Migration Hub Strategy Recommendations

User Guide

Migration Hub Strategy Recommendations: User Guide

Table of Contents

What is Migration Hub Strategy Recommendations?

Migration Hub Strategy Recommendations helps you plan migration and modernization initiatives by offering migration and modernization strategy recommendations for viable transformation paths for your applications.

Strategy Recommendations performs an analysis of your server inventory, runtime environment, and optionally, source code and database analysis. It combines this analysis with your business goals, and the transformation preferences of the applications and databases provided to recommend:

- The most effective migration strategy for each of your applications.
- Migration and modernization tools or programs that you can use.
- Application incompatibilities and anti-patterns to resolve for a specific option.

Migration Hub Strategy Recommendations recommends migration and modernization strategies for rehosting, replatforming, and refactoring with associated deployment destinations, tools, and programs. For information about rehosting, replatforming, and refactoring, see Migration terms - 7 Rs in the *AWS Prescriptive Guidance* glossary.

Strategy Recommendations might recommend straightforward options, such as rehosting on Amazon Elastic Compute Cloud (Amazon EC2) using AWS Application Migration Service (AWS MGN). More optimized recommendations might include replatforming to containers using AWS App2Container, or refactoring to open source technologies such as .NET Core and PostgreSQL.

Are you a first-time Strategy Recommendations customer?

If this is your first time using Strategy Recommendations, we recommend that you begin by reading the following sections:

- Strategy Recommendations overview (p. 1)
- Setting up Strategy Recommendations (p. 3)
- Getting started with Strategy Recommendations (p. 5)

Strategy Recommendations overview

You can start the assessment for your portfolio of servers and applications by using Migration Hub Strategy Recommendations from the AWS Migration Hub console. You use the console to set up and perform an assessment. After the assessment, you can use the console to view assessment data for each server and application, along with the recommended transformation tool.

To receive refactoring recommendations and a list of incompatibilities, you can use Strategy Recommendations to assess your application source code and databases.

You can also download the recommendations data in a Microsoft Excel file.

Related services

- AWS Migration Hub – You use the AWS Migration Hub console to access the Migration Hub Strategy Recommendations console. It also displays information about the servers that you are collecting data from.
- AWS Application Discovery Service – You use Application Discovery Service to collect data about your servers and applications in the AWS Migration Hub console before using Strategy Recommendations.
- AWS Application Migration Service – AWS Application Migration Service is the primary migration service recommended for lift-and-shift migrations to AWS.
- AWS Database Migration Service – AWS Database Migration Service is a web service you can use to migrate data from your database that is on-premises, on an Amazon Relational Database Service (Amazon RDS) DB instance, or in a database on an Amazon Elastic Compute Cloud (Amazon EC2) instance to a database on an AWS service.
- AWS App2Container – AWS App2Container (A2C) is a command line tool for modernizing .NET and Java applications into containerized applications.
- Porting Assistant for .NET – Use for .NET source code analysis. Porting Assistant for .NET is a compatibility scanner that reduces the manual effort required to port Microsoft .NET Framework applications to .NET Core. The Porting Assistant for .NET assesses the .NET application source code and identifies incompatible APIs and third-party packages.
- End-of-Support Migration Program for Windows Server – End-of-Support Migration Program (EMP) for Windows Server includes tooling to migrate your legacy applications from Windows Server 2003, 2008, and 2008 R2 to newer, supported versions on AWS, without any refactoring.
- AWS Schema Conversion Tool – You can use the AWS Schema Conversion Tool (AWS SCT) to convert your existing database schema from one database engine to another.
- Windows Web Application Migration Assistant – The Windows Web Application Migration Assistant for AWS Elastic Beanstalk is an interactive PowerShell utility that migrates ASP.NET and ASP.NET Core applications from on-premises IIS Windows servers to Elastic Beanstalk.
- Babelfish for Aurora PostgreSQL – Babelfish for Aurora PostgreSQL is a new capability for the Amazon Aurora PostgreSQL-Compatible Edition that enables Aurora to understand commands from applications written for the Microsoft SQL server.

Setting up Strategy Recommendations

Before you use Migration Hub Strategy Recommendations for the first time, complete the following tasks:

Sign up for AWS

In this section, you sign up for an AWS account. If you already have an AWS account, skip this step.

When you sign up for Amazon Web Services (AWS), your AWS account is automatically signed up for all AWS services, including Migration Hub Strategy Recommendations. You are charged only for the services that you use.

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 and entering a verification code on the phone keypad.

Create IAM users

When you create an AWS account, you get a single sign-in identity that has complete access to all of the AWS services and resources in the account. This identity is called the AWS account *root user*. Signing in to the AWS Management Console using the email address and password that you used to create the account gives you complete access to all of the AWS resources in your account.

We strongly recommend that you *not* use the root user for everyday tasks, even the administrative ones. Instead, follow the security best practice Create Individual IAM Users and create an AWS Identity and Access Management (IAM) administrator user. Then, securely lock away the root user credentials and use them to perform only a few account and service management tasks.

In addition to creating an administrative user, you must also create non-administrative IAM users. The following topics explain how to create both types of IAM users.

Topics

Creating an IAM Administrative User

By default, an administrator account inherits all the policies required for accessing Migration Hub Strategy Recommendations.

To create an administrator user

- Create an administrator user in your AWS account. For instructions, see Creating Your First IAM User and Administrators Group in the *IAM User Guide*.

Creating an IAM Non-Administrative User

This section describes how to grant the necessary permissions required for using Strategy Recommendations.

When creating a non-administrative IAM user for use with Strategy Recommendations, we recommend that you create two IAM users:

- To access the console, create a user with both the `AWSMigrationHubFullAccess` and the `AWSMigrationHubStrategyConsoleFullAccess` managed policies attached.
- To access the Strategy Recommendations application data collector, create a user with the `AWSMigrationHubStrategyCollector` managed policy attached.

Alternatively, you can create one user with all three managed policies attached.

IAM managed policies define the level of access to a service by non-administrative IAM users. The AWS Migration Hub `AWSMigrationHubFullAccess` managed policy grants a user access to the Migration Hub console. For more information, see Migration Hub Roles and Policies. For information about the `AWSMigrationHubStrategyConsoleFullAccess` and `AWSMigrationHubStrategyCollector` managed policies, see AWS managed policies for Migration Hub Strategy Recommendations (p. 41).

When creating non-administrative IAM users, follow the security best practice Grant Least Privilege and grant users minimum permissions.

To create a non-administrator IAM user to use with Strategy Recommendations

1. In AWS Management Console, navigate to the IAM console.
2. Create a non-administrator IAM user by following the instructions for creating a user with the console as described in Creating an IAM user in your AWS account in the *IAM User Guide*.

   While following the instructions in the *IAM User Guide*:

   - When on the step about selecting the type of access, select both **Programmatic access** and **AWS Management Console access**.
   - When on the step about the **Set permission** page, choose the option to **Attach existing policies to user directly**. Then, select the managed IAM policy **AWSMigrationHubFullAccess**, **AWSMigrationHubStrategyConsoleFullAccess**, or **AWSMigrationHubStrategyCollector** from the list of policies.
   - When on the step about viewing the user's access keys (access key IDs and secret access keys), follow the guidance in the **Important** note about saving the user's new access key ID and secret access key in a safe and secure place.

Getting started with Strategy Recommendations

This section describes how to get started with Migration Hub Strategy Recommendations.

Topics

Prerequisites for Strategy Recommendations

The following are the prerequisites for using Migration Hub Strategy Recommendations.

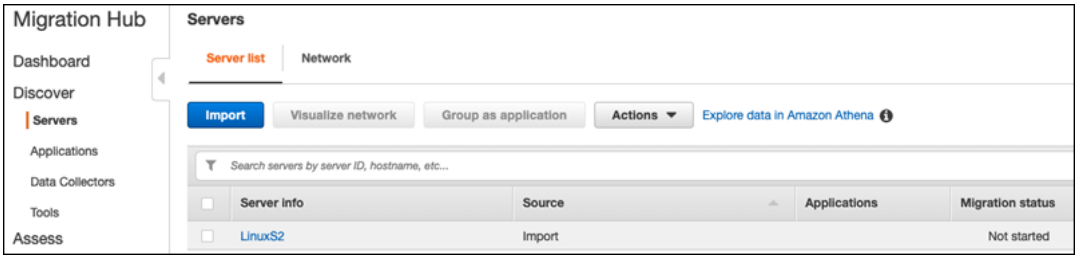
- You must have one or more AWS accounts, and IAM users set up for these accounts. For more information, see Setting up Strategy Recommendations (p. 3).
- The Strategy Recommendations application data collector client must be able to collect data remotely from servers. This requires that you use a set of credentials that work for all your Windows servers and a set of credentials that work for all of your Linux servers.
- The version of the collector that is deployed in vCenter supports VMware vCenter Server V6.0, V6.5, 6.7 or 7.0.

  You can also deploy the collector in an Amazon EC2 instance using the collector AMI.
- Verify that your operating system (OS) environment is supported:
  - **Linux**
    - Amazon Linux 2012.03, 2015.03
    - Amazon Linux 2 (9/25/2018 update and later)
    - Ubuntu 12.04, 14.04, 16.04, 18.04, 20.04
    - Red Hat Enterprise Linux 5.11, 6.10, 7.3, 7.7, 8.1
    - CentOS 5.11, 6.9, 7.3
    - SUSE 11 SP4, 12 SP5
  - **Windows**
    - Windows Server 2008 R1 SP2, 2008 R2 SP1
    - Windows Server 2012 R1, 2012 R2
    - Windows Server 2016
    - Windows Server 2019
- For source code analysis, your GitHub and GitHub Enterprise repositories must have a personal access token with the **repo** scope that can be shared with the Strategy Recommendations collector client. For more information about creating a personal access token with the **repo** scope, see Creating a personal access token in the *GitHub Docs*.

To analyze .NET repositories for Porting Assistant for .NET recommendations, you must provide a Windows machine that is set up with the Porting Assistant for .NET porting assessment tool. For more information, see Getting started with Porting Assistant for .NET in the *Porting Assistant for .NET User Guide*.

- To enable Strategy Recommendations for database analysis, you must enter credentials in AWS Secrets Manager. For more information, see Strategy Recommendations database analysis (p. 21).
- You must use AWS Application Discovery Service to collect data about your servers and applications in the AWS Migration Hub console before using Strategy Recommendations. You can use one of the following methods to collect the data.
  - **Migration Hub import** – With Migration Hub import, you can import information about your on-premises servers and applications into Migration Hub. For more information, see Migration Hub Import in the *Application Discovery Service User Guide*.
  - **AWS Agentless Discovery Connector** – The Discovery Connector is a VMware appliance that collects information about VMware virtual machines (VMs). For more information, see AWS Agentless Discovery Connector in the *Application Discovery Service User Guide*.
  - **AWS Application Discovery Agent** – The Discovery Agent is AWS software that you install on your on-premises servers and VMs to capture system information and details of the network connections between systems. For more information, see AWS Application Discovery Agent in the *Application Discovery Service User Guide*.

Note
To verify that the Migration Hub import completed successfully, in the Migration Hub console navigation pane, under **Discover**, choose **Servers**. All the imported servers should be listed.



Step 1: Download the Strategy Recommendations collector

Migration Hub Strategy Recommendations application data collector is a virtual appliance that you can install in your on-premises VMware environment. The Strategy Recommendations application data collector is also available as an Amazon Machine Image (AMI). If you want to use the AMI version of the collector to assess AWS applications or for some other reason, you don't need to download the collector. You can skip this section and go to Deploy the Strategy Recommendations collector in an Amazon EC2 instance (p. 8).

This section describes how to download the collector Open Virtualization Archive (OVA) file that you use to deploy the collector as a virtual machine (VM) in your VMware environment.

To download the collector OVA file

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy**.

3. On the **Migration Hub Strategy Recommendations** page, choose **Download data collector**.
4. Optionally, you can choose **Download the import template** if you want to import application data. For more information about importing data, see Importing data into Strategy Recommendations (p. 27).
5. Click on **Get recommendations** button and choose **Agree** to allow Migration Hub to create a service-linked role (SLR) in your account. When setting up Strategy Recommendations for the first time, you must create the SLR. For more information, see Using service-linked roles for Strategy Recommendations (p. 50).

Step 2: Deploy the Strategy Recommendations collector

This section describes how to deploy the Strategy Recommendations application data collector. An application data collector is an agentless data collector that identifies running applications on your servers, performs source code analysis, and analyzes your databases.

There are two ways to deploy the collector:

- Deploy as a virtual machine (VM) in your VMware vCenter Server. For more information, see Deploy the Strategy Recommendations collector in vCenter (p. 7).
- If you have AWS applications that you want to assess, you can use the Strategy Recommendations collector Amazon Machine Image (AMI). For more information, see Deploy the Strategy Recommendations collector in an Amazon EC2 instance (p. 8).

Deploy the Strategy Recommendations collector in vCenter

Migration Hub Strategy Recommendations application data collector is a virtual appliance that you can install in your on-premises VMware environment. This section describes how to deploy the collector Open Virtualization Archive (OVA) file as a virtual machine (VM) in your VMware environment.

The following procedure describes how to deploy the Strategy Recommendations collector in your VMware vCenter Server environment.

To deploy the collector in vCenter

1. Sign in to vCenter as a VMware administrator.
2. Deploy the OVA file that you downloaded in Step 1. The OVA file includes the collector and a CLI that can be used to access the Strategy Recommendations API.

   You can also download the OVA file from the following link:

   https://application-data-collector-release.s3.us-west-2.amazonaws.com/ova/latest/AWSMHubApplicationDataCollector.ova

We recommend the following specifications for the VM.

Strategy Recommendations collector VM specifications

- **RAM** – a minimum of 8 GB
- **CPUs** – at least 4

Note
To ensure that you are using the latest version of the collector with all the new features and bug fixes, upgrade the collector after you deploy the collector OVA file. For instructions about how to upgrade, see Upgrading the Strategy Recommendations collector (p. 27).

Deploy the Strategy Recommendations collector in an Amazon EC2 instance

If you have AWS applications that you would like to assess, you can use the Strategy Recommendations application data collector Amazon Machine Image (AMI).

The following procedure describes how to launch an Amazon EC2 instance from the collector AMI.

To deploy the collector Amazon EC2 instance

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation bar at the top of the screen, the current Region is displayed (for example, US East (Ohio)). Choose a Region that suits your needs from the Regions that Strategy Recommendations uses. For a list of these Regions, see Strategy Recommendations endpoints in the *AWS General Reference*.
3. In the navigation pane, under **Images** choose **AMIs**.
4. Choose **Public images** from the **Owned by me** dropdown.
5. Choose the search bar and select **AMI Name** from the menu.
6. Enter the name **AWSMHubApplicationDataCollector AMI**.
7. To ensure that the AMI is from a secure source, verify that the owner of the account is **703163444405**.
8. To launch an instance from this AMI, select it, and then choose **Launch**. For more information about launching an instance using the console, see Launching your instance from an AMI in the *Amazon EC2 User Guide for Linux Instances*.

   We recommend the following specifications for the Amazon EC2 instance.

   Strategy Recommendations collector Amazon EC2 instance specifications

   - **RAM** – A minimum of 8 GB
   - **CPUs** – At least 4

The Strategy Recommendations AMI includes the collector and a CLI that can be used to access the Strategy Recommendations API.

Note
To ensure that you are using the latest version of the collector with all the new features and bug fixes, upgrade the collector after you deploy the Strategy Recommendations collector as an Amazon EC2 instance. For instructions about how to upgrade, see Upgrading the Strategy Recommendations collector (p. 27).

Step 3: Sign in to the Strategy Recommendations collector

This section describes how to sign in to the deployed Migration Hub Strategy Recommendations application data collector. How you sign in to the collector depends on how you deployed it.

Sign in to the collector deployed in the vCenter based environment

To sign in to the Strategy Recommendations collector deployed in the vCenter based environment

1. Use the following command to connect to the collector using an SSH client.

```
ssh ec2-user@CollectorIPAddress
```

2. When prompted for a password, enter the default password **aq1@WSde3**. You must change the password the first time you sign in.

Sign in to the collector deployed as an Amazon EC2 instance

To sign in to the Strategy Recommendations collector deployed as an Amazon EC2 instance

- Use the following command to connect to the collector using an SSH client.

```
ssh -i "Keyname.pem" ec2-user@CollectorIPAddress
```

  Keyname.pem is the private key that was generated when you launched the Amazon EC2 instance from the collector AMI.

Step 4: Set up the Strategy Recommendations collector

This section describes how to use the command line `collector setup` commands to configure the Migration Hub Strategy Recommendations application data collector. These configurations are stored locally.

Before you can use `collector setup` commands, you must create a bash shell session in the collector Docker container using the following `docker exec` command.

```
docker exec -it application-data-collector bash
```

The `collector setup` command runs all of the following commands in succession but you can run them individually:

- `collector setup --aws-configurations` – Set up AWS configurations.
- `collector setup --vcenter-configurations` – Set up vCenter configurations.

Note
vCenter configuration setup is only available if the collector is hosted on vCenter. However, you can force vCenter configuration setup by using the command `collector setup --vcenter-configurations`.

- `collector setup --remote-server-configurations` – Set up remote server configurations.
- `collector setup --version-control-configurations` – Set up version control configurations.

To set up all the collector configurations at the same time

1. Enter the following command.

```
collector setup
```

2. Enter the information for AWS configurations as described in Set up AWS configurations (p. 10).
3. Enter the information for vCenter configurations as described in Set up vCenter configurations (p. 11).
4. Enter the information for remote server configurations as described in Set up remote server configurations (p. 12).
5. Enter the information for version control configurations as described in Set up version control configurations (p. 14).
6. Prepare your Windows and Linux servers for collector data collection by following the instructions in Prepare your remote Windows and Linux servers for data collection (p. 15).

Set up AWS configurations

To set up AWS configurations, when using the `collector setup` command or the `collector setup --aws-configurations` command.

1. Enter **Y** for yes to the **Have you setup IAM permissions...** question. You set up these permissions when you created an IAM user to access the collector using the `AWSMigrationHubStrategyCollector` managed policy following the steps in Creating an IAM Non-Administrative User (p. 4).
2. Enter your access key and secret key from the AWS account that has the IAM user that you created to access the collector following the steps in Creating an IAM Non-Administrative User (p. 4).
3. Enter a Region, for example, `us-west-2`. Choose a Region that suits your needs from the Regions that Strategy Recommendations uses. For a list of these Regions, see Strategy Recommendations endpoints in the *AWS General Reference*.
4. Enter **Y** for yes to the **Upload collector related metrics to migration hub strategy service?** question. Metrics information helps AWS provide you with appropriate support.
5. Enter **Y** for yes to the **Upload collector related logs to migration hub strategy service?** question. Information from logs helps AWS provide you with appropriate support.

The following example shows what displays, including example entries for the AWS configurations.

```
Have you setup IAM permissions in you AWS account as per the user guide? [Y/N]: Y
Choose one of the following options for providing user credentials:
1. Long term AWS credentials
2. Temporary AWS credentials
Enter your options [1-2]: 2
```

```
AWS session token:
AWS access key ID [None]:
AWS secret access Key [None]:
AWS region name [us-west-2]:
AWS configurations are saved successfully
Upload collector related metrics to migration hub strategy service? By default collector
 will upload metrics. [Y/N]: Y
Upload collector related logs to migration hub strategy service? By default collector will
 upload logs. [Y/N]: Y
Application data collector configurations are saved successfully
Start registering application data collector
Application data collector is registered successfully.
```

Set up vCenter configurations

To set up vCenter configurations, when using the `collector setup` command or the `collector setup --vcenter-configurations` command:

1. Enter **Y** for yes to the **Would you like to authenticate using VMware vCenter credentials** question, if you want to authenticate using VMware vCenter credentials.

   Note
   Authenticating using VMware vCenter credentials requires that VMware tools are installed on the target servers.

   Enter the **Host Url**, which can be either the vCenter IP address or URL. Then, enter the **Username** and **Password** for VMware vCenter.

2. Enter **Y** for yes to the **Do you have Windows machines managed by VMware vCenter** question, if you want to configure Windows servers.

   Enter the **Username** and **Password** for Windows.

   Note
   If your Windows Remote Server belongs to an Active Directory domain, you must enter the user name as *domain-name**username* when using the CLI to provide remote server configurations. For example, if the name of your domain is exampledomain and your user name is Administrator, then the user name you enter in the CLI is **exampledomain\Administrator**.

3. Enter **Y** for yes to the **Setup for Linux using VMware vCenter** question, if you want to configure Linux servers.

   Enter the **Username** and **Password** for Linux.

4. Enter **Y** for yes to the **Would you like to setup credentials for servers outside vCenter using NTLM for Windows** and **SSH/Cert based for Linux** questions, if you want to set up remote server credentials for servers outside of vCenter.

5. For the **Would you like to use the same Windows credentials used during vCenter setup** question, enter **Y** for yes if the credentials for the Windows machines managed outside of vCenter are the same as the credentials provided when configuring credentials for vCenter Windows machines. Otherwise, enter **N** for no.

   If you answer **Y** for yes, the following questions are asked.

   a. Enter **Y** for yes to the **Are you okay with collector accepting and locally storing server certificates on your behalf during first interaction with windows servers?** question.

   b. Enter **1** for the **Enter your options** question, if you want to configure for SSH authentication.

If you choose to use SSH authentication, you must copy the generated key credentials to your Linux servers. For more information, see Set up key-based authentication on Linux servers (p. 15).

The following example shows what displays and example entries for the VMware vCenter configurations.

```
Start setting up vCenter configurations for remote execution
Note: authenticating using VMware vCenter credentials requires VMware tools to be installed
 on the target servers
Would you like to authenticate using VMware vCenter credentials? [Y/N]: Y
Host Url for VMware vCenter: host-url
Username for VMware vCenter: username
Password for VMware vCenter:
Successfully stored vCenter credentials...
Setup for Windows using VMware vCenter? [Y/N]: Y
Username for Windows: username
Password for Windows:
Successfully stored vCenter windows credentials...
Setup for Linux using VMware vCenter? [Y/N]: Y
Username for Linux: username
Password for Linux:
Successfully stored vCenter linux credentials...
Would you like to setup credentials for servers outside vCenter using NTLM for windows and
 SSH/Cert based for linux? [Y/N]: Y
Would you like to use the same Windows credentials used during vCenter setup? [Y/N]: Y
Are you okay with collector accepting and locally storing server certificates on your
 behalf during first interaction with windows servers? These certificates will be used by
 collector for secure communication with windows servers [Y/N]:Y
Successfully stored windows server credentials...
Please note that all windows server certificates are stored in directory /opt/amazon/
application-data-collector/remote-auth/windows/certs

Please note the IP address of the collector and run the script specified in the user
 documentation on all the windows servers in your inventory
Would you like to setup credentials for servers not managed by vCenter using SSH/Cert based
 for Linux? [Y/N]: Y
Choose one of the following options for remote authentication:
1. SSH based authentication
2. Certificate based authentication
Enter your options [1-2]: 1
Would you like to use the same Linux credentials used during vCenter setup? [Y/N]: Y
Generating SSH key on this machine...
SSH key pair path: /opt/amazon/application-data-collector/remote-auth/linux/keys/
id_rsa_assessment
Please add the public key "id_rsa_assessment.pub" to the "$HOME/.ssh/authorized_keys" file
 in your remote machines.
Your Linux remote server configurations are saved successfully.
```

Set up remote server configurations

To set up remote server configurations, when using the `collector setup` command or the `collector setup --remote-server-configurations` command:

1. Enter **Y** for yes to the **Would you like to setup credentials for servers not managed by vCenter using NLTM for Windows** question, if you want to configure Windows servers.

   Enter the **Username** and **Password** for WinRM.

Note
If your Windows Remote Server belongs to an Active Directory domain, you must enter the user name as *domain-name**username* when using the CLI to provide remote server configurations. For example, if the name of your domain is exampledomain and your user name is Administrator, then the user name you enter in the CLI is **exampledomain \Administrator**.

Enter **Y** for yes to the **Are you okay with collector accepting and locally storing server certificates on your behalf during first interaction with windows servers?** question. Windows Server certificates are stored in the directory `/opt/amazon/application-data-collector/remote-auth/windows/certs`.

You must copy the generated server credentials to your Windows servers. For more information, see Set up remote server configuration on Windows servers (p. 16).

2. Enter **Y** for yes to the **Setup for Linux using SSH or Cert** question, if you want to configure Linux servers.
3. Enter **1** for the **Enter your options** question, if you want to configure for SSH key based authentication.

   If you choose to use SSH authentication, you must copy the generated key credentials to your Linux servers. For more information, see Set up key-based authentication on Linux servers (p. 15).
4. Enter **2** for the **Enter your options** question, if you want to configure for certificate-based authentication.

   For information about setting up certificate-based authentication, see Set up certificate-based authentication on Linux servers (p. 15).

The following example shows what displays and example entries for the remote server configurations.

```
Setting up target server for remote execution
Would you like to setup credentials for servers not managed by vCenter using NLTM for
 Windows [Y/N]: Y
Username for WinRM: username //Enter domain-name\username, if the server is in AD domain
Password for WinRM: password
Are you okay with collector accepting and locally storing server certificates on your
 behalf during first interaction with windows servers? These certificates will be used by
 collector for secure communication with windows servers [Y/N]: Y
Successfully stored windows server credentials...
Please note that all windows server certificates are stored in directory /opt/amazon/
application-data-collector/remote-auth/windows/certs

Please note the IP address of the collector and run the script specified in the user
 documentation on all the windows servers in your inventory
Would you like to setup credentials for servers not managed by vCenter using SSH/Cert based
 for Linux? [Y/N]: Y
Choose one of the following options for remote authentication:
1. SSH based authentication
2. Certificate based authentication
Enter your options [1-2]: 1
User name for remote server: username
Generating SSH key on this machine...
SSH key pair path: /opt/amazon/application-data-collector/remote-auth/linux/keys/
id_rsa_assessment
Please add the public key "id_rsa_assessment.pub" to the "$HOME/.ssh/authorized_keys" file
 in your remote machines.
Your Linux remote server configurations are saved successfully.
```

Set up version control configurations

To set up version control configurations, when using the `collector setup` command or the `collector setup --version-control-configurations` command:

1. Enter **Y** for yes to the **Set up source code analysis?** question.
2. Enter **1** for the **Enter your options** question, if you want to configure the Git server endpoint.

   Enter **github.com** for the **GIT server endpoint:**.
3. Enter **2** for the **Enter your options** question, if you want to configure a GitHub Enterprise Server.

   Enter the enterprise endpoint without https://, as follows: **GIT server endpoint:** *`git-enterprise-endpoint`*
4. Enter your Git *`username`* and personal access *`token`*.
5. Enter **Y** for yes to the **Do you have any csharp repositories that should be analyzed on a windows machine?** question, if you want to analyze C# code.

   Note
   To analyze .NET repositories for Porting Assistant for .NET recommendations, you must provide a Windows machine that is set up with the Porting Assistant for .NET porting assessment tool. For more information, see Getting started with Porting Assistant for .NET in the *Porting Assistant for .NET User Guide*.
6. For the **Would you like to reuse existing windows credentials on this machine?** question. Enter **Y** for yes, if the Windows machine for C# source code analysis uses the same credentials as the credentials previously provided as part of setting up `--remote-server-configurations` or `--vcenter-configurations`.

   Enter **N** for no, if you want to enter new credentials.
7. To use **VMWare vCenter Windows Machine** credentials, enter **1** for **Choose one of the following options for windows credentials**.
8. Enter the IP address for the Windows machine.

The following example shows what displays and example entries for the version control configurations.

```
Set up for source code analysis [Y/N]: y
Choose one of the following options for version control type:
1. GIT
2. GIT Enterprise
3. Azure DevOps - Git
Enter your options [1-3]: 3
Your server endpoint: dev.azure.com (http://dev.azure.com/)
Your DevOps Organization name: <Your organization name>
Personal access token [None]:
Your version control credentials are saved successfully.
Do you have any csharp repositories that should be analyzed on a windows machine? [Y/N]: y
Would you like to reuse existing windows credentials on this machine? [Y/N]: y
Choose one of the following options for windows credentials:
1. VMWare vCenter Windows Machine
2. Standard Windows Machine
Enter your options [1-2]:
1
Windows machine IP Address: <Your windows machine IP address>
Using VMWare vCenter Windows Machine credentials
Successfully stored windows server credentials...
```

Prepare your remote Windows and Linux servers for data collection

> **Note**
> This step isn't necessary if you setup the Strategy Recommendations applications data collector using vCenter credentials.

After you set up your remote server configurations, if you are using the `collector setup command` or the `collector setup --remote-server-configurations` command, you must prepare your remote servers so that the Strategy Recommendations applications data collector can collect data from them.

> **Note**
> You must make sure that the servers are reachable using their private IP address. For further instructions on how to set up the environment through a virtual private cloud (VPC) on AWS for remote running, see the Amazon Virtual Private Cloud User Guide.

To prepare your remote Linux servers, see Prepare remote Linux servers (p. 15).

To prepare your remote Windows servers, see Set up remote server configuration on Windows servers (p. 16).

Prepare remote Linux servers

Set up key-based authentication on Linux servers

If you choose to set up SSH key-based authentication for Linux when configuring remote server configurations, you must perform the following steps to set up key-based authentication on your servers so that data can be collected by the Strategy Recommendations applications data collector.

To set up key-based authentication on your Linux servers

1. Copy the public key generated with the name **id_rsa_assessment.pub** from the following folder in the container:

   /opt/amazon/application-data-collector/remote-auth/linux/keys.
2. Append the copied public key in the `$HOME/.ssh/authorized_keys` file for all the remote machines. If there is no file available, create it using the `touch` or `vim` command.
3. Make sure that the home folder on the remote server has permission level `755` or less. If it's `777`, it won't work. You can use the `chmod` command to restrict permissions.

Set up certificate-based authentication on Linux servers

If you choose to set up certificate-based authentication for Linux when configuring remote server configurations, you must perform the following steps so that data can be collected by the Strategy Recommendations application data collector.

We recommend this option if you already have Certificate Authority (CA) set up for your application servers.

To set up certificate-based authentication on your Linux servers

1. Copy the user name that works with all your remote servers.
2. Copy the public key of the collector to the CA.

The public key for the collector can be found in the following location:

/opt/amazon/application-data-collector/remote-auth/linux/keys/id_rsa_assessment.pub

This public key must be added to your CA for generating the certificate.

3. Copy the certificate generated in the previous step to the following location in the collector:

   /opt/amazon/application-data-collector/remote-auth/linux/keys

   The name of the certificate must be **id_rsa_assessment-cert.pub**.
4. Provide the certificate file name during the setup step.

Set up remote server configuration on Windows servers

If you choose to set up Windows when configuring remote server configurations in the collector setup, you must perform the following steps so that data can be collected by Strategy Recommendations.

To understand more about the PowerShell script that is executed on the remote server, read this note.
The script enables PowerShell remote and disables all authentication methods other than negotiate. This is used for Windows NT LAN Manager (NTLM) and sets the "AllowUnencrypted" WSMan protocol to false to ensure that the newly created listener accepts only encrypted traffic. Using the Microsoft provided script, `New-SelfSignedCertificateEx.ps1`, it creates a self-signed certificate.
Any WSMan Instance that has a HTTP listener is removed along with existing HTTPS listeners. Then, it creates a new HTTPS listener. It also creates an inbound firewall rule for TCP port 5986. In the final step, the WinRM service is restarted.

To set up data collection through a remote connection on your Windows 2008 servers

1. Use the following command to check the version of PowerShell installed on your server.

```
$PSVersionTable
```

2. If the PowerShell version is not 5.1, then download and install WMF 5.1 by following the instructions at Install and Configure WMF 5.1 in the Microsoft documentation.
3. Use the following command in a new PowerShell window to ensure that PowerShell 5.1 is installed.

```
$PSVersionTable
```

4. Follow the next set of steps, which describe how to set up data collection through a remote connection on Windows 2012 and above.

To set up data collection through a remote connection on your Windows 2012 and newer servers

1. Download the setup script from the following URL:

   https://application-data-collector-release.s3.us-west-2.amazonaws.com/scripts/WinRMSetup.ps1
2. Download the `New-SelfSignedCertificateEx.ps1` from the following URL and paste the script into the same folder in which you downloaded `WinRMSetup.ps1`:

   https://github.com/Azure/azure-libraries-for-net/blob/master/Samples/Asset/New-SelfSignedCertificateEx.ps1
3. To complete the setup, run the downloaded PowerShell script on all application servers.

```
.\WinRMSetup.ps1
```

Note
If Windows Remote Management (WinRM) is not set up properly on the Windows Remote Server, an attempt to collect data from that server will fail. If this happens, you must delete the certificate that corresponds to that server from the following location on the container: **/opt/amazon/application-data-collector/remote-auth/windows/certs/*ads-server-id*.cer** After you delete the certificate, wait for the data collection process to be retried.

Step 5: Use Strategy Recommendations in the Migration Hub console to get recommendations

This section describes how to use Strategy Recommendations in the Migration Hub console to get migration recommendations for the first time.

To get recommendations

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy**.
3. On the **Migration Hub Strategy Recommendations** page, choose **Get recommendations**.
4. Choose **Agree** if you agree to allow Migration Hub to create a service-linked role (SLR) in your account. For more information about the SLR, see Using service-linked roles for Strategy Recommendations (p. 50).
5. **Configure data sources**

   a. On the **Configure data sources** page, the application data collectors that you've set up are listed under **Registered application data collectors**. If you haven't set up any collectors yet, you can download the collector from this page and then go to Step 2: Deploy the Strategy Recommendations collector (p. 7).

      To get strategy recommendations, you must set up at least one application data collector or an application data import. If you want to add your application-level data without setting up a collector, use the application data import template. You can add more data sources at a later time.

   b. Under **Import details**, choose **Add new import** to upload your filled out import template file. Specify the name for your import and the S3 bucket URI for your import file.

   c. Choose **Next**.
6. **Specify preferences**

   a. On the **Specify preferences** page, set up your business goals and migration preferences. Strategy Recommendations recommends the optimal strategy for migrating and modernizing your applications and databases based on the preferences that you specify. You can change these preferences at a later time.

   b. Choose **Next**.
7. **Review and submit**.

   a. Review your configured data sources and migration preferences.

   b. If everything looks correct, choose **Start data analysis**.

Strategy Recommendations recommendations

This section describes how to view Strategy Recommendations migration and modernization recommendations for servers and applications in your migration portfolio.

Topics

Viewing strategy recommendations in Strategy Recommendations

This section describes how to use Strategy Recommendations in the AWS Migration Hub console to view migration strategy recommendations.

To view strategy recommendations

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Recommendations**.
3. On the **Recommendations** page, you can view and export summary recommendations of your portfolio and detailed migration "R" strategy recommendations. You can also view migration and modernization tools and destinations, and anti-patterns for your servers and application components.

   Anti-patterns are a list of known issues found in your portfolio that are categorized by severity. High severity anti-patterns represent incompatibilities that need to be resolved, medium severity anti-patterns represent warnings, and low severity anti-patterns represent informational issues. For information about the "R" strategy, see Migration terms - 7 Rs in the *AWS Prescriptive Guidance* glossary.

   - If a change occurs in your data center or if you update your preferences, we recommend reanalyzing your data. To reanalyze your data to get new recommendations, choose **Reanalyze data**.

     Until the reanalyze process completes, your recommendation data results can be a mix of prior data and new data.

     To download a report file with the recommendations, Choose **Export recommendations**.
4. On the **Application components** tab, you can view the recommendations for application components in your migration portfolio. For more information, see Strategy Recommendations application component recommendations (p. 19).
5. On the **Servers** tab, you can view the recommendations for the servers in your migration portfolio. For more information, see Strategy Recommendations server recommendations (p. 22).

6. On the **Preferences** tab, you can edit the preferences you specified in Step 5: Get recommendations (p. 17). For information about editing your preferences, see Strategy Recommendations preferences (p. 23).

Strategy Recommendations application component recommendations

This section describes how to use Strategy Recommendations in the Migration Hub console to view and analyze migration strategy recommendations for application components.

Topics

Working with application components in Strategy Recommendations

This section describes how to use Migration Hub Strategy Recommendations in the Migration Hub console to view and configure migration and modernization strategy recommendations.

Topics

Viewing application component recommendations

This section describes how to use Strategy Recommendations in the Migration Hub console to view migration strategy recommendations for application components.

To view recommendations details for application components

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Recommendations**.
3. On the **Recommendations** page, choose the **Application components** tab.
    a. Under **Application components summary**, is an overview of the various types of application components that you are running in your server portfolio.
    b. Under **Application components**, you view component name, component type, and migration "R" strategy recommendations. You can also view the migration destination, and the migration and modernization tools to use for various application components that are running in your server portfolio. For information about the "R" strategy, see Migration terms - 7 Rs in the *AWS Prescriptive Guidance* glossary.
4. To view the details for an application component, select an application component and then choose **View details**.

5. On the application component details page (the page with the component's name as the heading) under **Recommendation summary**, you can view **Recommendations** for the application component. You can also view identified **Anti-patterns**. Anti-patterns are a list of known issues found in your portfolio that are categorized by severity.
6. Choose the **Strategy options** tab to view the migration recommendation for the application component. You can override the recommended strategy by selecting a different strategy and then choosing **Set preferred**.
7. Depending on which type of application component you are viewing, there is a **Source configuration** or a **Database configuration** tab. For information about **Source configuration**, see Configure source code analysis for an application component (p. 20). For information about **Database configuration**, see Configure database analysis for an application component (p. 20).

Configure source code analysis for an application component

This section describes how to use Strategy Recommendations in the Migration Hub console to configure source code analysis for an application component.

To configure source code analysis for an application component

1. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Recommendations**.
2. On the **Recommendations** page, choose the **Application components** tab.
3. From the list of components under **Application components**, select an application component with a component type of **java**, **dotnetframework**, or **IIS**, and then choose **View details**.
4. On the application component details page (the page with the component's name as the heading), choose the **Source code configuration** tab.
5. Under **Source code configuration details**, choose **Analyze source code**.
6. On the **Analyze source code** page, provide the repository name, branch name, and project name (if applicable) that stores the source code for the application component. Select the type of GitHub source code version control that you want to use, and then choose **Analyze**.

   After the analysis is complete, you can view the updated recommendations on the application component details page.

For more information about source code analysis, see Strategy Recommendations source code analysis (p. 21).

Configure database analysis for an application component

This section describes how to use Strategy Recommendations in the Migration Hub console to configure database analysis for an application component.

To configure database analysis for an application component

1. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Recommendations**.
2. On the **Recommendations** page, choose the **Application components** tab.
3. From the list of components under **Application components**, select an application component with component type **SQLServer** and then choose **View details**.
4. On the application component details page (the page with the component's name as the heading), choose the **Database configuration** tab.
5. Under **Database configuration details**, choose **Analyze database details**.
6. Choose a secret name from the dropdown menu that you created in the AWS Secrets Manager to use for database credentials, and then choose **Analyze**.

After the analysis is complete, you can view the updated recommendations on the application component details page.

For more information about database analysis and setting up a secret name, see Strategy Recommendations database analysis (p. 21).

Strategy Recommendations source code analysis

Migration Hub Strategy Recommendations automatically identifies the applications in your portfolio and creates application components for them. For example, if there is a Java application in your portfolio, it's identified as an application component with a component type of **java**.

Strategy Recommendations analyzes the source code for the application components if you configure it to do so. For information about configuring an application component for source code analysis, see Configure source code analysis for an application component (p. 20).

Strategy Recommendations performs source code analysis for the Java and C# programming languages.

For information about the prerequisites for using Strategy Recommendations source code analysis, see Prerequisites for Strategy Recommendations (p. 5).

Strategy Recommendations database analysis

Strategy Recommendations automatically identifies the database servers in your portfolio and creates application components for them. For example, if there is a SQL Server database in your portfolio, it's identified as application component **sqlservr.exe**.

Strategy Recommendations analyzes individual databases in the identified SQL Server application component, sqlservr.exe, using the AWS Schema Conversion Tool. Strategy Recommendations also identifies incompatibilities in migrating the databases to AWS databases such as Amazon Aurora MySQL-Compatible Edition, Amazon Aurora PostgreSQL-Compatible Edition, Amazon RDS for MySQL, and Amazon RDS for PostgreSQL.

Currently, Strategy Recommendations database analysis is only available for SQL Server.

To configure Strategy Recommendations to analyze your databases, you must provide credentials for the Strategy Recommendations application data collector to connect to your databases. To do this, create a secret in AWS Secrets Manager in your AWS account.

For information about the permissions and privileges of the credentials that you provide, see Privileges needed for AWS Schema Conversion Tool credentials (p. 22). For information about creating a secret with the credentials, see Creating a secret in Secrets Manager for database credentials (p. 22).

After you set up the credentials and secret, you can configure AWS Schema Conversion Tool analysis on the database server. For more information, see Configure database analysis for an application component (p. 20).

After you configure database analysis for the application component, a AWS Schema Conversion Tool inventory task is scheduled. After this task completes, you'll see the new application components being created for every individual database on that database server. For example, if your SQL Server has two databases (exampledbs1 and exampledbs2), an application component is created for each of the databases with the names exampledbs1 and exampledbs2.

If you would like to see anti-patterns in migrating each identified database to AWS databases, set up analysis for each database following the steps in Configure database analysis for an application component (p. 20).

Privileges needed for AWS Schema Conversion Tool credentials

The username and password that you provide to AWS Secrets Manager only needs `VIEW SERVER STATE` and `VIEW ANY DEFINITION` privileges. Optionally, you can create a new login by using the script available at https://gitlab.aws.dev/dmaf-pub/dmaf/-/blob/master/create_mssql_ro_user.sql.

You can provide any login name and password that you want when creating the SQL Server login.

Creating a secret in Secrets Manager for database credentials

After the credentials are ready for the Strategy Recommendations application data collector to connect to a database, create a secret in AWS Secrets Manager in your AWS account as described in the following procedure.

To create a secret with AWS Secrets Manager in your AWS account

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the AWS Secrets Manager console at https://console.aws.amazon.com/secretsmanager/.
2. Choose **Store a new secret**.
3. Select the secret type as **Other type of secrets**.
4. Under **Key/value pairs**, enter the following information.

   username - *your-username*

   Then choose **+ Add row** and enter following information.

   password - *your-password*
5. Choose **Next**.
6. Enter **Secret name** as any string with the prefix **migrationhub-strategy-**. For example, **migrationhub-strategy-one**.

   Note
   Store your secret name in a safe place for later use.
7. Choose **Next**, and then choose **Next** again.
8. Choose **Store**.

You can use the secret you created for database credentials when setting up database analysis in Strategy Recommendations.

Strategy Recommendations server recommendations

This section describes how to use Migration Hub Strategy Recommendations in the Migration Hub console to view migration strategy recommendations for the servers in your migration portfolio.

To view recommendations for servers

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Recommendations**.

3. On the **Recommendations** page, choose the **Servers** tab.

    a. Under **Server summary**, you view an overview of the various types of servers that you are running in your portfolio.

    b. Under **Servers**, you view server and operating system details and migration "R" strategy recommendations. You can also view the migration destination and the number of anti-patterns identified on your servers, which are based on the recommendations. For information about the "R" strategy, see Migration terms - 7 Rs in the *AWS Prescriptive Guidance* glossary.
4. To view in-depth recommendation details for a server, select the server from the list, and then choose **View details**. You can view the metadata collected for the server, along with in-depth analysis and recommendations for it, which are based on the application components found running on the server.
5. On the server details page (the page with the server's name as the heading), under **Recommendation summary**, you can see an overview of **Strategy recommendations** for the server. You can also view identified **Anti-patterns**. Anti-patterns are a list of known issues found in your portfolio that are categorized by severity.
6. Choose the **Strategy options** tab to view the migration recommendation for the server. You can override the recommended strategy by selecting a different strategy and then choosing **Set preferred**.
7. Choose the **Application components** tab to view the list of application components associated with the server.
8. To view details about the application component, select the component from the list and then choose **View details.** For more information about application components, see Working with application components (p. 19).

Strategy Recommendations preferences

This section describes how to view and edit Migration Hub Strategy Recommendations preferences in the Migration Hub console.

You choose your recommendation preferences when you first set up Strategy Recommendations as described in Step 5: Get recommendations (p. 17). You can edit these preferences.

To edit recommendation preferences

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Recommendations**.
3. On the **Recommendations** page, choose the **Preferences** tab.
4. Under **Prioritized business goals**, you can drag and drop the business goals to rearrange them.
5. Choose the **Application preferences** and **Database preferences** that you want, and then choose **Save changes**.

If you change your preferences, a banner is displayed to remind you to choose **Reanalyze data**.

Strategy Recommendations data sources

This section describes the data sources that Strategy Recommendations uses.

Topics

Viewing Strategy Recommendations data sources

This section describes how to view Strategy Recommendations data sources in the AWS Management Console.

To view data sources

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Data sources**.
3. On the **Collectors** tab, you can view the Strategy Recommendations application data collectors that you set up. For more information about the collector, see Strategy Recommendations application data collector (p. 24).
4. On the **Imports** tab, you can import data and view your data imports. For more information, see Importing data into Strategy Recommendations (p. 27).
5. On the **Tools** tab, you can download the collector and application import data template.

Strategy Recommendations application data collector

This section describes how to use the Strategy Recommendations application data collector.

For information about downloading and setting up an application data collector, see Step 1: Download the Strategy Recommendations collector (p. 6).

Topics

Data collected by the Strategy Recommendations collector

This section describes the type of data that the Migration Hub Strategy Recommendations application data collector collects. An application data collector is an agentless data collector that identifies running applications on your servers, performs source code analysis, and analyzes your databases.

Data field	Description
OS type	Windows or Linux
OS version	The specific version of the OS. For example, Windows Server 2003, RHEL 5.2.
OS architecture	32-bit or 64-bit OS
Is Server VM	The server is a VM or a physical machine.
Virtualization software	For example, vCenter, Hyper-V.
Location	For example, Amazon Elastic Compute Cloud console (Amazon EC2), or on-premises.
Is dualBoot	Allows booting into multiple OSs
Firmware type	BIOS, UEFI
Boot loader	GRUB, GRUB 2
Partition table type	MBR, GPT
CPU speed	CPU speed in GHz. For example, 2.4 GHz.
Windows OS data	
Windows Edition	Standard, Data Center, Enterprise
.NET framework version	The version of the .NET framework installed.
.NET Core version	The version of .NET Core installed.
Linux data	
Linux OS distribution	RHEL, CentOS, SUSE, and so on.
Kernel version	uname -r output, such as `4.9.217-0.1.ac.205.84.332.metal1.x86_64`
For each disk volume	
File system type	FAT32, NTFS, ReFS, ext4, jfs, and so on.
Disk volume size	Total disk size
Disk volume free space	Free disk space
Virtual disk image format	vmdk, vhd, vhdx
Disk type (Windows)	Basic, Dynamic
Application level data	

Data field	Description
Application name	The name of the running process. For example, SQLServr.exe, MSdtsservr.exe, and so on.
Application type	IIS, JBoss, Tomcat, and so on.
Programming language & version	C#, Java
JDK version	The version of the JDK installed.
Is source code available	If you provide a source code repository, it indicates that source code is available.
Application bit size	16-bit, 32-bit, 64-bit
Windows	
.NET framework version used by app	The version of the .NET framework DLL being loaded at runtime for the application.
.NET Core version	The version .NET Core DLL being loaded at runtime for the application.
Uses WPF framework ?	Determines if the .NET based application is a type of WPF app or not.
Uses WCF framework ?	Determines if the .NET based application is a type of WCF app or not.
ASP.NET version	The version of ASP.NET.
IIS version	The version of IIS server installed on the Windows machine.
Application OS drivers bit size	32-bit, 64-bit
Windows registry usage	Queries the registry keys of the machine to find information like database version, Java version, .NET version, and so on.
All DLLs used by the application	Fetches the list of all the DLLs loaded at runtime by a Windows process.
PowerShell version	Checks the PowerShell version installed on the machine, which should be 5.1 or later.
Linux	
Application framework type	Tomcat, Spring Boot, JBoss, WebLogic, WebSphere
Application framework version	The version of the application framework.
Database	
Database type	MS SQL, Oracle, MySQL, and so on.
Database version	The version of the database.

Remove your data from Strategy Recommendations

To have all your data removed from Strategy Recommendations, contact AWS Support and request full data deletion.

Upgrading the Strategy Recommendations collector

The Migration Hub Strategy Recommendations application data collector upgrades automatically. You can use the following procedure to manually upgrade the collector, if needed.

To upgrade the Strategy Recommendations collector

1. Use the following command to connect to the collector VM using an SSH client.

```
ssh ec2-user@CollectorIPAddress
```

2. Change to the upgrade directory in the collector VM as shown in the following example.

```
cd  /home/ec2-user/collector/upgrades
```

3. Use the following command to run the upgrade script.

```
bash application-data-collector-upgrade
```

Importing data into Strategy Recommendations

As an alternative to using the application data collector, you can import information about the applications and servers for which you want migration and modernization recommendations.

When you import data, the recommendations are not as in-depth as they are when you use the data collector. For example, you cannot use source code analysis on imported data.

This section describes how to use the application import template to import data into Strategy Recommendations in the Migration Hub console.

To import data

1. Using the AWS account that you created in Setting up Strategy Recommendations (p. 3), sign in to the AWS Management Console and open the Migration Hub console at https://console.aws.amazon.com/migrationhub/.
2. In the Migration Hub console navigation pane, choose **Strategy** and then choose **Data sources**.
3. Choose the **Imports** tab.
4. Choose **Download import template** to download the application import template.
5. Fill out the template and then upload it to an Amazon S3 bucket.
6. Return to the **Imports** tab and then choose **Import**.
7. Enter a name for your import, enter the Amazon S3 object URI for your filled out data template and then choose **Start import**.

The Strategy Recommendations import template

The import template that you download is a `.json` file as shown in the following example.

```
{
    "ImportFormatVersion": 1,
    "Resources": [
        {
            "ResourceType": "SERVER",
            "ResourceName": "",
            "ResourceId": "",
            "IpAddress": "",
            "OSDistribution": "",
            "OSType": "",
            "HostName": "",
            "OSVersion": "",
            "CPUArchitecture": ""
        },
        {
            "ResourceType": "PROCESS",
            "ResourceName": "",
            "ResourceId": "",
            "ApplicationType": "",
            "DotNetFrameworkVersion": "",
            "ApplicationVersion": "",
            "DotNetCoreVersion": "",
            "JdkVersion": "",
            "ProgrammingLanguage": "",
            "DatabaseType": "",
            "DatabaseVersion": "",
            "DatabaseEdition": "",
            "AssociatedServerIds": []
        }
    ]
}
```

To help you fill out the import template, the valid values for the data fields are listed in the following table.

Name	Description	Type	Required	Valid values
ApplicationType	The type of application	String	Yes	"Tomcat", "JBoss", "Spring", "IIS", "Mongo DB", "DB2", "Maria DB", "MySQL", "Oracle", "SQLServer", "Sybase", "PostgreSQLServer", "Cassandra", "IBM WebSphere", "Oracle WebLogic", "Java Generic"
ApplicationVersion	The version of the application	String	Yes	"IIS 1.0", "IIS 2.0", "IIS 3.0", "IIS 4.0", "IIS 5.0", "IIS 5.1", "IIS 6.0", "IIS 7.0", "IIS 7.5", "IIS 8.0", "IIS 8.5", "IIS 10.0"
AssociatedServerIds	A list of server IDs on which the process is running.	String	Yes	The ResourceId from the "ResourceType": "SERVER" that you defined.
CPUArchitecture	The CPU architecture	String	No	"32bit", "64bit"
DatabaseEdition	The edition of the database	String	No	

Name	Description	Type	Required	Valid values
DatabaseType	The type database	String	No	"SQLServer", "Oracle", "Sybase", "Mongo DB", "Maria DB", "Apache Cassandra", "MySQL", "IBM DB2", "PostgreSQLServer"
DatabaseVersion	The version of the database	String	No	See the HTML version of the documentation.
DotNetCoreVersion	The version of .NET Core if the application is .NET Core based	String	No	".NET Core 1.0", ".NET Core 1.1", ".NET Core 2.0", ".NET Core 2.1", ".NET Core 2.2", ".NET Core 3.0", ".NET Core 3.1"
DotNetFrameworkVersion	The version of .NET Framework if the application is .NET Framework based	String	No	"DotnetFramework 1.0", "DotnetFramework 1.0 SP1", "DotnetFramework 1.0 SP2", "DotnetFramework 1.0 SP3", "DotnetFramework 1.1", "DotnetFramework 1.1 SP1", "DotnetFramework 2.0", "DotnetFramework 2.0 SP1", "DotnetFramework 2.0 SP2", "DotnetFramework 3.0", "DotnetFramework 3.0 SP1", "DotnetFramework 3.0 SP2", "DotnetFramework 3.5", "DotnetFramework 3.5 SP1", "DotnetFramework 4.0", "DotnetFramework 4.5", "DotnetFramework 4.5.1", "DotnetFramework 4.5.2", "DotnetFramework 4.6", "DotnetFramework 4.6.1", "DotnetFramework 4.6.2", "DotnetFramework 4.7", "DotnetFramework 4.7.1", "DotnetFramework 4.7.2", "DotnetFramework 4.8"
Hostname	The name of the host	String	No	Any string
IpAddress	The IP address of the server	Array	No	In the format xxx.xxx.xxx.xxx
JdkVersion	The version of the JDK, if the application uses the JDK	String	No	"JDK1.0", "JDK2.0", "JDK3.0", …, "JDK11.0"

Name	Description	Type	Required	Valid values
OSDistribution	Windows, Windows Server, Ubuntu	String	Yes	**Windows:** "Windows PC" , "Windows Server" **Linux:** "Ubuntu", "RHEL", "Amazon Linux", "DEBIAN", "SLES", "CENT_OS" , "ORACLE_LINUX", "FEDORA", "KALI"
OSType	The type of operating system	String	Yes	"Windows", "Linux"
OSVersion	The kernel version	String	Yes	See the HTML version of the documentation.
ProgrammingLanguage	The programming language for the application	String	No	"Java", "CSharp"
ResourceId	A unique ID for the resource	String	Yes	Any unique string
ResourceName	The name of the resource	String	Yes	Any string
ResourceType	The type of resource to import	String	Yes	"Server", "Process"

Removing your data from Strategy Recommendations

To have all your data removed from Migration Hub Strategy Recommendations, contact AWS Support.

Security in Migration Hub Strategy Recommendations

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from data centers and network architectures that are 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 Migration Hub Strategy Recommendations, 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 Strategy Recommendations. The following topics show you how to configure Strategy Recommendations to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Strategy Recommendations resources.

Topics

Data protection in Migration Hub Strategy Recommendations

The AWS shared responsibility model applies to data protection in Migration Hub Strategy Recommendations. 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. This content includes 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 user accounts with 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 recommend TLS 1.2 or later.
- Set up API and user activity logging with AWS CloudTrail.

- 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 personal data that is stored in Amazon S3.
- If you require FIPS 140-2 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) 140-2.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form fields such as a **Name** field. This includes when you work with Strategy Recommendations or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form 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.

Encryption at rest

All data stored in Strategy Recommendations' database is encrypted.

Encryption in transit

Strategy Recommendations internetwork communications support TLS 1.2 encryption between all components and clients.

Identity and access management for Migration Hub Strategy Recommendations

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 Strategy Recommendations resources. IAM is an AWS service that you can use with no additional charge.

Topics

Audience

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

Service user – If you use the Strategy Recommendations service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more Strategy Recommendations 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 Strategy Recommendations, see Troubleshooting Migration Hub Strategy Recommendations identity and access (p. 48).

Service administrator – If you're in charge of Strategy Recommendations resources at your company, you probably have full access to Strategy Recommendations. It's your job to determine which Strategy Recommendations features and resources your employees 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 Strategy Recommendations, see How Migration Hub Strategy Recommendations works with IAM (p. 36).

IAM administrator – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to Strategy Recommendations. To view example Strategy Recommendations identity-based policies that you can use in IAM, see Identity-based policy examples for Migration Hub Strategy Recommendations (p. 45).

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials. For more information about signing in using the AWS Management Console, see Signing in to the AWS Management Console as an IAM user or root user in the *IAM User Guide*.

You must be *authenticated* (signed in to AWS) as the AWS account root user, an IAM user, or by assuming an IAM role. You can also use your company's single sign-on authentication or even sign in using Google or Facebook. In these cases, your administrator previously set up identity federation using IAM roles. When you access AWS using credentials from another company, you are assuming a role indirectly.

To sign in directly to the AWS Management Console, use your password with your root user email address or your IAM user name. You can access AWS programmatically using your root user or IAM users access keys. AWS provides SDK and command line tools to cryptographically sign your request using your credentials. If you don't use AWS tools, you must sign the request yourself. Do this using *Signature Version 4*, a protocol for authenticating inbound API requests. For more information about authenticating requests, see Signature Version 4 signing process in the *AWS General Reference*.

Regardless of the authentication method that you use, you might also 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 Using multi-factor authentication (MFA) in AWS in the *IAM User Guide*.

AWS account root user

When you first create an AWS account, you begin with a single 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 do not use the root user for your everyday tasks, even the administrative ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then securely lock away the root user credentials and use them to perform only a few account and service management tasks.

IAM users and groups

An *IAM user* is an identity within your AWS account that has specific permissions for a single person or application. An IAM user can have long-term credentials such as a user name and password or a set of

access keys. To learn how to generate access keys, see Managing access keys for IAM users in the *IAM User Guide*. When you generate access keys for an IAM user, make sure you view and securely save the key pair. You cannot recover the secret access key in the future. Instead, you must generate a new access key pair.

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 When to create an IAM user (instead of a role) 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. You can temporarily assume an IAM role in the AWS Management Console by switching roles. 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 Using IAM roles in the *IAM User Guide*.

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

- **Temporary IAM user permissions** – An IAM user can assume an IAM role to temporarily take on different permissions for a specific task.
- **Federated user access** – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as *federated users*. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated users and roles in the *IAM User Guide*.
- **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 How IAM roles differ from resource-based policies 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.
  - **Principal permissions** – When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, Resources, and Condition Keys for Migration Hub Strategy Recommendations in the *Service Authorization Reference*.
  - **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 Creating 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 IAM 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 Using an IAM role to grant permissions to applications running on Amazon EC2 instances in the *IAM User Guide*.

To learn whether to use IAM roles or IAM users, see When to create an IAM role (instead of a user) in the *IAM User Guide*.

Managing access using policies

You control access in AWS by creating policies and attaching them to IAM identities or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. You can sign in as the root user or an IAM user, or you can assume an IAM role. When you then make a request, AWS evaluates the related identity-based or resource-based policies. 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**.

Every IAM entity (user or role) starts with no permissions. In other words, by default, users can do nothing, not even change their own password. To give a user permission to do something, an administrator must attach a permissions policy to a user. Or the administrator can add the user to a group that has the intended permissions. When an administrator gives permissions to a group, all users in that group are granted those permissions.

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 Creating IAM 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 Choosing 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 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 How SCPs work 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 Migration Hub Strategy Recommendations works with IAM

Before you use IAM to manage access to Strategy Recommendations, learn what IAM features are available to use with Strategy Recommendations.

IAM features you can use with Migration Hub Strategy Recommendations

IAM feature	Strategy Recommendations support
Identity-based policies (p. 37)	Yes

IAM feature	Strategy Recommendations support
Resource-based policies (p. 37)	No
Policy actions (p. 38)	Yes
Policy resources (p. 38)	No
Policy condition keys (p. 39)	No
ACLs (p. 39)	No
ABAC (tags in policies) (p. 40)	No
Temporary credentials (p. 40)	Yes
Principal permissions (p. 40)	Yes
Service roles (p. 41)	No
Service-linked roles (p. 41)	Yes

To get a high-level view of how Strategy Recommendations and other AWS services work with most IAM features, see AWS services that work with IAM in the *IAM User Guide*.

Identity-based policies for Strategy Recommendations

Supports identity-based policies	Yes

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 Creating IAM policies in the *IAM User Guide*.

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. You can't specify the principal in an identity-based policy because it applies to the user or role to which it is attached. To learn about all of the elements that you can use in a JSON policy, see IAM JSON policy elements reference in the *IAM User Guide*.

Identity-based policy examples for Strategy Recommendations

To view examples of Strategy Recommendations identity-based policies, see Identity-based policy examples for Migration Hub Strategy Recommendations (p. 45).

Resource-based policies within Strategy Recommendations

Supports resource-based policies	No

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.

To enable cross-account access, you can specify an entire account or IAM entities in another account as the principal in a resource-based policy. Adding a cross-account principal to a resource-based policy is only half of establishing the trust relationship. When the principal and the resource are in different AWS accounts, an IAM administrator in the trusted account must also grant the principal entity (user or role) permission to access the resource. They grant permission by attaching an identity-based policy to the entity. However, if a resource-based policy grants access to a principal in the same account, no additional identity-based policy is required. For more information, see How IAM roles differ from resource-based policies in the *IAM User Guide*.

Policy actions for Strategy Recommendations

Supports policy actions	Yes

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.

To see a list of Strategy Recommendations actions, see Actions Defined by Migration Hub Strategy Recommendations in the *Service Authorization Reference*.

Policy actions in Strategy Recommendations use the following prefix before the action:

```
migrationhub-strategy
```

To specify multiple actions in a single statement, separate them with commas.

```
"Action": [
      "migrationhub-strategy:action1",
      "migrationhub-strategy:action2"
         ]
```

To view examples of Strategy Recommendations identity-based policies, see Identity-based policy examples for Migration Hub Strategy Recommendations (p. 45).

Policy resources for Strategy Recommendations

Supports policy resources	No

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": "*"
```

To see a list of Strategy Recommendations resource types and their ARNs, see Resources Defined by Migration Hub Strategy Recommendations in the *Service Authorization Reference*. To learn with which actions you can specify the ARN of each resource, see Actions Defined by Migration Hub Strategy Recommendations .

To view examples of Strategy Recommendations identity-based policies, see Identity-based policy examples for Migration Hub Strategy Recommendations (p. 45).

Policy condition keys for Strategy Recommendations

Supports service-specific policy condition keys	No

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 `Condition` element (or `Condition` *block*) lets you specify conditions in which a statement is in effect. The `Condition` element is optional. You can create conditional expressions that use condition operators, such as equals or less than, to match the condition in the policy with values in the request.

If you specify multiple `Condition` elements in a statement, or multiple keys in a single `Condition` element, AWS evaluates them using a logical `AND` operation. If you specify multiple values for a single condition key, AWS evaluates the condition using a logical `OR` operation. All of the conditions must be met before the statement's permissions are granted.

You can also use placeholder variables when you specify conditions. For example, you can grant an IAM user permission to access a resource only if it is tagged with their IAM user name. For more information, see IAM policy elements: variables and tags in the *IAM User Guide*.

AWS supports global condition keys and service-specific condition keys. To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

To see a list of Strategy Recommendations condition keys, see Condition Keys for Migration Hub Strategy Recommendations in the *Service Authorization Reference*. To learn with which actions and resources you can use a condition key, see Actions Defined by Migration Hub Strategy Recommendations .

To view examples of Strategy Recommendations identity-based policies, see Identity-based policy examples for Migration Hub Strategy Recommendations (p. 45).

Access control lists (ACLs) in Strategy Recommendations

Supports ACLs	No

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.

Attribute-based access control (ABAC) with Strategy Recommendations

Supports ABAC (tags in policies)	No

Attribute-based access control (ABAC) is an authorization strategy that defines permissions based on attributes. In AWS, these attributes are called *tags*. You can attach tags to IAM entities (users or roles) and to many AWS resources. Tagging entities and resources is the first step of ABAC. Then you design ABAC policies to allow operations when the principal's tag matches the tag on the resource that they are trying to access.

ABAC is helpful in environments that are growing rapidly and helps with situations where policy management becomes cumbersome.

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 ABAC, see What is ABAC? in the *IAM User Guide*. To view a tutorial with steps for setting up ABAC, see Use attribute-based access control (ABAC) in the *IAM User Guide*.

Using Temporary credentials with Strategy Recommendations

Supports temporary credentials	Yes

Some AWS services don't work when you sign in using temporary credentials. For additional information, including which AWS services work with temporary credentials, see AWS services that work with IAM in the *IAM User Guide*.

You are using temporary credentials if you sign in to the AWS Management Console using any method except a user name and password. For example, when you access AWS using your company's single sign-on (SSO) link, that process automatically creates temporary credentials. You also automatically create temporary credentials when you sign in to the console as a user and then switch roles. For more information about switching roles, see Switching to a role (console) in the *IAM User Guide*.

You can manually create temporary credentials using the AWS CLI or AWS API. You can then use those temporary credentials to access AWS. AWS recommends that you dynamically generate temporary credentials instead of using long-term access keys. For more information, see Temporary security credentials in IAM.

Cross-service principal permissions for Strategy Recommendations

Supports principal permissions	Yes

When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that

then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, Resources, and Condition Keys for Migration Hub Strategy Recommendations in the *Service Authorization Reference*.

Service roles for Strategy Recommendations

Supports service roles	No

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 Creating a role to delegate permissions to an AWS service in the *IAM User Guide*.

> **Warning**
> Changing the permissions for a service role might break Strategy Recommendations functionality. Edit service roles only when Strategy Recommendations provides guidance to do so.

Service-linked roles for Strategy Recommendations

Supports service-linked roles	Yes

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 IAM account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

For details about creating or managing Strategy Recommendations service-linked roles, see Using service-linked roles for Strategy Recommendations (p. 50).

AWS managed policies for Migration Hub Strategy Recommendations

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: AWSMigrationHubStrategyConsoleFullAccess

You can attach the `AWSMigrationHubStrategyConsoleFullAccess` policy to your IAM identities.

The `AWSMigrationHubStrategyConsoleFullAccess` policy grants an IAM user account full access to the Strategy Recommendations service through the AWS Management Console.

Permissions details

This policy includes the following permissions.

- `migrationhub-strategy` – Allows the IAM user account full access to Strategy Recommendations.
- `s3` – Allows the IAM user account to create and read from the S3 buckets used by Strategy Recommendations.
- `secretsmanager` – Allows the IAM user account to list secrets access in the Secrets Manager.
- `discovery` – Allows the IAM user account access to get discovery summary in Application Discovery Service.
- `iam` – Allows a service-linked role to be created for the IAM user account, which is a requirement for using Strategy Recommendations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "migrationhub-strategy:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "arn:aws:s3:::*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:CreateBucket",
                "s3:PutEncryptionConfiguration",
                "s3:PutBucketPublicAccessBlock",
                "s3:PutBucketPolicy",
                "s3:PutBucketVersioning",
```

```
                "s3:PutLifecycleConfiguration"
            ],
            "Resource": "arn:aws:s3:::migrationhub-strategy-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:ListSecrets"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "discovery:GetDiscoverySummary"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": "migrationhub-strategy.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:GetRole"
            ],
            "Resource": "arn:aws:iam::*:role/aws-service-role/migrationhub-
strategy.amazonaws.com/AWSMigrationHubStrategyServiceRolePolicy*"
        }
    ]
}
```

AWS managed policy: AWSMigrationHubStrategyCollector

You can attach the `AWSMigrationHubStrategyCollector` policy to your IAM identities.

The `AWSMigrationHubStrategyCollector` policy grants an IAM user account access to the Strategy Recommendations service, read/write access to the S3 buckets that are related to the service, Amazon API Gateway access to upload logs and metrics to AWS, and AWS Secrets Manager access to fetch credentials.

Permissions details

This policy includes the following permissions.

- `s3` – Allows the IAM user account write access to the S3 buckets used by Strategy Recommendations.
- `migrationhub-strategy` – Allows the IAM user account access to register and send messages to Strategy Recommendations.
- `execute-api` – Allows the IAM user account to access Amazon API Gateway to upload logs and metrics to AWS.

- `secretsmanager` – Allows the IAM user account to access secrets in the Secrets Manager that are used by Strategy Recommendations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetBucketAcl"
            ],
            "Resource": "arn:aws:s3:::migrationhub-strategy-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "arn:aws:s3:::*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "execute-api:Invoke",
                "execute-api:ManageConnections"
            ],
            "Resource": [
                "arn:aws:execute-api:*:*:*/prod/*/put-log-data",
                "arn:aws:execute-api:*:*:*/prod/*/put-metric-data"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "migrationhub-strategy:RegisterCollector",
                "migrationhub-strategy:GetAntiPattern",
                "migrationhub-strategy:GetMessage",
                "migrationhub-strategy:SendMessage",
                "migrationhub-strategy:ListAntiPatterns",
                "migrationhub-strategy:ListJarArtifacts"
            ],
            "Resource": "arn:aws:migrationhub-strategy:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:migrationhub-strategy-*"
        }
    ]
}
```

Strategy Recommendations updates to AWS managed policies

View details about updates to AWS managed policies for Strategy Recommendations since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the Strategy Recommendations Document history page.

Change	Description	Date
AWSMigrationHubStrategyConsoleFullAccess (p. 42) – New policy made available at launch	`AWSMigrationHubStrategyConsoleFullAccess` grants an IAM user account full access to the Strategy Recommendations service through the AWS Management Console.	October 25, 2021
AWSMigrationHubStrategyCollector (p. 43) – New policy made available at launch	`AWSMigrationHubStrategyCollector` grants an IAM user account access to the Strategy Recommendations service and read/write access to the S3 buckets that are related to the service. It also grants Amazon API Gateway access to upload logs and metrics to AWS, and AWS Secrets Manager access to fetch credentials.	October 25, 2021
AWSMigrationHubStrategyServiceRolePolicy (p. 50) – New policy made available at launch	The `AWSMigrationHubStrategyServiceRolePolicy` service-linked role policy provides access to AWS Migration Hub and AWS Application Discovery Service. This policy also grants permissions for storing reports in Amazon Simple Storage Service (Amazon S3).	October 25, 2021
Strategy Recommendations started tracking changes	Strategy Recommendations started tracking changes for its AWS managed policies.	October 25, 2021

Identity-based policy examples for Migration Hub Strategy Recommendations

By default, IAM users and roles don't have permission to create or modify Strategy Recommendations 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 actions on the resources that 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 IAM policies in the *IAM User Guide*.

Topics

Policy best practices

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Strategy Recommendations 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 using AWS managed policies** – To start using Strategy Recommendations quickly, use AWS managed policies to give your employees the permissions they need. These policies are already available in your account and are maintained and updated by AWS. For more information, see Get started using permissions with AWS managed policies in the *IAM User Guide*.
- **Grant least privilege** – When you create custom policies, grant only the permissions required to perform a task. Start with a minimum set of permissions and grant additional permissions as necessary. Doing so is more secure than starting with permissions that are too lenient and then trying to tighten them later. For more information, see Grant least privilege in the *IAM User Guide*.
- **Enable MFA for sensitive operations** – For extra security, require IAM users to use multi-factor authentication (MFA) to access sensitive resources or API operations. For more information, see Using multi-factor authentication (MFA) in AWS in the *IAM User Guide*.
- **Use policy conditions for extra security** – To the extent that it's practical, define the conditions under which your identity-based policies allow access to a resource. For example, you can write conditions to specify a range of allowable IP addresses that a request must come from. You can also write conditions to allow requests only within a specified date or time range, or to require the use of SSL or MFA. For more information, see IAM JSON policy elements: Condition in the *IAM User Guide*.

Using the Strategy Recommendations console

To access the Migration Hub Strategy Recommendations console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Strategy Recommendations 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.

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.

To ensure that users and roles can still use the Strategy Recommendations console, also attach the Strategy Recommendations `ConsoleAccess` or `ReadOnly` AWS managed policy to the entities. For more information, see Adding permissions to a user in the *IAM User Guide*.

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": "*"
        }
    ]
}
```

Accessing one Amazon S3 bucket

In this example, you want to grant an IAM user in your AWS account access to one of your Amazon S3 buckets, `examplebucket`. You also want to allow the user to add, update, and delete objects.

In addition to granting the `s3:PutObject`, `s3:GetObject`, and `s3:DeleteObject` permissions to the user, the policy also grants the `s3:ListAllMyBuckets`, `s3:GetBucketLocation`, and `s3:ListBucket` permissions. These are the additional permissions required by the console. Also, the `s3:PutObjectAcl` and the `s3:GetObjectAcl` actions are required to be able to copy, cut, and paste objects in the console. For an example walkthrough that grants permissions to users and tests them using the console, see An example walkthrough: Using user policies to control access to your bucket.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"ListBucketsInConsole",
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets"
         ],
         "Resource":"arn:aws:s3:::*"
      },
      {
         "Sid":"ViewSpecificBucketInfo",
         "Effect":"Allow",
         "Action":[
            "s3:ListBucket",
            "s3:GetBucketLocation"
         ],
         "Resource":"arn:aws:s3:::examplebucket"
      },
      {
         "Sid":"ManageBucketContents",
         "Effect":"Allow",
         "Action":[
            "s3:PutObject",
            "s3:PutObjectAcl",
            "s3:GetObject",
            "s3:GetObjectAcl",
            "s3:DeleteObject"
```

```
        ],
        "Resource":"arn:aws:s3:::examplebucket/*"
      }
   ]
}
```

Troubleshooting Migration Hub Strategy Recommendations identity and access

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

Topics

I am not authorized to perform an action in Strategy Recommendations

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 fictional *`my-example-widget`* resource but does not have the fictional `migrationhub-strategy:`*`GetWidget`* permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 migrationhub-strategy:GetWidget on resource: my-example-widget
```

In this case, Mateo asks his administrator to update his policies to allow him to access the *`my-example-widget`* resource using the `migrationhub-strategy:`*`GetWidget`* 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, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password. Ask that person to update your policies to allow you to pass a role to Strategy Recommendations.

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 Strategy Recommendations. However, the action requires the service to have permissions 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 asks her administrator to update her policies to allow her to perform the `iam:PassRole` action.

I want to view my access keys

After you create your IAM user access keys, you can view your access key ID at any time. However, you can't view your secret access key again. If you lose your secret key, you must create a new access key pair.

Access keys consist of two parts: an access key ID (for example, `AKIAIOSFODNN7EXAMPLE`) and a secret access key (for example, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Like a user name and password, you must use both the access key ID and secret access key together to authenticate your requests. Manage your access keys as securely as you do your user name and password.

> **Important**
> Do not provide your access keys to a third party, even to help find your canonical user ID. By doing this, you might give someone permanent access to your account.

When you create an access key pair, you are prompted to save the access key ID and secret access key in a secure location. The secret access key is available only at the time you create it. If you lose your secret access key, you must add new access keys to your IAM user. You can have a maximum of two access keys. If you already have two, you must delete one key pair before creating a new one. To view instructions, see Managing access keys in the *IAM User Guide*.

I'm an administrator and want to allow others to access Strategy Recommendations

To allow others to access Strategy Recommendations, you must create an IAM entity (user or role) for the person or application that needs access. They will use the credentials for that entity to access AWS. You must then attach a policy to the entity that grants them the correct permissions in Strategy Recommendations.

To get started right away, see Creating your first IAM delegated user and group in the *IAM User Guide*.

I want to allow people outside of my AWS account to access my Strategy Recommendations 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 Strategy Recommendations supports these features, see How Migration Hub Strategy Recommendations works with IAM (p. 36).
- 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 How IAM roles differ from resource-based policies in the *IAM User Guide*.

Using service-linked roles for Strategy Recommendations

Migration Hub Strategy Recommendations 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 Strategy Recommendations. Service-linked roles are predefined by Strategy Recommendations and include all the permissions that the service requires to call other AWS services on your behalf.

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

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 Strategy Recommendations

Strategy Recommendations uses the service-linked role named **AWSServiceRoleForMigrationHubStrategy** and associates it with **AWSMigrationHubStrategyServiceRolePolicy** IAM policy – Provides access to AWS Migration Hub and AWS Application Discovery Service. This policy also grants permissions for storing reports in Amazon Simple Storage Service (Amazon S3).

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

- `migrationhub-strategy.amazonaws.com`

The role permissions policy allows Strategy Recommendations to complete the following actions.

AWS Application Discovery Service actions

```
discovery:ListConfigurations

discovery:DescribeConfigurations
```

AWS Migration Hub actions

```
mgh:GetHomeRegion
```

Amazon S3 actions

```
s3:GetBucketAcl

s3:GetBucketLocation

s3:GetObject

s3:ListAllMyBuckets

s3:ListBucket

s3:PutObject

s3:PutObjectAcl
```

The following is the full policy showing which resources the above actions apply to:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "permissionsForAds",
            "Effect": "Allow",
            "Action": [
                "discovery:ListConfigurations",
                "discovery:DescribeConfigurations",
                "mgh:GetHomeRegion"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "arn:aws:s3:::*"
        },
        {
            "Sid": "permissionsForS3",
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketAcl",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": "arn:aws:s3:::migrationhub-strategy-*"
        }
    ]
}
```

To view the update history of this policy, see Strategy Recommendations updates to AWS managed policies (p. 44).

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 Strategy Recommendations

You don't need to manually create a service-linked role. When you agree to allow Migration Hub to create a service-linked role (SLR) in your account in the AWS Management Console, Strategy Recommendations 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 agree to allow Migration Hub to create a service-linked role (SLR) in your account, Strategy Recommendations creates the service-linked role for you again.

Editing a service-linked role for Strategy Recommendations

Strategy Recommendations does not allow you to edit the **AWSServiceRoleForMigrationHubStrategy** service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using the Strategy Recommendations console, CLI, or API.

Deleting a service-linked role for Strategy Recommendations

To manually delete the service-linked role using IAM

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

When deleting Strategy Recommendations resources used by the **AWSServiceRoleForMigrationHubStrategy** SLR, you cannot have any running assessments (tasks for generating recommendations). No background assessments can be running, either. If assessments are running, the SLR deletion fails in the IAM console. If the SLR deletion fails, you can retry the deletion after all background tasks have completed. You don't need to clean up any created resources before you delete the SLR.

Supported Regions for Strategy Recommendations service-linked roles

Strategy Recommendations supports using service-linked roles in all of the regions where the service is available. For more information, see AWS Regions and Endpoints.

Migration Hub Strategy Recommendations and interface VPC endpoints (AWS PrivateLink)

You can establish a private connection between your VPC and Migration Hub Strategy Recommendations by creating an *interface VPC endpoint*. Interface endpoints are powered by AWS PrivateLink. With AWS PrivateLink, you can privately access Strategy Recommendations API operations without an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC don't need public IP addresses to communicate with Strategy Recommendations API operations. Traffic between your VPC and Strategy Recommendations stays within the Amazon network.

Each interface endpoint is represented by one or more Elastic Network Interfaces in your subnets.

For more information, see Interface VPC endpoints (AWS PrivateLink) in the *Amazon VPC User Guide*.

Considerations for Strategy Recommendations VPC endpoints

Before you set up an interface VPC endpoint for Strategy Recommendations, ensure that you review Interface endpoint properties and limitations and AWS PrivateLink quotas in the *Amazon VPC User Guide*.

Strategy Recommendations supports making calls to all of its API actions from your VPC. To use all of Strategy Recommendations, you must create a VPC endpoint.

Creating an interface VPC endpoint for Strategy Recommendations

You can create a VPC endpoint for Strategy Recommendations using either the Amazon VPC console or the AWS Command Line Interface (AWS CLI). For more information, see Creating an interface endpoint in the *Amazon VPC User Guide*.

Create a VPC endpoint for Strategy Recommendations using the following service name:

- `com.amazonaws.region.migrationhub-strategy`

If you use private DNS for the endpoint, you can make API requests to Strategy Recommendations using its default DNS name for the Region. For example, you can use the name `migrationhub-strategy.us-east-1.amazonaws.com`.

For more information, see Accessing a service through an interface endpoint in the *Amazon VPC User Guide*.

Creating a VPC endpoint policy for Strategy Recommendations

You can attach an endpoint policy to your VPC endpoint that controls access to Strategy Recommendations. The policy specifies the following information:

- The principal that can perform actions.
- The actions that can be performed.
- The resources on which these actions can be performed.

For more information, see Controlling access to services with VPC endpoints in the *Amazon VPC User Guide*.

Example: VPC endpoint policy for Strategy Recommendations actions

The following is an example of an endpoint policy for Strategy Recommendations. When attached to an endpoint, this policy grants access to the listed Strategy Recommendations actions for all principals on all resources.

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
            "migrationhub-strategy:ListContacts",
         ],
         "Resource":"*"
      }
   ]
}
```

Compliance validation for Migration Hub Strategy Recommendations

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 Strategy Recommendations 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.

- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- Evaluating Resources with Rules in the *AWS Config Developer Guide* – AWS Config; 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 standards and best practices.

Working with other services

This section describes other AWS services that interact with Migration Hub Strategy Recommendations.

Topics

Logging Strategy Recommendations API calls with AWS CloudTrail

Migration Hub Strategy Recommendations is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Strategy Recommendations. CloudTrail captures API calls for Strategy Recommendations as events. The calls captured include calls from the Strategy Recommendations console and code calls to the Strategy Recommendations API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Strategy Recommendations. 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 Strategy Recommendations, 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.

Strategy Recommendations information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in Strategy Recommendations, 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 Strategy Recommendations, 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 the following:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple regions and Receiving CloudTrail log files from multiple accounts

Strategy Recommendations supports logging the following actions as events in CloudTrail log files:

- GetApplicationComponentStrategies
- GetApplicationComponentDetails
- GetAssesment

- GetImportFileTask
- GetPortfolioPreferences
- GetPortfolioSummary
- GetServerDetails
- GetServerStrategies
- ListApplicationComponents
- ListCollectors
- ListImportFileTask
- ListServers
- PutPortfolioPreferences
- StartAssessment
- StartImportFileTask
- StopAssessment
- UpdateApplicationComponetConfig
- UpdateServerConfig

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or AWS Identity and Access Management (IAM) 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 Strategy Recommendations 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 GetServerDetails action.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "777777777777",
        "arn": "arn:aws:sts::111122223333:assumed-role/myUserName/...",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "777777777777",
                "arn": "arn:aws:iam::111122223333:role/myUserName",
                "accountId": "111122223333",
                "userName": "myUserName"
```

```
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-09-20T01:07:16Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2021-09-20T01:07:43Z",
    "eventSource": "migrationhub-strategy.amazonaws.com",
    "eventName": "GetServerDetails",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "",
    "userAgent": "",
    "requestParameters": {
        "serverId": "ads-server-006"
    },
    "responseElements": null,
    "requestID": "07D681279BD94AED",
    "eventID": "cdc4b7ed-e171-4cef-975a-ad829d4123e8",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Quotas for Migration Hub Strategy Recommendations

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is Region-specific. You can request increases for some quotas, and other quotas cannot be increased.

To view a list of the quotas for Migration Hub Strategy Recommendations, see Strategy Recommendations service quotas.

You can also view the quotas for Strategy Recommendations, by opening the Service Quotas console. In the navigation pane, choose **AWS services** and select **Migration Hub Strategy Recommendations**.

To request a quota increase, see Requesting a Quota Increase in the *Service Quotas User Guide*. If the quota is not yet available in Service Quotas, use the limit increase form.

Release notes

Topics

April 18, 2022

New feature

- Collector v1.1.7
- Added the capability to dynamically download A2C binary from the public URL.

Bug fixes

- A2C v1.1.5

February 25, 2022

Bug fixes

- SCT v5.6.9
- A2C v1.1.2
- Collector v1.1.4

February 10, 2022

Bug fixes

- SCT v5.6.8
- A2C v1.1.1
  - Added a check for the **tar** command on Linux.
  - Fixed the issue of checking application images in Amazon ECR.
  - Fixed the issue requiring container removal for pre-validation.
- Collector v1.1.3
  - Fixed the 4xx error for remote 32-bit machine.
  - Updated the A2C error codes.

- Validated the IP address in C# for source code analysis of the remote machine.

January 28, 2022

New feature

- Collector v1.1.2
- Added Azure DevOps Git repository support for source code analysis.

January 14, 2022

New feature

- Collector v1.1.1
- Added Babelfish recommendations for SQL databases.

December 21, 2021

Issue resolved

- Collector v1.1.0
- Database analysis has been restored.

December 15, 2021

Known issue

- Collector v1.0.4
- Database analysis is currently unsupported (CVE-2021-44228).

October 25, 2021

New feature

- Collector v1.0.0
- Initial release of the Migration Hub Strategy Recommendations User Guide.

Document and version history

The following table describes the documentation releases for Strategy Recommendations.

Change	Description
Security updates	Establish a private connection with interface VPC endpoint.
New feature	Added Azure DevOps Git repository support for source code analysis.
New feature	Added Babelfish recommendations for SQL databases.
Initial release	Initial release of the Migration Hub Strategy Recommendations User Guide.