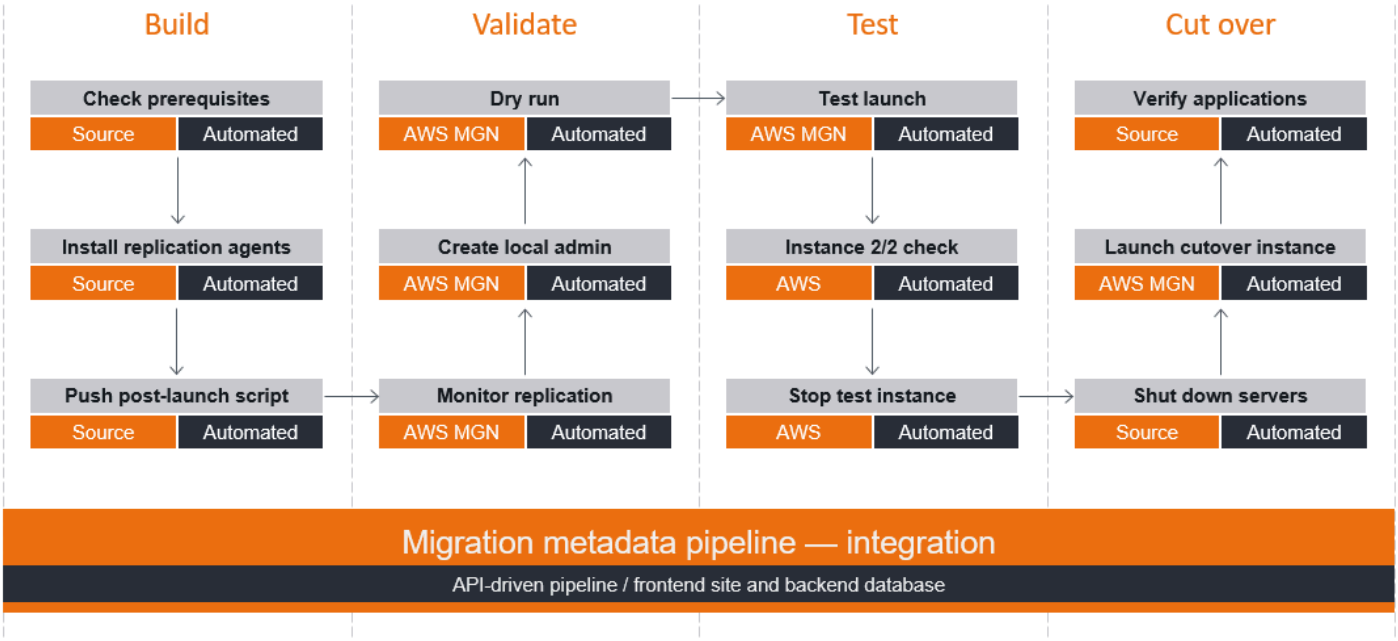
The migration metadata pipeline tool

Cloud Migration Factory includes a migration metadata pipeline tool and automation scripts. The metadata pipeline tool integrates with other migration tools and scripts through Representational State Transfer (REST) APIs, as shown in the following diagram. This enables migration metadata to flow from one tool to another to support end-to-end automation. Currently, Cloud Migration Factory is natively integrated with CloudEndure APIs and the AWS Managed Services (AMS) workload ingest process. By integrating these tools and processes, Cloud Migration Factory can automate migration tasks across multiple tools.



Cloud Migration Factory automation scripts

The following diagram shows the automation scripts included in Cloud Migration Factory. These scripts cover most of the automation tasks for rehost migration using AWS Transform MGN. Automation scripts can be connected to the source machines or to AWS APIs, as shown in the following diagram.



Cloud Migration Factory includes scripts for the following phases and tasks:

- Build phase:
  - Checking prerequisites for the migration
  - Installing replication agents for multiple servers
  - Pushing the post-launch script
- Validate phase:
  - Monitoring replication
  - Creating the local administrator account
  - Performing a dry run
- Test phase:
  - Testing the Amazon Elastic Compute Cloud (Amazon EC2) instance launch

  - Performing 2/2 (system status and instance status) health checks on instances
  - Terminating test instances
- Cutover phase:
  - Shutting down servers at the source location
  - Orchestrating the cutover process
  - Verifying that all application servers are up and running

These automation scripts help you save significant time and effort in your large-scale migration tasks. For example:

- Automating replication agent installation for 100 or more servers. Installing the replication agent on one server takes about 5 minutes. However, if you have 100 servers running Microsoft Windows and Linux for 10 different AWS accounts, there can be 20 different ways to install the agent on the source machine, and this process could take over 500 minutes. The automation script reduces the agent installation time from 500 minutes to less than 5 minutes of operator time, and it works for both Windows and Linux operating systems and any target AWS account.
- Orchestrating the cutover process. This process involves checking the replication status, checking the server status, updating the Amazon EC2 launch templates, launching servers in cutover mode, verifying job status, cleaning up the server, and many other tasks. It is a long process even for one server, and it could be a nightmare if you have hundreds of servers in a single cutover. The Cloud Migration Factory solution automates and orchestrates the entire process for you.

These tasks and the Cloud Migration Factory scripts that automate them are described in more detail in the following sections.

How to get started with Cloud Migration Factory

The Cloud Migration Factory solution is available to all AWS customers and partners, and it can be deployed to your AWS account in just a few minutes. To deploy Cloud Migration Factory, see the AWS Cloud Migration Factory Solution on the AWS Solutions website. The source code is available in a GitHub repository. If you have any questions, email AWS Professional Services at migration-factory-support@amazon.com.

Prerequisites

Cloud Migration Factory requires the following:

- Set up your AWS infrastructure.
- Complete the initial portfolio discovery and wave planning.
- Follow the instructions in the AWS Transform MGN user guide to initialize and configure permissions for this service in the target AWS accounts.
- Follow the instructions in the AWS Cloud Migration Factory Solution implementation guide to deploy Cloud Migration Factory.

After you complete these prerequisites, we can help you complete the steps described in the following sections to perform the migration. If you have multiple waves, you must repeat the steps for each wave. The recommended wave size is 25–35 servers. If you are planning to cut over more (for example, 100 servers) in the same cutover window, we recommend that you split the 100 servers into multiple waves and run the automation multiple times, because smaller waves are easier to troubleshoot from our experience.

The Cloud Migration Factory migration process

The Cloud Migration Factory process consists of five main steps. These are discussed in the following sections:

- Step 1. Prepare the CSV file and import the data into Cloud Migration Factory
- Step 2. Build the servers
- Step 3. Validate the migration
- Step 4. Perform migration boot-up testing
- Step 5. Cut over to the new server instances on AWS

> **Important**
>
> The automation scripts for the migration steps described in this guide require Cloud Migration Factory. To get Cloud Migration Factory, see the AWS Cloud Migration Factory Solution on the AWS Solutions website. If you have any questions, email AWS Professional Services at migration-factory-support@amazon.com.

Step 1. Prepare the CSV file and import the data into Cloud Migration Factory

The first step in a large-scale migration is to prepare the application and server metadata. This metadata is usually collected from portfolio analysis and wave planning, and placed in a comma-separated values (CSV) file. You use the metadata to automate the migration process and to update the Amazon Elastic Compute Cloud (Amazon EC2) launch templates in order to launch target EC2 instances. The default CSV file that we'll provide includes the following attributes:

- `wave_name` – Unique wave ID, based on priority and dependencies.
- `app_name` – Application to be migrated.
- `aws_region` – Target AWS Region for the source machines.
- `aws_accountid` – 12-digit account ID of the destination AWS account.
- `server_name` – On-premises server to be migrated.

- `server_os_family` – Operating system (Windows or Linux) running on the on-premises server.
- `server_os_version` – Version of the server operating system.
- `server_fqdn` – Fully qualified domain name (FQDN) of the server.
- `server_tier` – Type of server (web, application, or database).
- `server_environment` – Host environment for server (development, test, production, QA, preproduction).
- `r_type` – Migration strategy for the source server, such as rehost or replatform.
- `subnet_IDs` – IDs of the AWS subnets to be used for the EC2 instances after cutover.
- `securitygroup_IDs` – IDs of the AWS security groups to be used after cutover.
- `subnet_IDs_test` – IDs of the AWS subnets to be used for testing.
- `securitygroup_IDs_test` – IDs of the AWS security groups to be used for testing.
- `instanceType` – EC2 instance type to be used for the servers.
- `tenancy` – Whether the instance runs on shared hardware, on single-tenant hardware (dedicated), or on an isolated server (Dedicated Host).
- `tags` – AWS tags for target EC2 instances.

Cloud Migration Factory includes an automation script that you run on the migration execution server to import the metadata from the CSV file to Cloud Migration Factory.

For detailed instructions, see Importing data in the *Cloud Migration Factory Implementation Guide*.

Step 2. Build the servers

After you import the application and server metadata, you verify the source machines and install the replication agent to start data replication.

Verify prerequisites for source servers

In this step, you make sure that your source servers have the required configuration to start the data replication. For example, if the source server is a Windows server, it must meet these requirements:

- TCP port 443 outbound must be open for the source machine to connect to the AWS Transform MGN console.

- TCP port 1500 outbound must be open for the source machine to connect to the MGN replication server in the target virtual private cloud (VPC) on AWS.
- The server must be running .NET Framework version 3.5 or later.
- The server must have at least 3 GB of free space on drive C.

Cloud Migration Factory includes an automation script that you run on the migration execution server to verify the prerequisites automatically for all source Windows and Linux servers.

For detailed instructions, see Check prerequisites in the *Cloud Migration Factory Implementation Guide*.

Install the replication agent

After checking the prerequisites, you install the replication agent on the source machines. This process usually takes 5-10 minutes per server, but Cloud Migration Factory includes an automation script to push the agents to all the source servers in the same wave. This script works for multiple target AWS Regions and AWS accounts.

The agent installation script uses the AWS API to pull the installation token for the target AWS account.

For detailed instructions, see Install the replication agents in the *Cloud Migration Factory Implementation Guide*.

Push the post-launch script

One of the common tasks of rehost migration is uninstalling old software such as VMware tools and backup software from target EC2 instances, and installing new software such as the AWS Systems Manager agent. Completing these activities manually could take 15–30 minutes per server, but Cloud Migration Factory automates this process to accelerate the cutover time.

MGN supports post-launch scripts that help you automatically run operating system configuration tasks, such as installing or uninstalling software.

For detailed instructions, see Push the post-launch scripts in the *Cloud Migration Factory Implementation Guide*.

Step 3. Validate the migration

After installing the replication agent on the source machines, you monitor the status of data replication and resolve issues such as permissions or network performance.

If you have a small migration, you can verify the replication status manually from the MGN console. However, if you have large-scale migrations, servers across multiple projects, and servers in multiple waves, this verification can be difficult. For example, if you have 100 servers in wave 1, you must repeat the following steps 100 times to verify their replication status:

- Find the target AWS Region and AWS account for the server.
- Log in to the MGN console, and then search for the server name.
- Check the progress bar and update the status of the server on your spreadsheet.

Cloud Migration Factory includes an automation script that you run once for all servers. The script retries every 5 minutes until the status of every server in wave 1 changes to *Continuous Data Replication*, and it updates the replication status in the Cloud Migration Factory database.

For detailed instructions, see Verify the replication status in the *Cloud Migration Factory Implementation Guide*.

Step 4. Perform migration boot-up testing

After data replication is complete on all servers, you need to test the instance boot-up process and make sure that everything works as expected from the operating system perspective. That is, the EC2 instance must pass the 2/2 (system status and instance status) health checks.

Launch servers for boot-up testing

If you're migrating a small number of servers, you can select the server and launch it directly from the MGN console. However, for large-scale migrations, it's more efficient to launch all the servers together from the Cloud Migration Factory web console. This console provides a single **Launch servers** button to automate the following processes:

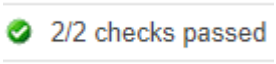
- Verifying replication status and making sure that the lag time is less than 180 minutes.
- Updating the Amazon EC2 launch templates for all servers in the given wave with the metadata in the Cloud Migration Factory database.

- Sending all servers to an MGN job and launching them in test mode.

For detailed instructions, see Launch instances for testing in the *Cloud Migration Factory Implementation Guide*.

Verify instance boot-up status

It will take 15–30 minutes for the server instances to boot up. You can check the status manually by logging into the Amazon EC2 console, searching for the server name, and checking the status. You will see a "2/2 checks passed" message, which indicates that the instance is healthy from an infrastructure perspective.



However, for a large-scale migration, it's time-consuming to check the status of each instance, so Cloud Migration Factory provides a single automation script to verify the 2/2 status for all machines in a given wave.

If an instance fails the 2/2 status checks, contact AWS Support for assistance.

For detailed instructions, see Verify the target instance status in the *Cloud Migration Factory Implementation Guide*.

Step 5. Cut over

The last step in a typical rehost migration is to schedule a cutover window and prepare the resources to support the cutover.

Verify the replication status

First, you must verify the replication status and make sure that the status of all servers in the given wave is healthy.

As in step 3, you can run a Cloud Migration Factory script to automate this step. The script retries every 5 minutes until the status of every server in the given wave changes to healthy, and updates the replication status in the Cloud Migration Factory database.

For detailed instructions, see Verify the replication status in the *Cloud Migration Factory Implementation Guide*.

Shut down source servers in preparation for cutover

After you verify the source servers' replication status, you are ready to shut down the source servers to stop transactions from the client applications to the servers. Usually, you can shut down the source servers in the cutover window. Shutting down the source servers manually could take you 5 minutes per server, and, for large waves, it could take a few hours in total. Instead, you can run a Cloud Migration Factory automation script to shut down all your servers in the given wave.

For detailed instructions, see Shut down the in-scope source servers in the *Cloud Migration Factory Implementation Guide*.

Launch target EC2 instances for cutover

After shutting down the source servers, you can launch the target EC2 server instances. As in step 4, you can use a single **Launch servers** button to launch all the servers in the given wave in cutover mode. The only difference here is that you choose **Cutover** as the launch type. As in boot-up testing, the **Launch servers** button automates the following processes:

- Verifying replication status and making sure that the lag time is less than 180 minutes.
- Updating the Amazon EC2 launch templates for all servers in the given wave with the metadata in the Cloud Migration Factory database.
- Sending all servers to an MGN job and launching them in cutover mode.

For detailed instructions, see Launch instances for cutover in the *Cloud Migration Factory Implementation Guide*.

Verify instance boot-up status

After launching the instances in the cutover mode, wait for at least 15 minutes before the next step, which is verifying instance boot-up status. When cutover launch is complete, you can run the Cloud Migration Factory automation script to verify the 2/2 status for all machines in the given wave.

If an instance fails the 2/2 status checks, contact AWS Support for assistance.

For detailed instructions, see Verify the target instance status in the *Cloud Migration Factory Implementation Guide*.

(Optional) Get new IP addresses for target instances

If the target server instances use new IP addresses, the next step is to update the DNS server with the new IP addresses. In some scenarios, target instances support dynamic DNS registration and register the new IP address automatically with the DNS server. For example, if a Windows server uses a domain controller as the DNS server, DNS registration could be automatic. On the other hand, if the DNS update is a manual process, you need to get the new IP addresses for all the target instances. In this case, you can use the Cloud Migration Factory automation script to export the new IP addresses for all the instances in the given wave to a CSV file.

For detailed instructions, see Retrieve the target instance IP in the *Cloud Migration Factory Implementation Guide*.

Test RDP/SSH access to target servers

After you update the DNS records, you can connect to the target instances with the host name. In this step, you check to see if you can log in to the operating system by using Remote Desktop Protocol (RDP) or through Secure Shell (SSH) access. You can manually log in to each server individually, but it is more efficient to test the server connection by using the Cloud Migration Factory automation script.

For detailed instructions, see Verify the target server connections in the *Cloud Migration Factory Implementation Guide*.

Reconfigure application and networking settings

After the migration team completes operating system level testing, the application team makes changes at the application level. These changes might include the following:

- If the application requires a load balancer, change the application endpoint in the load balancer to point to the new instance IPs in AWS.
- Change the connection string for the application web tier to connect to the database.
- Change other application-specific settings.

Test the application

Application testing, which takes place after the updates described in the previous section, is generally handled by the application owner or support team. It involves logging in to the new

servers and confirming that the application works as expected. If it doesn't, the application owner or support team works with the migration team to troubleshoot and fix issues.

Complete the cutover

This is the final step of the migration. The application owner decides whether the target application in AWS meets their expectations from both functionality and performance perspectives. If a rollback is required, it usually involves these activities:

- Terminating all AWS instances for the affected application.
- Turning on on-premises servers for the given application.
- Reverting DNS records to the old server IP addresses.

Cloud Migration Factory FAQ

This section provides answers to commonly raised questions about deploying and using Cloud Migration Factory.

How do I know if this solution is suitable for my project?

Cloud Migration Factory is suitable for customers who have more than 100 servers that they want to rehost on AWS. Those servers must also be within the list of supported operating systems for AWS Transform MGN.

How much does it cost to run Cloud Migration Factory?

Cloud Migration Factory is a serverless solution that uses AWS services such as Amazon Simple Storage Service (Amazon S3), Amazon CloudFront, Amazon DynamoDB, and AWS Lambda. Your cost depends on your usage.

How can I deploy or get support for Cloud Migration Factory?

To deploy Cloud Migration Factory, see the AWS Cloud Migration Factory Solution on the AWS Solutions website. The source code is available in a GitHub repository. If you have any questions, email AWS Professional Services at migration-factory-support@amazon.com. We will be happy to help you.

After I deploy Cloud Migration Factory, can AWS help me migrate my servers using this solution?

Yes, the AWS Professional Services team can help you. Please email us at migration-factory-support@amazon.com and tell us about your use cases. This process includes:

- An initial conversation to discuss your business case
- A one-hour Cloud Migration Factory introduction and demo session with your team
- Migration immersion day with Cloud Migration Factory, including a half-day lab session followed by Q&A

- Discussion of resources you might need from AWS Professional Services

What permissions do I need to deploy the migration factory solution?

Ideally, you will need to use AWS administrator credentials to deploy the stack. If not, we can provide AWS Identity and Access Management (IAM) policies for you to attach to the user who will deploy the factory.

What permissions do I need to run the automation scripts?

For Windows, you need a domain user with local administrator credentials to access all Windows servers. For Linux, you need a sudo user with `NOPASSWD` enabled.

What firewall rules do I need to create for the migration execution server?

The migration execution server needs to access source servers and external APIs (AWS Transform MGN, Cloud Migration Factory, and AWS APIs). The server needs access to the internet to access these APIs. You can use a proxy server, but if you do, the proxy server can't use authentication. The server needs SSH access to the source Linux server on TCP port 22, or Windows Remote Management (WinRM) access to the source Windows server on TCP port 5985.

Can I customize the Cloud Migration Factory schema to capture additional attributes?

Yes, you can modify or extend the schema to capture additional server or application attributes.

Why doesn't the CSV file include all Amazon EC2 launch template attributes?

Cloud Migration Factory hides uncommon attributes, such as IAM role and public IP. However, you can add these attributes to the CSV file and import them to update the Amazon Elastic Compute Cloud (Amazon EC2) launch templates.

Can I bulk-update the Amazon EC2 launch templates for all my servers?

Yes, you can set hardcoded values for any Amazon EC2 launch template attribute. For example, Cloud Migration Factory sets a private IP for all servers and sets the disk to use gp2 volumes by default, but you can update these attributes.

Does Cloud Migration Factory support SAML federation?

Security Assertion Markup Language (SAML) federation isn't currently supported.

What should I do if using Launch Server or Check Status in Cloud Migration Factory displays an "Unexpected timeout" error?

If you receive a timeout error, your Cloud Migration Factory login token might have expired. Refresh the web page to get a new token.

Document history

The following table describes significant changes to this guide. If you want to be notified about future updates, you can subscribe to an RSS feed.

Change	Description	Date
Removed the CloudEndure Migration service	The CloudEndure Migration service has been discontinued. In the Cloud Migration Factory, it has been replaced by AWS Transform MGN.	September 30, 2022
Updated name of AWS solution	We updated the name of the referenced AWS solution from *CloudEndure Migration Factory* to *Cloud Migration Factory*.	May 6, 2022
CloudEndure Migration Factory is now available as an AWS solution	Added links to the CloudEndure Migration Factory Implementation Guide for step-by-step deployment and migration instructions.	June 4, 2020
Initial publication	—	April 30, 2020