AWS DataSync: User Guide

Copyright © 2020 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

What Is AWS DataSync? ................................................................. 1
Use Cases ...................................................................................... 1
Benefits ........................................................................................ 1
Additional AWS DataSync Resources ........................................... 2
How AWS DataSync Works ............................................................ 3
AWS DataSync Architecture ......................................................... 3
  Transfer Data from On-Premises to AWS ................................. 3
  Transfer Data from In-Cloud NFS to In-Cloud NFS or S3 ............ 4
  Transfer from S3 to In-Cloud NFS ............................................. 5
Components and Terminology ...................................................... 5
  Agent ....................................................................................... 6
  Location .................................................................................. 6
  Task ......................................................................................... 6
  Task Execution ......................................................................... 6
How DataSync Transfers Files ..................................................... 7
  How AWS DataSync Verifies Data Integrity ............................. 7
  How DataSync Handles Open and Locked Files ......................... 7
Setting Up ....................................................................................... 8
  Sign Up For AWS .................................................................... 8
  AWS Regions and Endpoints ................................................ 8
  How to Access AWS DataSync ................................................ 8
  DataSync Pricing ..................................................................... 8
Requirements ............................................................................... 9
  Agent Requirements ............................................................... 9
    Supported Hypervisors ....................................................... 9
    Virtual Machine Requirements ............................................ 9
    Amazon EC2 Instance Requirements .................................... 10
  Network Requirements for DataSync ........................................ 10
    Network Requirements to Connect to Your NFS or SMB Storage 10
    Network Requirements When Using VPC Endpoints ................. 11
    Network Requirements When Using Public Service Endpoints or FIPS Endpoints 14
Getting Started ............................................................................ 20
  Create an Agent ...................................................................... 20
    Deploy an Agent .................................................................. 20
    Choose a Service Endpoint ............................................... 25
    Activate Your Agent .......................................................... 27
  Configure a Source Location ................................................ 27
  Configure a Destination Location .......................................... 29
  Configure Task Settings ...................................................... 30
  Review and Create Your Task ............................................... 31
  Start Your Task ..................................................................... 32
  Clean Up Resources ............................................................ 32
Using the AWS CLI ........................................................................ 33
  Step 1: Create an Agent ....................................................... 33
  Step 2: Create Locations ...................................................... 36
    Create an NFS Location ...................................................... 36
    Create an SMB Location .................................................... 37
    Create an Amazon EFS Location ......................................... 37
    Create an Amazon FSx for Windows File Server Location ....... 39
    Create an Amazon S3 Location ............................................ 40
  Step 3: Create a Task ........................................................... 42
  Step 4: Start a Task Execution ............................................... 43
  Step 5: Monitor Your Task Execution ..................................... 43
    Monitor Your Task Execution in Real Time ........................... 44
What Is AWS DataSync?

AWS DataSync is an online data transfer service designed to simplify, automate, and accelerate copying large amounts of data to and from AWS storage services. DataSync copies data over the internet or AWS Direct Connect. As a fully managed service, DataSync removes much of the need to modify applications, develop scripts, or manage infrastructure.

DataSync supports data transfer between Network File System (NFS) and Amazon Elastic File System (Amazon EFS) or Amazon Simple Storage Service (Amazon S3). It also supports data transfer between Server Message Block (SMB) and Amazon EFS, Amazon S3, or Amazon FSx for Windows File Server.

In this guide, you can find a description of the components of DataSync, detailed instructions on how to get started, and the API reference.

Topics
- Use Cases (p. 1)
- Benefits (p. 1)
- Additional AWS DataSync Resources (p. 2)

Use Cases

These are some of the main use cases for AWS DataSync:

- **Data migration** – Move active datasets rapidly over the network into Amazon S3, Amazon EFS, or Amazon FSx for Windows File Server. DataSync includes automatic encryption and data integrity validation to help make sure that your data arrives securely, intact, and ready to use.
- **Data movement for timely in-cloud processing** – Move data into or out of AWS for processing when working with systems that generate data on-premises. This approach can speed up critical hybrid cloud workflows across many industries. These include video production in media and entertainment, seismic research in oil and gas, machine learning in life science, and big data analytics in finance.
- **Data archiving** – Move cold data from expensive on-premises storage systems directly to durable and secure long-term storage such as Amazon S3 Glacier or S3 Glacier Deep Archive. By doing this, you can free up on-premises storage capacity and shut down legacy storage systems.
- **Data protection** – Move data into all Amazon S3 storage classes, and choose the most cost-effective storage class for your needs. You can also send the data to Amazon EFS or Amazon FSx for Windows File Server for a standby file system.

Benefits

By using AWS DataSync, you can get the following benefits:

- Simplify and automate data movement. Using DataSync, you can easily transfer data between on-premises sources and AWS storage over the network. AWS DataSync automates management of the infrastructure and the transfer processes for you. DataSync also includes encryption and data validation. This approach minimizes the time for in-house development and management that is otherwise needed for fast, reliable, and secure transfers.
- Transfer data fast over the network into AWS, at a rate up to 10 Gbps. This approach speeds up migrations, hybrid workflows for analytics and machine learning, and data protection processes.
• Reduce data transfer costs and move data cost-effectively with the flat, per-gigabyte pricing in DataSync. You also can save on script development and management costs, and avoid the need for costly commercial transfer tools.

Additional AWS DataSync Resources

We recommend that you read the following:

• Transferring files from on-premises to AWS and back without leaving your VPC using AWS DataSync – This technical blog post discusses file transfer within your VPC using AWS DataSync.
• AWS DataSync technical blog posts – Other technical blog posts about AWS DataSync.
• AWS DataSync developer forum – The AWS DataSync developer forum.
• AWS DataSync pricing – AWS DataSync pricing information.

AWS DataSync also supports Terraform. To learn more about DataSync deployment automation with Terraform, see the Terraform documentation.
How AWS DataSync Works

In this section, you can find information about components, terms, and how DataSync works.

Topics
- AWS DataSync Architecture (p. 3)
- Components and Terminology (p. 5)
- How DataSync Transfers Files (p. 7)

AWS DataSync Architecture

Topics
- Transfer Data from On-Premises to AWS (p. 3)
- Transfer Data from In-Cloud NFS to In-Cloud NFS or S3 (p. 4)
- Transfer from S3 to In-Cloud NFS (p. 5)

The architectural diagrams show how DataSync transfers data between on-premises storage systems and AWS storage services, and between in-cloud storage systems and AWS storage services.

For a list of all DataSync supported source and destination endpoints, see Working with Locations (p. 64).

Transfer Data from On-Premises to AWS

In the following diagram, you can see a high-level view of the DataSync architecture for transferring files between on-premises storage and AWS services.
Transfer Between SMB File Shares and Amazon S3, Amazon EFS or Amazon FSx for Windows File Server

You can transfer data between SMB file shares and Amazon S3, Amazon EFS or Amazon FSx for Windows File Server. You can copy data between on-premises SMB file shares without writing scripts, modifying your applications, or managing infrastructure. You can manage authentication and authorization of DataSync for reading and writing to your share in Microsoft Active Directory.

Transfer Data from In-Cloud NFS to In-Cloud NFS or S3

DataSync can transfer data from an in-cloud NFS file system to AWS. To perform this transfer, the DataSync agent must be located in the same AWS Region and same AWS account where the file system is deployed. This type of transfer includes transfers from EFS to EFS, transfers from self-managed NFS to Amazon EFS, and transfers to S3.

In the following diagram, you can see a high-level view of the DataSync architecture for transferring data from in-cloud NFS to in-cloud NFS or S3.
Deploy the agent in the AWS Region and AWS account where the source EFS or self-managed NFS file system resides.

For detailed instructions on how to get started, see Getting Started with AWS DataSync (p. 20).

Transfer from S3 to In-Cloud NFS

DataSync can transfer data from S3 to an in-cloud NFS file system that is located in the same AWS account and AWS Region where the agent is deployed. This approach includes transfers from S3 to EFS, or from S3 to self-managed NFS.

In the following diagram, you can see a high-level view of the DataSync architecture for transferring data from S3 to an in-cloud NFS file system.

Components and Terminology

The components of DataSync include the following:

- **Agent** – A virtual machine used to read data from or write data to an on-premises location.
- **Location** – Any source or destination location used in the data transfer (for example, Amazon S3, Amazon EFS, or Amazon FSx for Windows File Server).
- **Task** – A task includes two locations (source and destination), and also the configuration of how to transfer the data from one location to the other. Configuration settings can include options such as
how to treat metadata, deleted files, and copy permission. A task is the complete definition of a data transfer.

- Task execution – An individual run of a task, which includes options such as start time, end time, bytes written, and status.

Agent

An agent is a virtual machine (VM) that is owned by the user, and is used to read or write data from on-premises storage system. The agent is currently deployed on a VMware ESXi hypervisor. You use the AWS DataSync Management Console or the API to set up and activate your agent. The activation process associates your agent VM with your AWS account. For information about agents, see Working with Agents (p. 78).

An agent that is functioning properly has the status ONLINE. If an agent is unable to communicate with AWS, it transitions to OFFLINE status. This transition can result from issues with a network partition, firewall misconfiguration, and other events that make the agent VM unable to connect to AWS. The status of an agent that is powered off also shows as OFFLINE.

Location

A location is an endpoint of a task. Each task has two locations—a source location and destination location. AWS DataSync supports Network File System (NFS), Server Message Block (SMB), Amazon EFS, Amazon FSx for Windows File Server, and Amazon S3 as location types. For more information, see Working with Locations (p. 64).

Task

A task includes two locations (source and destination), and the configuration of how to transfer the data from one location to the other. Configuration settings can include options such as how to treat metadata, deleted files, and permission. A task is the complete definition of a data transfer.

Task Execution

A task execution is an individual run of a task, which shows information such as start time, end time, number of transferred files, and status.

A task execution has five transition phases and two terminal statuses, as shown in the following diagram.

Task Execution Transition Phases

If the VerifyMode option is not enabled, a terminal status occurs after the TRANSFERRING phase. Otherwise, it occurs after the VERIFYING phase. The two terminal statuses are these:

- SUCCESS
- ERROR
How DataSync Transfers Files

Topics

- How AWS DataSync Verifies Data Integrity (p. 7)
- How DataSync Handles Open and Locked Files (p. 7)

When a task starts, it goes through different statuses: LAUNCHING, PREPARING, TRANSFERRING and VERIFYING. In the LAUNCHING status, DataSync initializes the task execution. In the PREPARING status, DataSync examines the source and destination file systems to determine which files to sync. It does so by recursively scanning the contents of the source and destination file systems for differences. The time that DataSync spends in the PREPARING status depends on the number of files in both the source and destination file systems. It also depends on the performance of these file systems and usually takes between a few minutes to a few hours. For more information, see Starting a Task (p. 57).

After the scanning is done, and the differences are calculated, DataSync transitions to the TRANSFERRING status. At this point, DataSync starts transferring files from the source file system to the destination. Only files that have been added, modified, or deleted are transferred.

When creating or starting a task, you can configure options that determine which metadata in the source file system that you want to preserve. You can also configure your task's settings to keep or delete files in the destination even if they aren't found in the source file system.

After the TRANSFERRING phase is done, DataSync verifies consistency between the source and destination file systems. This is the VERIFYING phase. By default, DataSync performs a full consistency verification of your source and destination. DataSync rescans the content of the source and destination for any differences. If no differences are found, the task succeeds. Otherwise, the task is marked with a verification failure. For information about DataSync status, see Understanding Task Creation Statuses (p. 49).

How AWS DataSync Verifies Data Integrity

AWS DataSync locally calculates the checksum of every file in the source file system and the destination and compares them. Additionally, DataSync compares the metadata of every file in the source and destination and compares them. If there are differences in either one, verification fails with an error code that specifies precisely what failed. For examples, you see error codes such as Checksum failure, Metadata failure, Files were added, Files were removed, and so on.

For more information, see Understanding Task Creation Statuses (p. 49) and Enable verification in the Configuring Task Settings (p. 55) section.

How DataSync Handles Open and Locked Files

In general, DataSync can transfer open files without any limitations.

If a file is open and it's being written to during the transfer, DataSync detects data inconsistency in the VERIFYING phase. That is, this is when DataSync detects if the file on the source is different from the file on the destination.

If a file is locked and the server prevents DataSync from opening it, DataSync skips transferring it. DataSync logs an error during the TRANSFERRING phase and sends a verification error.
Setting Up

To get started, you first sign up for AWS. If you are a first-time user, we recommend that you read the regions and requirements section.

Topics
- Sign Up For AWS (p. 8)
- AWS Regions and Endpoints (p. 8)
- How to Access AWS DataSync (p. 8)
- DataSync Pricing (p. 8)

Sign Up For AWS

To use AWS DataSync, you need an AWS account that gives you access to all AWS resources, forums, support, and usage reports. You aren't charged for any of the services unless you use them. If you already have an AWS account, you can skip this step.

To sign up for AWS account
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.

AWS Regions and Endpoints

AWS DataSync is available in the following AWS Regions.

How to Access AWS DataSync

You can use the DataSync Management Console to perform various sync configuration and management tasks.

Additionally, you can use the AWS DataSync API or the AWS CLI to programmatically configure and manage DataSync. For more information about the API, see API Reference (p. 124).

You can also use the AWS SDKs to develop applications that interact with DataSync. The AWS SDKs for Java, .NET, and PHP wrap the underlying DataSync API to simplify your programming tasks. For information about downloading the SDK libraries, see Sample Code Libraries.

DataSync Pricing

For information about AWS DataSync pricing, see AWS DataSync Pricing on the DataSync pricing page.
Requirements for AWS DataSync

In this topic, you can find information about AWS DataSync agent requirements. These include supported hypervisors, virtual machine requirements, and Amazon EC2 instance requirements.

You can also find information about network configuration. This information includes network port configuration for public endpoints and VPC endpoints, and requirements for giving access through firewalls and routers.

Topics
- Agent Requirements (p. 9)
- Network Requirements for DataSync (p. 10)

Agent Requirements

In this section, you can find information about the requirements for an AWS DataSync agent, such as supported hypervisors and virtual machine requirements.

Topics
- Supported Hypervisors (p. 9)
- Virtual Machine Requirements (p. 9)
- Amazon EC2 Instance Requirements (p. 10)

Supported Hypervisors

You run DataSync on-premises as a virtual machine (VM).

DataSync supports the following hypervisor versions and hosts:

- VMware ESXi Hypervisor (version 6.0 or 6.5 or 6.7) – A free version of VMware is available on the VMware website. You also need a VMware vSphere client to connect to the host.

  Note
  When VMware ends general support for an ESXi hypervisor version, DataSync also ends support for that version. For information about VMware's supported hypervisor versions, see VMware Lifecycle Policy on the VMware website.

- Amazon EC2 instance – DataSync provides an Amazon Machine Image (AMI) that contains the DataSync VM image. For the recommended instance types, see See Amazon EC2 Instance Requirements (p. 10).

Virtual Machine Requirements

When deploying AWS DataSync on-premises, make sure that the underlying hardware where you deploy the DataSync VM can dedicate the following minimum resources:

- Virtual processors – Four virtual processors assigned to the VM.
- Disk space – 80 GB of disk space for installation of VM image and system data.
• **RAM** – Depending on your configuration, one of the following:
  - 32 GB of RAM assigned to the VM, for tasks to transfer EC2 instance types with up to 20 million files.
  - 64 GB of RAM assigned to the VM, for tasks to transfer more than 20 million files.

## Amazon EC2 Instance Requirements

When deploying AWS DataSync on Amazon EC2, the instance size must be at least 2xlarge for your AWS DataSync to function.

We recommend using one of the following instance types:

- m5.2xlarge – For tasks to transfer up to 20 million files.
- m5.4xlarge – For tasks to transfer more than 20 million files.

## Network Requirements for DataSync

Using DataSync to transfer your data requires access to certain network ports and endpoints. The following section describes how to configure network access for DataSync agents that transfer data through public service endpoints, Federal Information Processing Standard (FIPS) endpoints, and VPC endpoints.

### Topics

- Network Requirements to Connect to Your NFS or SMB Storage  (p. 10)
- Network Requirements When Using VPC Endpoints  (p. 11)
- Network Requirements When Using Public Service Endpoints or FIPS Endpoints (p. 14)

## Network Requirements to Connect to Your NFS or SMB Storage

Deploy the DataSync agent close to the source file system to minimize the distance traveled using native protocols such as Network File System (NFS). Doing this ensures that files travel over the network using our purpose-built, accelerated protocol. It keeps the distance traveled by using native, slower protocols to a minimum, and significantly speeds up transfers.

The following ports are required for communication between the DataSync agent and your NFS or Server Message Block (SMB) storage.

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>NFS server</td>
<td>TCP/UDP</td>
<td>2049 (NFS)</td>
<td>By the DataSync agent to mount a source NFS file system. Supports NFS v3.x, NFS v4.0, and NFS v4.1.</td>
</tr>
</tbody>
</table>
### Network Requirements When Using VPC Endpoints

When using only private IP addresses, you can ensure that your VPC can't be reached over the internet, and prevent any packets from entering or exiting the network. Doing this means that you can eliminate all internet access from your on-premises systems, and still use DataSync for data transfers to and from AWS using private IP addresses.

DataSync requires the following ports for its operation when your agent is using private endpoints.

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Your web browser</td>
<td>DataSync agent</td>
<td>TCP</td>
<td>80 (HTTP)</td>
<td>By your computer to obtain the agent activation key. After successful activation, DataSync closes the agent's port 80. The DataSync agent doesn't require port 80 to be publicly accessible. The required level of access to port 80 depends on your network configuration.</td>
</tr>
<tr>
<td>Agent</td>
<td>Your DataSync VPC endpoint</td>
<td>TCP</td>
<td>1024 – 1064</td>
<td>For control traffic between the DataSync agent and the AWS service.</td>
</tr>
</tbody>
</table>
and choose **Endpoints** from the navigation pane on the left. Choose the DataSync endpoint and check the **Subnets** list to find the private IP address that corresponds to the subnet that you chose for your VPC endpoint setup.

For more information, see step 5 in Configuring DataSync to Use Private IP Addresses for Data Transfer (p. 79).
<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>Your task's elastic network interfaces.</td>
<td>TCP</td>
<td>443 (HTTPS)</td>
<td>For data transfer from the DataSync VM to the AWS service.</td>
</tr>
<tr>
<td></td>
<td>To find the related IP addresses, open the Amazon EC2 console and choose <strong>Network Interfaces</strong> from the dashboard. Enter your task ID into the search filter to see the four elastic network interfaces for the task. For more information, see step 9 in Configuring DataSync to Use Private IP Addresses for Data Transfer (p. 79).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Your DataSync VPC endpoint</td>
<td>TCP</td>
<td>22 (Support channel)</td>
<td>To allow AWS Support to access your DataSync to help you with troubleshooting DataSync issues. You don't need this port open for normal operation, but it's required for troubleshooting.</td>
</tr>
</tbody>
</table>
Network Requirements When Using Public Service Endpoints or FIPS Endpoints

Your agent VM requires access to the following endpoints to communicate with AWS when using public service endpoints, or when using FIPS endpoints. Enabling this access is not necessary when using DataSync with VPC endpoints.

If you use a firewall or router to filter or limit network traffic, configure your firewall or router to allow these service endpoints. They’re required to enable outbound communication between your network and AWS.

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
<th>Endpoints Accessed by the Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Your web browser</td>
<td>DataSync agent</td>
<td>TCP</td>
<td>80 (HTTP)</td>
<td>Used by your computer to obtain the agent activation key. After successful</td>
<td></td>
</tr>
<tr>
<td>From</td>
<td>To</td>
<td>Protocol</td>
<td>Port</td>
<td>How Used</td>
<td>Endpoints Accessed by the Agent</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>----------</td>
<td>------</td>
<td>--------------------------------------------------------------------------</td>
<td>----------------------------------</td>
</tr>
</tbody>
</table>
| Agent  | AWS   | TCP      | 443 (HTTPS) | Used by the DataSync agent to activate with your AWS account. This is for agent activation only. You can block the endpoints after activation. | For public endpoint activation: activation.datasync. $region.amazonaws  
For FIPS endpoint activation: activation.datasync-fips. $region.amazonaws |

Activation, DataSync closes the agent's port 80.
The DataSync agent doesn't require port 80 to be publicly accessible. The required level of access to port 80 depends on your network configuration.
## Network Requirements When Using Public Service Endpoints or FIPS Endpoints

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
<th>Endpoints Accessed by the Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>AWS</td>
<td>TCP</td>
<td>443 (HTTPS)</td>
<td>For communication between the DataSync agent and the AWS service endpoint.</td>
<td>API endpoints:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>datasync.%region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Data transfer endpoints:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>#taskId.datasync-dp.%region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>cp.datasync.%region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$region.datasync.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Data transfer endpoints for FIPS:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>cp.datasync-fips.%region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$region.datasync.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agent updates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo.%region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo.default.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>packages.%region.amazonaws.com</td>
</tr>
<tr>
<td>Agent</td>
<td>Domain Name Service (DNS) server</td>
<td>TCP/UDP</td>
<td>53 (DNS)</td>
<td>For communication between DataSync agent and the DNS server.</td>
<td></td>
</tr>
</tbody>
</table>
## Network Requirements When Using Public Service Endpoints or FIPS Endpoints

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
<th>Endpoints Accessed by the Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>AWS</td>
<td>TCP</td>
<td>22 (Support channel)</td>
<td>Allows AWS Support to access your DataSync to help you with troubleshooting DataSync issues. You don't need this port open for normal operation, but it is required for troubleshooting.</td>
<td>54.201.223.107</td>
</tr>
</tbody>
</table>
## Network Requirements When Using Public Service Endpoints or FIPS Endpoints

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Protocol</th>
<th>Port</th>
<th>How Used</th>
<th>Endpoints Accessed by the Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>NTP server</td>
<td>UDP</td>
<td>123 (NTP)</td>
<td>Used by local systems to synchronize VM time to the host time.</td>
<td>0.amazon.pool.ntp.org</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.amazon.pool.ntp.org</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.amazon.pool.ntp.org</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.amazon.pool.ntp.org</td>
</tr>
</tbody>
</table>

**Note**

If you want to change the default NTP configuration of your VMware agent to use a different NTP server using the local console, see Configuring a Network Time Protocol (NTP) Server (p. 90).

Following is an illustration of the ports required by DataSync when using public service endpoints or FIPS endpoints.
Getting Started with AWS DataSync

In this topic, you can find step-by-step instructions on how to get started using AWS DataSync on the AWS Management Console.

Before you begin, we recommend reading How AWS DataSync Works (p. 3) to understand the components and terms used in DataSync and how DataSync works. We also recommend reading Using Identity-Based Policies (IAM Policies) for DataSync (p. 97) section to understand the AWS Identity and Access Management (IAM) permissions that DataSync requires.

To use AWS DataSync

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. At top right, choose the AWS Region where you want to run DataSync. The AWS Region should be the one where you plan to locate your Amazon S3 bucket, Amazon EFS file system, or Amazon FSx for Windows File Server file system. If you haven't created DataSync resources in this AWS Region, the DataSync home page appears.
3. On the DataSync home page, choose Get started to begin using DataSync.
   
   If this is your first time using DataSync in this AWS Region, the Create agent page appears. From this page, you can download your virtual machine (VM) or create an Amazon EC2 instance.
   
   If you have used DataSync in this AWS Region, the Agents page appears and you can see your agents listed.

Next, take the following steps.

Topics

• Create an Agent (p. 20)
• Configure a Source Location (p. 27)
• Configure a Destination Location (p. 29)
• Configure Task Settings (p. 30)
• Review Your Settings and Create Your Task (p. 31)
• Start Your Task (p. 32)
• Clean Up Resources (p. 32)

Create an Agent

To access your on-premises storage, you first deploy and activate an AWS DataSync agent. The activation process associates your agent with your AWS account.

Topics

• Deploy an AWS DataSync Agent (p. 20)
• Choose a Service Endpoint (p. 25)
• Activate Your Agent (p. 27)

Deploy an AWS DataSync Agent

You can deploy an AWS DataSync agent in either of the following ways:
AWS DataSync User Guide
Deploy an Agent

- Download and deploy an DataSync agent into your VMware environment and then activate it. For more information, see Deploy Your DataSync Agent on VMware (p. 21).
- Deploy an DataSync agent as an Amazon EC2 instance to transfer files to or from an in-cloud file system based on Network File System (NFS). An example is Amazon EFS. For more information, see Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21).

Your agent can connect to public internet endpoints or private endpoints in virtual private clouds (VPCs). The activation process associates your agent with your AWS account.

Topics
- Deploy Your DataSync Agent on VMware (p. 21)
- Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21)

Deploy Your DataSync Agent on VMware

You can download and deploy an AWS DataSync agent in your VMware environment and then activate it. You can also use an existing agent instead of deploying a new one. You can use a previously created agent if it can access your on-premises storage and if it's activated in the same AWS Region.

To deploy an agent on VMware

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. If you don't have an agent, on the Create agent page in the console, choose Download image in the Deploy agent section. Doing this downloads the agent and deploys it in your VMware ESXi hypervisor. The agent is available as a VM. If you want to deploy the agent as an Amazon EC2 instance, see Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21).

   AWS DataSync currently supports the VMware ESXi hypervisor. For information about hardware requirements for the VM, see Virtual Machine Requirements (p. 9). For information about how to deploy an .ova file in a VMware host, see the documentation for your hypervisor.

   If you have previously activated an agent in this AWS Region and want to use that agent, choose that agent and choose Create agent. The Configure a Source Location (p. 27) page appears.

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

   Note
   The VM's default credentials are the login admin and the password password.
   You can change the password on the local console. You don't need to log in to the VM for DataSync functionality. Login is mainly required for troubleshooting, such as running a connectivity test or opening a support channel with AWS. It's also required for network-specific settings, such as setting up a static IP address.

After you have deployed an agent, you choose a service endpoint (p. 25).

Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud

You can use your agent to transfer data between two locations in AWS, including cross-region and cross-account transfers. Doing this enables you to perform the following tasks:

- Transfer data from one EFS file system to another – migrate data from one AWS account to another, or periodically copy recently added files to a second EFS file system.
• Migrate from self-managed NFS to EFS – migrate to benefit from a more scalable, fully managed, elastic, and highly available file storage that has an NFS interface.

• Transfer data from Amazon S3 to in-cloud NFS, and from in-cloud NFS to Amazon S3 – use this approach for cases such as high-performance computing (HPC) processing.

To get started, choose the Amazon Machine Image (AMI) for your agent for the AWS Region where your EFS or self-managed NFS file system resides:

• To copy between EFS file systems, or from a self-managed NFS to EFS, create the EC2 agent in the source AWS Region.

• To copy from S3, create the agent in the destination AWS Region.

Important
We don't recommend using a DataSync agent that is deployed as an EC2 instance to read data from an on-premises source location. This approach doesn't deliver maximum throughput.

You can use the procedures following to transfer files from an in-cloud NFS file system to Amazon S3. In this case, the in-cloud NFS file system is an Amazon EFS file system.

To choose the agent AMI for your AWS Region

• Use the following CLI command to programmatically get the latest AMI ID for DataSync.

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

Example Example command and output

```
aws ssm get-parameter --name /aws/service/datasync/ami --region us-east-1

{
    "Parameter": {
        "Name": "/aws/service/datasync/ami",
        "Type": "String",
        "Value": "ami-01234db92d824a123",
        "Version": 6,
        "LastModifiedDate": 1569946277.996,
        "ARN": "arn:aws:ssm:us-east-1:parameter/aws/service/datasync/ami"
    }
}
```

You can also identify the AMI ID for your AWS Region in the following table. You use this AMI ID to deploy your DataSync agent. For the recommended instance types, see Amazon EC2 Instance Requirements (p. 10).

If you activated an agent in this AWS Region and want to use that agent, choose the agent and choose Create agent. The Configure a Source Location (p. 27) page appears.

In the following table, you can find the available DataSync AMIs by AWS Region.

<table>
<thead>
<tr>
<th>AWS Region</th>
<th>AMI Name</th>
<th>AMI ID</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ap-east-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-0a2228f0c42627cf4</td>
<td>Launch instance</td>
</tr>
<tr>
<td>AWS Region</td>
<td>AMI Name</td>
<td>AMI ID</td>
<td>URL</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------</td>
<td>-----------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ap-northeast-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-05c17aa2b793ebd3e</td>
<td>Launch instance</td>
</tr>
<tr>
<td>ap-northeast-2</td>
<td>aws-datasync-1578601791</td>
<td>ami-04709cb141754b5a2</td>
<td>Launch instance</td>
</tr>
<tr>
<td>ap-south-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-0b3027fa88024fa10</td>
<td>Launch instance</td>
</tr>
<tr>
<td>ap-southeast-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-05e784c943c9d3bc</td>
<td>Launch instance</td>
</tr>
<tr>
<td>ap-southeast-2</td>
<td>aws-datasync-1578601791</td>
<td>ami-017f985d24cf954d5</td>
<td>Launch instance</td>
</tr>
<tr>
<td>ca-central-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-0678ddd97941bb7fb</td>
<td>Launch instance</td>
</tr>
<tr>
<td>eu-central-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-07bdc05550c9e299</td>
<td>Launch instance</td>
</tr>
<tr>
<td>eu-north-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-0410d0dc8bc70f33b</td>
<td>Launch instance</td>
</tr>
<tr>
<td>eu-west-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-0122531d38a12bf9</td>
<td>Launch instance</td>
</tr>
<tr>
<td>eu-west-2</td>
<td>aws-datasync-1578601791</td>
<td>ami-059bd3460ccf61554</td>
<td>Launch instance</td>
</tr>
<tr>
<td>eu-west-3</td>
<td>aws-datasync-1578601791</td>
<td>ami-05c6a0e4c16de4be</td>
<td>Launch instance</td>
</tr>
<tr>
<td>me-south-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-0cbe61a092cd25ad7</td>
<td>Launch instance</td>
</tr>
<tr>
<td>sa-east-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-04ad9afea2d0b2b3f0</td>
<td>Launch instance</td>
</tr>
<tr>
<td>us-east-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-03a252bd5a382c840</td>
<td>Launch instance</td>
</tr>
<tr>
<td>us-east-2</td>
<td>aws-datasync-1578601791</td>
<td>ami-007674065fe323f83</td>
<td>Launch instance</td>
</tr>
<tr>
<td>us-west-1</td>
<td>aws-datasync-1578601791</td>
<td>ami-00a8467244673ddcb</td>
<td>Launch instance</td>
</tr>
<tr>
<td>us-west-2</td>
<td>aws-datasync-1578601791</td>
<td>ami-0e1afcd862bc1329d</td>
<td>Launch instance</td>
</tr>
<tr>
<td>us-gov-east-1</td>
<td>aws-datasync-1578601568</td>
<td>ami-022e97af081000ce6</td>
<td>Launch instance</td>
</tr>
<tr>
<td>us-gov-west-1</td>
<td>aws-datasync-1578601568</td>
<td>ami-ca6150ab</td>
<td>Launch instance</td>
</tr>
</tbody>
</table>
To deploy your DataSync agent as an EC2 instance

1. From the AWS account where the source EFS resides, launch the agent using your AMI from the Amazon EC2 launch wizard. Use the following URL to launch the AMI.


   In the URL, replace the `source-efs-or-nfs-region` and `ami-id` with your own source AWS Region and AMI ID. The Choose an Instance Type page appears on the Amazon EC2 console. For a list of AMI IDs by AWS Region, see Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21).

2. Choose one of the recommended instance types for your use case, and choose Next: Configure Instance Details. For the recommended instance types, see Amazon EC2 Instance Requirements (p. 10).

3. On the Configure Instance Details page, do the following:
   a. For Network, choose the virtual private cloud (VPC) where your source EFS or NFS file system is located.
   b. For Auto-assign Public IP, choose a value. For your instance to be accessible from the public internet, set Auto-assign Public IP to Enable. Otherwise, set Auto-assign Public IP to Disable. If a public IP address isn't assigned, activate the agent in your VPC using its private IP address.

   When you transfer files from an in-cloud NFS, to increase performance we recommend that you choose a Placement Group value where your NFS server resides.

4. Choose Next: Add Storage. The agent doesn't require additional storage, so you can skip this step and choose Next: Add tags.

5. (Optional) On the Add Tags page, you can add tags to your EC2 instance. When you're finished on the page, choose Next: Configure Security Group.

6. On the Configure Security Group page, do the following:
   a. Make sure that the selected security group allows inbound access to HTTP port 80 from the web browser that you plan to use to activate the agent.
   b. Make sure that the security group of the source EFS or NFS system allows inbound traffic from the agent. In addition, make sure that the agent allows outbound traffic to the source EFS or NFS system. The traffic goes through the standard NFS port, 2049.

   For the complete set of network requirements for DataSync, see Network Requirements for DataSync (p. 10).

   If you deploy your agent using a VPC endpoint, you need to allow additional ports. For information, see How DataSync Works with VPC Endpoints (p. 79).

7. Choose Review and Launch to review your configuration, then choose Launch to launch your instance. Remember to use a key pair that's accessible to you. A confirmation page appears and indicates that your instance is launching.

8. Choose View Instances to close the confirmation page and return to the EC2 instances screen. When you launch an instance, its initial state is pending. After the instance starts, its state changes to running. At this point, it is assigned a public Domain Name System (DNS) name and IP address, which can be found in the Descriptions tab.

9. If you set Auto-assign Public IP to Enable, choose your instance and note the public IP address in the Description tab. You use this IP address later to connect to your sync agent.
If you set **Auto-assign Public IP** to **Disable**, launch or use an existing instance in your VPC to activate the agent. In this case, you use the private IP address of the sync agent to activate the agent from this instance in the VPC.

After you have deployed an agent, you choose a service endpoint (p. 25).

### Choose a Service Endpoint

You can activate your agent by using one of the following endpoint types:

- **Public endpoints** – If you use public endpoints, all communication from your DataSync agent to AWS occurs over the public internet. For instructions, see Choose a Public Service Endpoint (p. 25).
- **Federal Information Processing Standard (FIPS) endpoints** – If you need to use FIPS 140-2 validated cryptographic modules when accessing the AWS GovCloud (US-East) or AWS GovCloud (US-West) Region, use this endpoint to activate your agent. You use the AWS CLI or API to access this endpoint. For more information, see Federal Information Processing Standard (FIPS) 140-2.
- **Virtual private cloud (VPC) endpoints** – If you use a VPC endpoint, all communication from DataSync to AWS services occurs through the VPC endpoint in your VPC in AWS. This approach provides a private connection between your on-premises data center, your VPC, and AWS services. It increases the security of your data as it is copied over the network. For instructions, see Using AWS DataSync in a Virtual Private Cloud (p. 79).

**Note**

After you choose a service endpoint type and activate your agent, you can't change it to use a different service endpoint type later. If you need to transfer data to multiple endpoint types, create a DataSync agent for each endpoint type that you use.

For more information about service endpoints, see AWS DataSync in the **AWS General Reference**.

### Topics

- Choose a Public Service Endpoint (p. 25)
- Choose a FIPS Service Endpoint (p. 25)
- Choose a VPC Endpoint (p. 26)

### Choose a Public Service Endpoint

If you use a public endpoint, all communication from your DataSync agent to AWS occurs over the public internet.

**To choose a public service endpoint**

1. From the DataSync **Agents** page on the DataSync Management Console, choose **Create agent** to open the Create agent page.
2. In the **Service endpoint** section, choose **Public service endpoints in AWS Region name**. For a list of supported AWS Regions, see AWS DataSync in the **AWS General Reference**.

**Next Step:** the section called “Activate Your Agent” (p. 27)

### Choose a FIPS Service Endpoint

If you use a FIPS service endpoint, DataSync communicates with the AWS GovCloud (US) or Canada (Central) Region.
To choose a FIPS service endpoint

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. Choose Agents, and then choose Create agent to open the Create agent page.
3. In the Service endpoint section, choose the FIPS endpoint that you want. For information about supported FIPS endpoint, see AWS DataSync in the AWS General Reference.

Next Step: the section called “Activate Your Agent” (p. 27)

Choose a VPC Endpoint

If you use a VPC endpoint, all communication from DataSync to AWS services occurs through the VPC endpoint in your VPC in AWS. This approach provides a private connection between your on-premises data center, your VPC, and AWS services.

You can also use a VPC endpoint outside your VPC to connect your data center directly to AWS resources. In this case, you use a virtual private network (VPN) or AWS Direct Connect. You set up a VPC route table to use the endpoint to access the service. For detailed information, see Routing for Gateway Endpoints.

To choose a VPC endpoint

1. Create a VPC endpoint. For instructions, see Creating an Interface Endpoint. If you already have a VPC endpoint in the AWS Region, you can use it.
   
   **Important**
   
   In step 4 of the instructions mentioned preceding, choose com.amazonaws.region.datasync for Service Name in the table of endpoints. For information about supported AWS Regions, see AWS DataSync in the AWS General Reference.

2. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
3. From the Agents page, choose Create agent to open the Create agent page.
4. In the Service endpoint section, choose VPC endpoints using AWS PrivateLink. This is the VPC endpoint that the agent has access to.

   ![VPC endpoint screenshot](image-url)

5. For VPC Endpoint, choose the private VPC endpoint that you want your agent to connect to. You noted the endpoint ID when you created the VPC endpoint.
6. For Subnet, choose the subnet in which you want to run your task. This is the subnet where the elastic network interface is created.
7. For Security Group, choose a security group for your task. This is the security group that protects your network interface for tasks that run on your agent.
Activate Your Agent

To activate your agent, first get the IP address of your agent and use it to get an activation key. The activation key securely associates the agent with your AWS account.

The activation process requires the agent's port 80 to be accessible from your browser. After the agent is activated, it closes port 80 and the port is no longer accessible. For information about ports and network requirements, see Network Requirements for DataSync (p. 10).

To activate your agent

1. In the **Service Endpoint** section, choose the endpoint type for this agent.

   A DataSync agent can transfer data through public service endpoints, Federal Information Processing Standard (FIPS) endpoints, and VPC endpoints. For more information about endpoint activation, see Creating and Activating an Agent (p. 78).

   **Note**

   When you configure your agent to use VPC endpoints, the data transferred between your agent and the DataSync service doesn't cross the public internet. Thus, this data doesn't require public IP addresses. For more information about configuring your agent to use VPC endpoints, see Using AWS DataSync in a Virtual Private Cloud (p. 79).

2. In the **Activation key** section, enter the agent's IP address or the domain name for your agent for **Agent address**, and choose **Get key**. Your browser connects to the IP address and gets a unique activation key from your agent.

   Activation information appears in the **Activation key** section. If activation succeeds, the activation key is displayed. If the activation fails, make sure that your security group is configured properly and check your firewall to make sure that you allowed the required ports.

3. (Optional) For **Agent name**, enter a name for your agent.

4. (Optional) For **Tags**, enter a key and value to add a tag to your agent. A **tag** is a key-value pair that helps you manage, filter, and search for your agents.

5. Choose **Create agent**. Your agent is listed on the **Agents** page. In the **Service endpoint** column, verify that your service endpoint is correct.

6. In the **Tasks** section of the page, choose **Create task**. The **Configure source location** page appears.

Configure a Source Location

A **task** consists of a pair of locations that data will be transferred between. The **source location** defines the storage system or service that you want to read data from. The **destination location** defines the storage system or service that you want to write data to.
For a list of all DataSync supported source and destination endpoints, see Working with Locations (p. 64).

In the following walkthrough, we give an example of configuring a Network File System (NFS) file system as the source location.

To configure a different location type as your source location, see the following topics:

- Creating a Location for NFS (p. 65)
- Creating a Location for SMB (p. 66)
- Creating a Location for Amazon EFS (p. 67)
- Creating a Location for Amazon FSx for Windows File Server (p. 69)
- Creating a Location for Amazon S3 (p. 70)

To create an NFS location

1. On the Configure source location page, choose Create a new location or Choose existing location. Create a new location enables you to define a new location and Choose existing location enables you to choose from locations that you have previously created in this AWS Region.

2. For Location type in the Configuration section, choose your NFS server from the list.

3. For Agents, choose your agent from the list. You can add more than one agent. For this walkthrough, we add only one agent.

   Note
   In many cases, you might be transferring from an in-cloud NFS file system or an Amazon EFS file system. In such cases, make sure that you choose an agent that you created in an Amazon EC2 instance that can access this file system.
   You can't use agents that are created with different endpoint types for the same task.

4. For NFS server, enter the IP address or domain name of your NFS server. An agent that is installed on-premises uses this host name to mount the NFS server in a network. The NFS server should allow full access to all files.
5. For **Mount path**, enter a path that's exported by the NFS server, or a subdirectory that can be mounted by other NFS clients in your network. The path is used to read data from or write data to your NFS server.

6. Choose **Next** to open the **Configure destination location** page.

Configure a Destination Location

A task consists of a pair of locations that data will be transferred between. The **source location** defines the storage system or service that you want to read data from. The **destination location** defines the storage system or service that you want to write data to.

For a list of all DataSync supported source and destination endpoints, see Working with Locations (p. 64).

**Note**

When copying between two Amazon EFS file systems, we recommend using the NFS (source) to EFS (destination) transfer. When copying between two Amazon FSx for Windows File Server file systems, we recommend using the SMB (source) to Amazon FSx for Windows File Server (destination) transfer.

In the following walkthrough, we give an example of configuring an S3 bucket as the destination location. If you don't have an S3 bucket in the current AWS Region, create one. For information about how to create an Amazon S3 bucket, see **Create a Bucket** in the Amazon Simple Storage Service Getting Started Guide.

To configure a different location type, see the following topics:

- Creating a Location for NFS (p. 65)
- Creating a Location for SMB (p. 66)
- Creating a Location for Amazon EFS (p. 67)
- Creating a Location for Amazon FSx for Windows File Server (p. 69)
- Creating a Location for Amazon S3 (p. 70)

To create an S3 location

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose **Locations**. You see the locations that you previously created.
3. On the **Create location** page, choose **Create location**.
4. For **Type**, choose **Amazon S3 bucket**.
5. For **S3 Bucket**, choose the S3 bucket that you want to use as an endpoint. You configure this location as a source or destination later.
6. For **S3 storage class**, choose a storage class that you want to transfer objects directly into. For information about storage classes, see Amazon S3 Storage Classes. Some storage classes have behaviors that can affect your S3 cost. For more information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).
7. For **Folder**, provide the name of a folder. This folder in Amazon S3 is the one that is used to read data from the S3 source location or write data to the S3 destination.
8. For **IAM role**, choose **Autogenerate** for DataSync to automatically create a role with the required permissions.

If DataSync has previously created such an IAM role for that S3 bucket, that role is chosen as the default in the list. You can also manually create your own role and choose it from the list. For instructions on how to create an IAM role manually, see Manually Configuring an IAM Role to Access Your S3 Bucket (p. 73).
Configure Task Settings

Now that you have an agent and have configured source and destination locations, you configure settings for your task. A task is a set of two locations (source and destination) and a set of options that you use to control the behavior of a task. If you don't specify options when you create a task, AWS DataSync populates the options with service default settings.

To configure settings

1. On the Configure settings page, enter a name for your task for Task name - optional.
2. In the Options section, configure the options that you want to use to run your task.

   These options control the behavior of a task execution. Behavior includes preserving metadata such as the user ID (UID) or group ID (GID), preserving file permissions, and data integrity verification. If you don't specify values for these options, DataSync uses a set of default options that can be overridden for a task execution.

   Available options are as follows:

   • **Verify data** – Set an optional verification to compare source and destination at the end of the transfer. As DataSync transfers data, it always performs data integrity checks during the transfer. You can enable additional verification to compare source and destination at the end of a transfer. As part of that, you can check either the entire dataset or just the files or objects that DataSync transferred.

     Data verification options include the following:

     • **Verify only the data transferred** (recommended) calculates the checksum of transferred files and metadata on the source. It then compares this checksum to the checksum calculated on those files at the destination at the end of the transfer. We recommend this option when transferring to S3 Glacier or S3 Glacier Deep Archive storage classes. For more information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

     • **Verify all data in the destination** performs a scan at the end of the transfer of the entire source and entire destination to verify that source and destination are fully synchronized. This option isn't supported when transferring to S3 Glacier or S3 Glacier Deep Archive storage classes. For more information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

     • **Check integrity during the transfer** doesn't run additional verification at the end of the transfer. All data transmissions are still integrity-checked with checksum verification during the transfer.

     • **Copy ownership** – If you choose this option, DataSync copies file ownership such as the group ID of the file's owners and the user ID of the file's owner.

     • **Copy permissions** – If you choose this option, DataSync copies file POSIX permissions from the source to the destination.

     • **Copy timestamps** – If you choose this option, DataSync copies the timestamp metadata from the source to the destination.

     • **Keep deleted files** – If you choose this option, DataSync keeps files in the destination that don't exist in the source file system.
If your task deletes objects, you might incur minimum storage duration charges for certain storage classes. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

- **Overwrite files** – If you choose this option, files at the destination are overwritten by files from the source. If you don’t choose this option, the destination file isn’t replaced by the source file, even if the destination file differs from the source file.

If your task overwrites objects, you might incur minimum storage duration charges for certain storage classes. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

- **Use available** or **Set bandwidth limit (MiB/s)** — If you choose **Use available**, DataSync uses all the network bandwidth that is available for the transfer. If you choose **Set bandwidth limit (MiB/s)**, you limit the maximum bandwidth that you want DataSync to use for this task.

- **Queueing** – If you are using a single agent to run multiple tasks, choose this option to make the tasks run in series (first in, first out). For more information, see Queueing Task Executions (p. 57).

3. In the Filtering configuration - Optional section, enter a pattern to use as a filter. This pattern defines the criteria for specific files, folders, and objects to exclude from your transfer. To add additional patterns to your filter, choose Add pattern. For detailed information about filtering and syntax for creating patterns, see Filtering the Data Transferred by AWS DataSync (p. 58).

If you want to specify an include filter to transfer only specific files, you can do that later when you start a task. For more information, see Including Data in a Transfer (p. 60).

4. In the Schedule (optional) section, configure your task to run on a schedule that you specify.

5. For Frequency, configure how frequently you want the task to run. For frequency configuration options, see Configuring a Task Schedule (p. 62).

6. In the Tags - optional section, enter Key and Value to tag your task. A tag is a key-value pair that helps you manage, filter, and search for your tasks. We recommend that you create a name tag for your task.

7. **Task logging - optional** – If you choose this option, DataSync publishes logs for individual files or objects to the CloudWatch log group that you specify.

   To upload logs to your CloudWatch log group, DataSync requires a resource policy that grants sufficient permissions. For an example of such a policy, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).

   For more information about using log groups and streams, see Working with Log Groups and Log Streams in the Amazon CloudWatch Logs User Guide.

   Use the Log level option to set the level of detail that is logged to CloudWatch Logs. Log level options include the following:

   - **Log basic information such as transfer errors** – If you choose this option, only basic information such as transfer errors is published to CloudWatch.
   - **Log all transferred objects, files, and folders** – If you choose this option, log records are published to CloudWatch Logs for all files or objects that the task copies and integrity checks.
   - **Do not send logs to CloudWatch** – If you choose this option, DataSync logs are not published to CloudWatch.

8. Choose Next to open the Review page.

### Review Your Settings and Create Your Task

Next, you review your settings and create your task.
To review your settings

1. On the Review page, review and edit your configuration and settings if necessary. You can edit the settings on the page by choosing Previous at the bottom of the page. For more information about task settings, see Creating a Task (p. 50).

2. When you are done reviewing, choose Create task. The Status value of the task is now Creating. During the Creating status, AWS DataSync attempts to mount the source NFS location. Wait for the task to transition to the Available status before you run the task.

Start Your Task

Next, you start your task. You can further review your configuration settings before you start the task,

To start your task

1. When the Status of the task changes from Creating to Available, choose Start to open the Start task page.

2. On the Start task page, choose Start.

3. (Optional) In the Filtering configuration - Optional section, enter a pattern to use as a filter. This pattern defines the criteria for specific files, folders, and objects that your task transfers. These files, folders, and objects are the only ones that your task transfers. To add additional patterns to your filter, choose Add pattern. For detailed information about filtering and syntax for creating patterns, see Filtering the Data Transferred by AWS DataSync (p. 58).

4. After you have chosen your options and added your filters, choose Start.

When you create a task, it first enters the Creating state. During Creating, AWS DataSync attempts to mount the on-premises Network File System (NFS) location. The task transitions to the Available state without waiting for the AWS location to become mounted. If required, AWS DataSync mounts the AWS location before each task execution.

If an agent that is associated with an NFS location goes offline, the task transitions to the Unavailable status. If the task remains in the Creating status for more than a few minutes, your agent might be having trouble mounting the NFS file system. Check the task’s ErrorCode and ErrorDetail values. Mount issues are often caused by either a misconfigured firewall or a mistyped NFS server host name.

For information about how DataSync transfers files, see How DataSync Transfers Files (p. 7).

Clean Up Resources

If you used DataSync for a test or don't need the resources you created, delete them so you don't get charged for resources you're not using.

To clean up resources

1. Delete tasks that you don't need. For instructions about how to delete a task, see Deleting a Task (p. 63).

2. Delete locations that you don't need. For instructions on how to delete a location, see Deleting a Location (p. 76).

3. Delete agents that you don't need. For instructions about how to delete an agent, see Deleting an Agent (p. 83).
Using the AWS Command Line Interface with AWS DataSync

In this section, you can find examples of using the AWS Command Line Interface (AWS CLI) commands for AWS DataSync. You can use these commands to create an agent, create source and destination locations, and run a task.

Before you begin, we recommend reading How AWS DataSync Works (p. 3) to understand the components and terms used in DataSync and how DataSync works. We also recommend reading Using Identity-Based Policies (IAM Policies) for DataSync (p. 97) to understand the AWS Identity and Access Management (IAM) permissions that DataSync requires.

Before you use AWS CLI commands, install the AWS CLI. For information about how to install the AWS CLI, see Installing the AWS Command Line Interface in the AWS Command Line Interface User Guide. After you install the CLI, you can use the `help` command to see the DataSync operations and the parameters associated with them.

To see the available operations, enter the following command.

```bash
aws datasync help
```

To see the parameters associated with a specific operation, enter the following command.

```bash
aws datasync operation help
```

For more information about the AWS CLI, see What Is the AWS Command Line Interface?

Topics

- Step 1: Create an Agent (p. 33)
- Step 2: Create Locations (p. 36)
- Step 3: Create a Task (p. 42)
- Step 4: Start a Task Execution (p. 43)
- Step 5: Use the CLI to Monitor Your Task Execution (p. 43)

For information about supported AWS Regions and endpoints, see DataSync AWS Regions and Endpoints.

For information about DataSync Amazon Resource Name (ARN) values, see DataSync Amazon Resource Names.

Step 1: Create an Agent

Use the following procedure to create an agent that reads from a source. This source can be a Network File System (NFS) location that is on-premises or in the cloud, or a Server Message Block (SMB) location.
A DataSync agent can transfer data through public service endpoints, Federal Information Processing Standard (FIPS) endpoints, and VPC endpoints. To learn more, see Creating and Activating an Agent (p. 78)

**Note**
When you configure your agent to use VPC endpoints, the data transferred between your agent and the DataSync service doesn't cross the public internet and doesn't require public IP addresses. For end-to-end instructions for this configuration, see Using AWS DataSync in a Virtual Private Cloud (p. 79).

**To create an agent to read from an NFS or SMB source file system**

1. Download the current DataSync .ova image or launch the current DataSync Amazon Machine Image (AMI) based on Amazon EC2 from the AWS DataSync Management Console. For information about how to get the .ova image or EC2 AMI, see Create an Agent (p. 20). For information about hardware requirements and recommended EC2 instance types, see Virtual Machine Requirements (p. 9).

2. Make sure that you satisfy the network connectivity requirements for the agent. For information about network requirements, see Network Requirements for DataSync (p. 10).

3. Deploy the .ova image in your on-premises hypervisor, power on the hypervisor, and note the `agent-ip-address`. Make sure that you can reach the agent on port 80. You can use the following command to check.

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

   **Note**
   The .ova default credentials are login `admin`, password `password`. You can change the password on the VM local console. You don't need to log in to the VM for basic DataSync functionality. Login is mainly required for troubleshooting, network-specific settings, and so on.
   You log in to the agent VM local console using the VMware vSphere client. For information about how to use the VM local console, see Configuring Your Agent on the VM Local Console (p. 84).

4. Send an HTTP/1.1 GET request to the agent to get the activation key. You can do this by using standard Unix tools:

   - To activate an agent using a public service endpoint, use the following command.

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

   - To activate an agent using a virtual private cloud (VPC) endpoint, use the Elastic IP address of the VPC endpoint. Use the following command.

     ```bash
     $ curl "http://agent-ip-address/?gatewayType=SYNC&activationRegion=aws-region&privateLinkEndpoint=Elastic IP address of VPC endpoint&endpointType=PRIVATE_LINK&no_redirect"
     ```

   To find the correct IP address, open the Amazon VPC console at https://console.aws.amazon.com/vpc/ and choose **Endpoints** from the navigation pane at left. Choose the DataSync endpoint, and check **Subnets list** to find the private IP address that corresponds to the subnet that you chose for your VPC endpoint setup.

   For more information about VPC endpoint configuration, see step 5 in Configuring DataSync to Use Private IP Addresses for Data Transfer (p. 79).
AWS DataSync User Guide
Step 1: Create an Agent

- To activate an agent using a Federal Information Processing Standard (FIPS) endpoint, specify endpointType=FIPS. Also, the activationRegion value must be set to an AWS Region within the United States. To activate a FIPS endpoint, use the following command.

```
$ curl "http://agent-IP-address/?gatewayType=SYNC&activationRegion=US-based-aws-region&endpointType=FIPS&no_redirect"
```

This command returns an activation key similar to the one following.

F0EFT-7FPPR-GG7MC-3I9R3-27DOH

5. After you have the activation key, do one of the following:

- To activate your agent using a public endpoint or FIPS endpoint, use the following command.

```
$ aws datasync create-agent --agent-name agent's name --activation-key obtained activation key
```

- To activate your agent using a VPC endpoint, use the following command.

```
$ aws datasync create-agent --agent-name agent's name --vpc-endpoint-id vpc endpoint id --subnet-arns subnet arns --security-group-arns security group arns --activation-key obtained activation key
```

In this command, use the following arguments:

- **vpc endpoint id** – The AWS endpoint that the agent connects to. To find the endpoint ID, open the Amazon VPC console at https://console.aws.amazon.com/vpc/, and choose Endpoints from the navigation pane on the left. Copy the **Endpoint ID** value of the DataSync endpoint. For more information about VPC endpoint configuration, see step 5 in Configuring DataSync to Use Private IP Addresses for Data Transfer (p. 79).

- **security group arn** – The Amazon Resource Name (ARN) of the security group to use for the task's endpoint. This is the security group that you created in step 3 of Configuring DataSync to Use Private IP Addresses for Data Transfer (p. 79).

- **task's subnet arn** – The ARN of the subnet where the task endpoints for the agent are created.

This is the subnet that you chose in step 1 of Configuring DataSync to Use Private IP Addresses for Data Transfer (p. 79).

These commands return the ARN of the agent that you just activated. The ARN is similar to the one following.

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

**Note**
After you choose a service endpoint, you can't change it later.

After the agent is activated, it closes port 80 and the port is no longer accessible. If you can't connect to the agent after you have activated it, check whether the activation was successful by using this command: $ aws datasync list-agents.

**Note**
Make sure that you are using the same AWS credentials throughout the whole process. Don't switch between multiple terminals where you are authenticated with different AWS credentials.
Step 2: Create Locations

Each DataSync task is made up of a pair of locations between which data is transferred. The source location defines the storage system or service that you want to read data from. The destination location defines the storage system or service that you want to write data to.

For a list of all DataSync supported source and destination endpoints, see Working with Locations (p. 64).

Topics
- Create an NFS Location (p. 36)
- Create an SMB Location (p. 37)
- Create an Amazon EFS Location (p. 37)
- Create an Amazon FSx for Windows File Server Location (p. 39)
- Create an Amazon S3 Location (p. 40)

Create an NFS Location

Use the following procedure to create an NFS location using the AWS CLI. An NFS location defines a file system on an NFS server that can be read from or written to. You can also create an NFS location using the AWS Management Console. For more information, see Creating a Location for NFS (p. 65).

To create an NFS location using the CLI
- Use the following command to create an NFS source location.

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

For the preceding command, the following applies:
- The path that you provide for the `--subdirectory` parameter should be a path that's exported by the NFS server, or a subdirectory. Other NFS clients in your network should be able to mount this path. To see all the paths exported by your NFS server, run the command `showmount -e nfs-server-address` from an NFS client with access to your server. You can specify any directory that appears in the results, and any subdirectory of that directory.
- To transfer all the data in the folder that you specified, DataSync needs permissions to read all the data. To give DataSync permissions, you can do one of two things. You can configure the NFS export with `no_root_squash`. Or, for the all files that you want DataSync to access, you can make sure that the permissions allow read access for all users. Doing either enables the agent to read the files. For the agent to access directories, you must additionally give all users execute access.
- Make sure that the NFS export is accessible without Kerberos authentication.

DataSync automatically chooses the NFS version that it uses to read from an NFS location. To specify an NFS version, use the optional `Version` parameter in the `NfsMountOptions (p. 220)` API operation.

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

Use the following procedure to create an SMB location using the AWS CLI. An SMB location defines a file system on an SMB server that can be read from or written to. You can also create an SMB location using the console. For more information, see Creating a Location for SMB (p. 66).

To create an SMB location using the CLI

1. Use the following command to create an SMB source location.

```bash
$ aws datasync create-location-smb --server-hostname smb-server-address --user user-name --domain domain-of-the-smb-server --password user's-password AgentArns=agent-arns --subdirectory smb-export-path
```

The path that you provide for the --subdirectory parameter should be a path that's exported by the SMB server, or a subdirectory. Specify the path using forward slashes, for example /path/to/folder. Other SMB clients in your network should be able to access this path.

DataSync automatically chooses the SMB version that it uses to read from an SMB location. To specify an SMB version, use the optional Version parameter in the SmbMountOptions (p. 229) API operation.

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

```json
{ "LocationArn": "arn:aws:datasync:us-east-1:11122233444:location/loc-0f01451b140b2af49" }
```

Create an Amazon EFS Location

Use the following procedure to create an EFS location using the AWS CLI. An EFS location is the endpoint for an Amazon EFS file system, which defines an EFS file system that can be read from or written to. You can also create an EFS location using the console. For more information, see Creating a Location for Amazon EFS (p. 67).

To create an Amazon EFS location using the CLI

1. If you don't have an Amazon EFS file system, create one. For information about how to create an EFS file system, see Getting Started with Amazon Elastic File System in the Amazon Elastic File System User Guide.
2. Identify a subnet that has at least one mount target for that file system. You can see all the mount targets and the subnets associated with an EFS file system by using the `describe-mount-targets` command.

```bash
# aws --region aws-region efs describe-mount-targets --file-system-id file-system-id
```

**Note**
The AWS Region that you specify is the one where your target S3 bucket or EFS file system is located.

This command returns information about the target similar to the information shown following.

```json
{
   "MountTargets": [
      {
         "OwnerId": "111222333444",
         "MountTargetId": "fsmt-2234a10",
         "FileSystemId": "fs-123456ab",
         "SubnetId": "subnet-f12a0e34",
         "LifeCycleState": "available",
         "IpAddress": "11.222.0.123",
         "NetworkInterfaceId": "eni-1234a044"
      }
   ]
}
```

3. Specify an Amazon EC2 security group that can be used to access the mount target. You can run the following command to find out the security group of the mount target.

```bash
# aws --region aws-region efs describe-mount-target-security-groups --mount-target-id mount-target-id
```

The security group that you provide needs to be able to communicate with the security group on the mount target in the subnet specified.

The relationship between security group M on the mount target and security group S, which you provide for DataSync to use at this stage, is as follows:

- Security group M, which you associate with the mount target, must allow inbound access for the TCP protocol on the NFS port (2049) from security group S.

You can enable an inbound connection either by its IP address (CIDR range) or its security group.

- Security group S, which you provide to DataSync to access EFS, should have a rule that enables outbound connections to the NFS port. It enables outbound connections on one of the file system's mount targets.

You can enable outbound connections either by IP address (CIDR range) or security group.

For information about security groups and mount targets, see Security Groups for Amazon EC2 Instances and Mount Targets in the Amazon Elastic File System User Guide.

4. Create the EFS location. To create the EFS location, you need the ARNs for your Amazon EC2 subnet, EC2 security group, and an EFS file system. Because the DataSync API accepts fully qualified ARNs, you can construct these ARNs. For information about how to construct ARNs for different services, see Amazon Resource Names (ARNs) in the AWS General Reference.

Use the following command to create an EFS location.
Create an Amazon FSx for Windows File Server Location

Use the following procedure to create an Amazon FSx for Windows File Server location using the AWS CLI. An Amazon FSx location is the endpoint for an Amazon FSx for Windows File Server. This endpoint defines the Amazon FSx file share that you can read from or write to.

You can also create an Amazon FSx location using the console. For more information, see Creating a Location for Amazon FSx for Windows File Server (p. 69)

To create an Amazon FSx for Windows File Server location using the CLI

- Use the following command to create an Amazon FSx location.

```bash
# aws datasync create-location-fsx-windows \
--fsx-filesystem-arn arn:aws:fsx:region:account-id:file-system/filesystem-id \
--security-group-arns arn:aws:ec2:region:account-id:security-group/group-id \
--user smb-user --password password
```

In the create-location-fsx-windows command, specify the following:

- The fully qualified Amazon Resource Name (ARN) of the file system that you want to read from or write to.

The DataSync API accepts fully qualified ARNs, and you can construct these ARNs. For information about how to construct ARNs for different services, see Amazon Resource Names (ARNs) in the AWS General Reference.

- The ARN of an Amazon EC2 security group that can be applied to the Elastic Network Interfaces of the file system's preferred subnet. For more information, see Creating a VPC with an Instance Tenancy of Dedicated in the Amazon EC2 User Guide.

- The AWS Region. The Region that you specify is the one where your target Amazon FSx file system is located.

The preceding command returns a location ARN similar to the one shown following.

```json
{ "LocationArn": "arn:aws:datasync:us-west-2:111222333444:location/loc-07db7abfc326c50fb" }
```
Create an Amazon S3 Location

Use the following procedure to create an Amazon S3 location using the AWS CLI. An Amazon S3 location requires an Amazon S3 bucket that can be read from or written to. To create an Amazon S3 bucket, see Create a Bucket in the Amazon S3 Console User Guide.

For DataSync to access a destination Amazon S3 bucket, DataSync needs an AWS Identity and Access Management (IAM) role that has the required permissions. With the following procedure, you create the IAM role, required IAM policies, and the S3 location using the AWS CLI.

You can also create an S3 location using the console. For more information, see Creating a Location for Amazon S3 (p. 70).

To create an S3 location using the CLI

1. Create an IAM trust policy that allows DataSync to assume the IAM role required to access your S3 bucket.

   The following is an example of a trust policy.

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

2. Create a temporary file for the IAM policy, as shown in the following example.

   ```bash
   $ ROLE_FILE=$(mktemp -t sync.iam.role.XXXXXX.json)
   $ IAM_ROLE_NAME="YourBucketAccessRole"
   $ cat<<EOF> ${ROLE_FILE}
   {
   "Version": "2012-10-17",
   "Statement": [
   {
       "Effect": "Allow",
       "Principal": {
           "Service": "datasync.amazonaws.com"
       },
       "Action": "sts:AssumeRole"
   }
   ]
   }
   EOF
   ```

3. Create an IAM role and attach the IAM policy to it.

   The following command creates an IAM role and attaches the policy to it.

   ```bash
   $ aws iam create-role --role-name ${IAM_ROLE_NAME} --assume-role-policy-document file:///${ROLE_FILE}
   ```
4. Allow the IAM role that you created to write to your S3 bucket.

Attach an IAM policy that has sufficient permissions to access your S3 bucket to the IAM role (for example, AmazonS3FullAccess). You can also create a policy that is more restrictive. If you do, the minimal permissions needed for DataSync to read and write to an S3 location are shown the following example.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:HeadBucket"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::YourBucket"
        },
        {
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:DeleteObject",
                "s3:GetObject",
                "s3:ListMultipartUploadParts",
                "s3:PutObject"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::YourBucket/*"
        }
    ]
}
```

To attach the policy to your IAM role, run the following command.

```
$ aws iam attach-role-policy --role-name role-name --policy-arn 'arn:aws:iam::aws:policy/AmazonS3FullAccess'
```

5. Create the S3 location.

Use the following commands to create your Amazon S3 location.
Step 3: Create a Task

After you have created an agent and configured your source and destination, you create a task, as described following.

To create a task using the CLI

1. Create a CloudWatch Logs group using the following command.

   ```bash
   # aws logs create-log-group --log-group-name your-log-group
   ```

2. Attach an IAM resource policy to your CloudWatch Logs group. For instructions on how to attach the policy, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).

3. Create a task by using the following command.

   ```bash
   ```

   This command returns the Amazon Resource Name (ARN) for a task, similar to the one shown following.

   ```json
   { "TaskArn": "arn:aws:datasync:us-east-1:11122233444:task/task-08de6e667796f026" }
   ```

Your task is created with the default configuration options. If you want to configure different options as part of your task creation, add the `--options` parameter to your `create-task` command. The following example shows how to specify different options. For a description of these options, see the section called "Options" (p. 222).

```bash
# aws datasync create-task --source-location-arn 'arn:aws:datasync:region:account-id:location/location-id' --destination-location-arn 'arn:aws:datasync:region:account-id:location/location-id' --options 'key1=value1 key2=value2'
```
Step 4: Start a Task Execution

When a task execution starts, the task execution changes from LAUNCHING to PREPARING status within about 10 minutes. The time that the task execution takes to move through its other phases is proportional to the size of your volume. For information about task execution phases, see Task Execution (p. 6).

Use the following command to start a task execution.

```
$ aws datasync start-task-execution --task-arn 'arn:aws:datasync:region:account-id:task/task-id'
```

The command returns a task execution Amazon Resource Name (ARN) similar to the one shown following.

```
{ "TaskExecutionArn": "arn:aws:datasync:us-east-1:209870788375:task/task-08de6e6697796f026/exec/04ce9d516d69bd52f" }
```

You can override the task's options by specifying different options for the current execution, as shown in the example following. For a description of these options, see the section called “Options” (p. 222).

```
$ aws datasync start-task-execution [...] --override-options --VerifyMode=NONE,OverwriteMode=NEVER,PosixPermissions=NONE
```

When you run a task, you can optionally configure the task to include specific files, folders, and objects to transfer. For more information, see Filtering the Data Transferred by AWS DataSync (p. 58).

**Note**
Each agent can run a single task at a time.

Step 5: Use the CLI to Monitor Your Task Execution

Use the CLI to monitor the status of your task execution by using the describe-task-execution command.

```
$ aws datasync describe-task-execution --task-execution-arn 'arn:aws:datasync:region:account-id:task/task-id-execution/task-execution-id'
```
Monitor Your Task Execution in Real Time

This command returns information about a task execution similar to that shown following.

```json
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-1:11223445566:task/task-08de6e697796f026/execution/exec-04ce9d516d69bd52f",
    "Status": "VERIFYING",
    "Options": {
        "VerifyMode": "POINT_IN_TIME_CONSISTENT",
        "Atime": "BEST_EFFORT",
        "Mtime": "PRESERVE",
        "Uid": "INT_VALUE",
        "Gid": "INT_VALUE",
        "PreserveDevices": "NONE",
        "PosixPermissions": "PRESERVE",
        "PreserveDeletedFiles": "PRESERVE",
        "OverWriteMode": "NEVER",
        "TaskQueueing": "ENABLED"
    },
    "StartTime": 1532658526.949,
    "EstimatedFilesToTransfer": 0,
    "EstimatedBytesToTransfer": 0,
    "FilesTransferred": 0,
    "BytesWritten": 0,
    "BytesTransferred": 0,
    "Result": {
        "PrepareDuration": 4355,
        "PrepareStatus": "Ok",
        "TransferDuration": 5889,
        "TransferStatus": "Ok",
        "VerifyDuration": 4538,
        "VerifyStatus": "Pending"
    }
}
```

If the task execution succeeds, the value of `Status` changes to `SUCCESS`. If the `describe-task-execution` command fails, the result sends error codes that can help you troubleshoot issues. For information about the error codes, see the section called “TaskExecutionResultDetail” (p. 232) in the DataSync API Reference.

Monitor Your Task Execution in Real Time

To monitor the progress of your task execution in real time from the command line, you can use the standard Unix `watch` utility. The watch utility doesn't recognize the DataSync alias, so invoke the CLI directly as shown in the example following.

```
# pass '-n 1' to update every second and '-d' to highlight differences
# watch -n 1 -d "aws datasync describe-task-execution --task-execution-arn 'arn:aws:datasync:region:account-id:task/task-id/execution/task execution-id'"
```


Monitoring Your Task

You can monitor AWS DataSync using Amazon CloudWatch, which collects and processes raw data from DataSync into readable, near real-time metrics. These statistics are retained for a period of 15 months, so that you can access historical information and gain a better perspective on how DataSync. By default, DataSync metric data is automatically sent to CloudWatch at 5-minute periods. For more information about CloudWatch, see What Are Amazon CloudWatch, CloudWatch Events, and CloudWatch Logs? in the Amazon CloudWatch User Guide.

Accessing Amazon CloudWatch Metrics for DataSync

Amazon CloudWatch provides metrics that you can use to get information about DataSync performance. You can see CloudWatch metrics for DataSync in many ways. You can view them through the CloudWatch console, or you can access them using the CloudWatch CLI or the CloudWatch API. You can also see these metrics on the task execution details page in the AWS DataSync console. For information about how to use CloudWatch metrics, see Using Amazon CloudWatch Metrics in the Amazon CloudWatch User Guide.

DataSync CloudWatch Metrics

The AWS/DataSync namespace includes the following metrics.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BytesVerifiedSource</td>
<td>The total number of bytes of data that are verified at the source location.</td>
</tr>
<tr>
<td></td>
<td>Units: Bytes</td>
</tr>
<tr>
<td>BytesPreparedSource</td>
<td>The total number of bytes of data that are prepared at the source location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Bytes</td>
</tr>
<tr>
<td>FilesVerifiedSource</td>
<td>The total number of files that are verified at the source location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Count</td>
</tr>
<tr>
<td>FilesPreparedSource</td>
<td>The total number of files that are prepared at the source location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Count</td>
</tr>
<tr>
<td>Metric</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BytesVerifiedDestination</td>
<td>The total number of bytes of data that are verified at the destination location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Bytes</td>
</tr>
<tr>
<td>BytesPreparedDestination</td>
<td>The total number of bytes of data that are prepared at the destination location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Bytes</td>
</tr>
<tr>
<td>FilesVerifiedDestination</td>
<td>The total number of files that are verified at the destination location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Count</td>
</tr>
<tr>
<td>FilesPreparedDestination</td>
<td>The total number of files that are prepared at the destination location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Count</td>
</tr>
<tr>
<td>FilesTransferred</td>
<td>The actual number of files or metadata that were transferred over the network. This value is calculated and updated on an ongoing basis during the TRANSFERRING phase. It's updated periodically when each file is read from the source location and sent over the network.</td>
</tr>
<tr>
<td></td>
<td>If failures occur during a transfer, this value can be less than EstimatedFilesToTransfer. This value can also be greater than EstimatedFilesTransferred in some cases. This element is implementation-specific for some location types, so don't use it as an indicator for a correct file number or to monitor your task execution.</td>
</tr>
<tr>
<td></td>
<td>Unit: Count</td>
</tr>
<tr>
<td>BytesTransferred</td>
<td>The total number of bytes that are transferred over the network when the agent reads from the source location to the destination location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Bytes</td>
</tr>
<tr>
<td>BytesWritten</td>
<td>The total logical size of all files that have been transferred to the destination location.</td>
</tr>
<tr>
<td></td>
<td>Unit: Bytes</td>
</tr>
</tbody>
</table>

### CloudWatch Events for DataSync

CloudWatch events describe changes in DataSync resources. You can set up rules to match these events and route them to one or more target functions or streams.

The following CloudWatch events are available for AWS DataSync.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>State Changes for an Agent</td>
<td>For details, see Understanding Agent Statuses (p. 83).</td>
</tr>
<tr>
<td>ONLINE</td>
<td>The agent is configured properly and is available to use.</td>
</tr>
<tr>
<td>OFFLINE</td>
<td>The agent's VM is turned off or the agent is in an unhealthy state.</td>
</tr>
</tbody>
</table>
### DataSync Dimensions

DataSync metrics use the AWS/DataSync namespace and provide metrics for the following dimensions:

- **AgentId**—the unique ID of the agent.
- **TaskId**—the unique ID of the task. It takes the form `task-01234567890abcdef`.

### Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups

You can use CloudWatch Log Groups to monitor and debug your tasks. For DataSync to upload logs to your CloudWatch Log Group, DataSync requires a resource policy that grants sufficient permissions.

The following is an example of the simplest policy that grants such permissions.

```json
{
}
```
To apply the policy, save this policy statement to a file on your local computer. Then run the following AWS CLI command to apply the resource policy:

```bash
aws logs put-resource-policy --policy-name trustDataSync --policy-document file:///full-path-to-policy-file
```

**Note**

Run this command using the same AWS account and region that your DataSync agent is activated in.


```
"Statement": [
    {
        "Sid": "DataSyncLogsToCloudWatchLogs",
        "Effect": "Allow",
        "Action": [
            "logs:PutLogEvents",
            "logs:CreateLogStream"
        ],
        "Principal": {
            "Service": "datasync.amazonaws.com"
        },
        "Resource": "*"
    }
],
"Version": "2012-10-17"
```
Working with Tasks

In this section, you can find information about how to work with tasks.

A task is a set of two locations (source and destination) and a set of default options that you use to control the behavior of the task.

Topics
- Understanding Task Creation Statuses (p. 49)
- Understanding Task Execution Statuses (p. 49)
- Creating a Task (p. 50)
- Starting a Task (p. 57)
- Filtering the Data Transferred by AWS DataSync (p. 58)
- Scheduling Your Task (p. 61)
- Deleting a Task (p. 63)

Understanding Task Creation Statuses

Following, you can find information about the possible statuses (phases) a task might go through when it is created.

<table>
<thead>
<tr>
<th>Task Creation Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATING</td>
<td>AWS DataSync attempts to mount the Network File System (NFS) or Server Message Block (SMB) location. The task transitions to the AVAILABLE status without waiting for the AWS location to mount. Instead, AWS DataSync mounts an AWS location before every task execution and then unmounts it after every task execution. If the task remains in the CREATING status for more than a few minutes, your agent might be having trouble mounting the NFS or SMB file system. Check the task's ErrorCode and ErrorDetail values. Mount issues are often caused by either a misconfigured firewall or a mistyped NFS or SMB server host name.</td>
</tr>
<tr>
<td>RUNNING</td>
<td>The task is running.</td>
</tr>
<tr>
<td>AVAILABLE</td>
<td>The task is configured properly and is available to be started.</td>
</tr>
<tr>
<td>UNAVAILABLE</td>
<td>If an agent that is associated with a location goes offline, the task transitions to the UNAVAILABLE status.</td>
</tr>
<tr>
<td>QUEUED</td>
<td>There is another task running and it is using the same agent. DataSync runs the task in the queue in series. For more information, see Queueing Task Executions (p. 57).</td>
</tr>
</tbody>
</table>

Understanding Task Execution Statuses

Following, you can find information about the possible statuses (phases) a task execution might go through.
## DataSync Phase or Status

<table>
<thead>
<tr>
<th>Phase or Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUEUEING</td>
<td>This is the first phase of a task execution if there is another task running and it is using the same agent. For more information, see Queueing Task Executions (p. 57).</td>
</tr>
<tr>
<td>LAUNCHING</td>
<td>This is the first phase of a task execution if there is no other task running and using the same agent or queueing isn't enabled. At this point, AWS DataSync is initializing the task execution. This status usually goes quickly, but can take up to a few minutes.</td>
</tr>
<tr>
<td>PREPARING</td>
<td>This is the second phase of a task execution. AWS DataSync is computing which files need to be transferred. The time that this phase takes is proportional to the number of files in the source location. It usually takes between a few minutes to a few hours depending on both the source and destination file systems and the performance of these file systems. For more information, see Starting a Task (p. 57).</td>
</tr>
<tr>
<td>TRANSFERRING</td>
<td>This is the third phase of a task execution. DataSync is performing the actual transfer of your data to AWS. While the DataSync is transferring files, the number of bytes and files that is transferred is updated in real time.</td>
</tr>
<tr>
<td>VERIFYING</td>
<td>This is the fourth and optional phase of a task execution. If the VerifyMode sync option is set to POINT_IN_TIME_CONSISTENT, DataSync performs a full data and metadata integrity verification. This verification ensures that the data in your destination is an exact copy of the data in your source location. This process requires reading back all files in the destination location and can take a significant amount of time on very large volumes. If you want to skip verification, you specify VerifyMode=NONE when configuring the task execution. Alternatively, in your task's options in the console, don't choose Enable verification. For more information, see How AWS DataSync Verifies Data Integrity (p. 7).</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>This value is returned if the data transfer is successful. If the VerifyMode option isn’t set, this status occurs after the TRANSFERRING phase. Otherwise, it occurs after the VERIFYING phase. For more information, see Task Execution (p. 6).</td>
</tr>
<tr>
<td>ERROR</td>
<td>This value is returned if the data transfer fails. If the VerifyMode option isn’t set, this status occurs after the TRANSFERRING phase. Otherwise, it occurs after the VERIFYING phase.</td>
</tr>
</tbody>
</table>

## Creating a Task

If this is your first time using DataSync, the instructions in Getting Started with AWS DataSync (p. 20) walk you through the process of creating a task.

You can create tasks that transfer from different source and destination location types. For a list of all DataSync supported source and destination endpoints, see Working with Locations (p. 64).

- Creating a Location for NFS (p. 65)
- Creating a Location for SMB (p. 66)
- Creating a Location for Amazon EFS (p. 67)
- Creating a Location for Amazon FSx for Windows File Server (p. 69)
When you initially create a task, it enters the **CREATING** status. During the **CREATING** status, AWS DataSync attempts to mount the NFS or SMB location. The task transitions to the **AVAILABLE** status without waiting for the AWS location to become available. If necessary, AWS DataSync mounts the AWS location before every task execution and then unmounts it after every task execution. If an agent that is associated with an NFS or SMB location goes offline, the task transitions to the **UNAVAILABLE** status.

If the status of the task remains in the **CREATING** status for more than a few minutes, it means that your agent might be having trouble mounting the source NFS or SMB file system. You can check the task's `ErrorCode` and `ErrorDetail` values in the API Reference section in this guide. Mount issues are often caused by either a misconfigured firewall or a mistyped NFS or SMB server hostname. For troubleshooting information, see Troubleshooting AWS DataSync Issues (p. 113).

### Topics
- Creating a Task to Transfer Data Between On-Premises NFS or SMB and AWS (p. 51)
- Creating a Task to Transfer Between In-Cloud Locations (p. 51)
- Configuring Task Settings (p. 55)

### Creating a Task to Transfer Data Between On-Premises NFS or SMB and AWS

If you have previously created a task and want to create additional tasks, use the following procedure.

**To create a task**

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. On the navigation pane, choose **Tasks**, and then choose **Create task**.
3. On the **Configure source location** page, choose **Create new location** and configure a new location if you want to use a new location for your source. Provide the configuration settings and choose **Next**. For instructions on how to create a location, see Working with Locations (p. 64).
   
   If you want to use a source location that you previously created, choose **Choose existing location**, choose your source location from the list, and then choose **Next**.

   For step-by-step instruction, see Configure a Source Location (p. 27).

### Creating a Task to Transfer Between In-Cloud Locations

Use the following instructions to set up the DataSync Amazon EC2 agent for data transfers. The examples in this section cover these use cases:

- **Transfer Data from In-Cloud NFS to In-Cloud NFS or S3 (p. 4)** – Transfer data from Amazon EFS to Amazon EFS, from self-managed NFS to EFS, or to Amazon S3.
- **Transfer from S3 to In-Cloud NFS (p. 5)** – Transfer data from Amazon S3 to Amazon EFS, or from Amazon S3 to self-managed NFS.
Creating a Task to Transfer from In-Cloud NFS to In-Cloud NFS or S3

Use the following instructions to transfer data from an in-cloud NFS file system to AWS. To perform this transfer, the DataSync agent must be located in the same AWS Region and same AWS account where the file system is deployed. This type of transfer includes transfers from EFS to EFS, transfers from self-managed NFS to Amazon EFS, and transfers to Amazon S3. For information about how in-cloud NFS to in-cloud NFS or S3 works, see Transfer Data from In-Cloud NFS to In-Cloud NFS or S3 (p. 4).

**Note**
Deploy the agent in the AWS Region and AWS account where the source EFS or self-managed NFS file system resides.

Deploying Your DataSync Agent as an EC2 Instance to Read Files from In-Cloud

To deploy the DataSync agent as an EC2 instance

1. From the AWS account where the source EFS resides, launch the agent using your Amazon Machine Image (AMI) from the Amazon EC2 launch wizard. Use the following URL to launch the AMI.

   ```plaintext
   ```

   In the URL, replace the `source-efs-or-nfs-region` and `ami-id` with your own.

   After the AMI launches, the **Choose an Instance Type** appears on the Amazon EC2 console. For a list of AMI IDs by AWS Region, see Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21).

2. Choose one of the recommended instance types for your use case, and choose **Next: Configure Instance Details**. For the recommended instance types, see Amazon EC2 Instance Requirements (p. 10).

3. On the **Configure Instance Details** page, do the following:
   a. For **Network**, choose the VPC where your source EFS or NFS is located.
   b. Choose a value for **Auto-assign Public IP**. If you want your instance to be accessible from the public internet, set **Auto-assign Public IP** to **Enable**. Otherwise, set **Auto-assign Public IP** to **Disable**. If a public IP address isn't assigned, activate the agent in your VPC using its private IP address.

   When you transfer files from an in-cloud NFS, to increase performance, we recommend that you choose the **Placement Group** where your NFS server resides.

4. Choose **Next: Add Storage**. The agent doesn't require additional storage, so you can skip this step and choose **Next: Add tags**.

5. (Optional) On the **Add Tags** page, you can add tags to your EC2 instance. When you're finished on the page, choose **Next: Configure Security Group**.

6. On the **Configure Security Group** page, do the following:
   a. Make sure that the selected security group allows inbound access to HTTP port 80 from the web browser that you plan to use to activate the agent.
   b. Make sure that the security group of source EFS or NFS allows inbound traffic from the agent. In addition, make sure that the agent allows outbound traffic to the source EFS or NFS. The traffic goes through the standard NFS port, 2049.

For the complete set of network requirements for DataSync, see Network Requirements for DataSync (p. 10).
7. Choose **Review and Launch** to review your configuration, then choose **Launch** to launch your instance. Remember to use a key pair that's accessible to you. A confirmation page appears and indicates that your instance is launching.

8. Choose **View Instances** to close the confirmation page and return to the EC2 instances screen. When you launch an instance, its initial state is **pending**. After the instance starts, its state changes to **running**. At this point, it is assigned a public Domain Name System (DNS) name and IP address, which can be found in the **Descriptions** tab.

9. If you set **Auto-assign Public IP** to **Enable**, choose your instance and note the public IP address in the **Description** tab. You use this IP address later to connect to your sync agent.

   If you set **Auto-assign Public IP** to **Disable**, launch or use an existing instance in your VPC to activate the agent. In this case, you use the private IP address of the sync agent to activate the agent from this instance in the VPC.

---

**Creating a Task to Transfer Data from EFS or Self-Managed NFS or SMB**

Next, you create a task to transfer data.

**Note**
Create the task in the AWS Region and AWS account where the destination EFS or S3 bucket resides.

**To create a task**

1. Open the DataSync console in the AWS Region where your destination Amazon EFS file system is located. The destination EFS or S3 bucket must be in the same AWS account.

2. Choose **Create task**, then choose **On-premises to AWS** on the **Use case options** page, and then choose **Create agent**.

3. In the Create agent wizard's **Activation** section, enter the EC2 instance's IP address for **Agent address**, and then choose **Get key**. This IP address can be private or public. For more details, see step 9 of To deploy the DataSync agent as an EC2 instance (p. 52).

   Your browser connects to this IP address to get a unique activation key from your agent. This key securely associates your agent with your AWS account. This IP address doesn't need to be accessible from outside your network, but must be accessible from your browser.

4. Enter an agent name that you can easily identify later, and choose **Create agent** when done. You can optionally add tags to the agent.

5. Choose **Tasks** from the navigation pane.

6. Choose **On-premises to AWS**, and choose **Next** to open the **Source configuration** page.

7. In the **Source location options**, choose **Create new location** and choose **Network File System (NFS) or Server Message Block (SMB)**. Fill in the following options:

   - For agent, choose your newly created agent from the list.
   - If you are copying from EFS, do the following:
     - For **NFS Server**, enter the **DNS name** of your source EFS.
     - For **Mount path**, enter `/ (backslash) and choose **Next**.
   - If you are copying from self-managed NFS or SMB, do the following:
     - For **NFS Server**, enter the private DNS or IP address of your source NFS.
     - For **Mount path**, enter a path that is exported by your NFS server and choose **Next**. For more information, see Create an NFS Location (p. 36).

8. Choose **Create new location**. This is the destination location for your data transfer. Fill in the following options:

   - If you are copying to EFS, do the following:
Creating a Task to Transfer Between In-Cloud Locations

- For **Location type**, choose EFS.
- Choose your destination EFS.
- For **Mount path**, enter / (backslash).
- For **Subnet** and **Security groups**, use the default settings and choose **Next**.
- If you are copying to S3, do the following:
  - For **Location type**, choose Amazon S3 bucket.
  - For **S3 bucket**, choose your source S3 bucket.
  - For **Folder**, choose a folder prefix to use for the transfer, or you can keep it blank.
  - Choose your destination S3 bucket and an optional folder. DataSync can autogenerate an AWS Identity and Access Management (IAM) role to access your bucket, or you can create on your own.

9. Choose **Next**, and optionally name the task and add tags.
10. Choose or create an Amazon CloudWatch Logs log group at the bottom of the page, and choose **Next**. For more information on working with CloudWatch Logs, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).
11. Review the settings on the next page, and choose **Create task**.
12. Choose **Start** to run the task that you just created to start transferring data.

Creating a Task to Transfer from S3 to In-Cloud NFS

Use the following instructions to transfer data from S3 to an in-cloud NFS file system that is located in the same AWS account and AWS Region where the agent is deployed. This approach includes transfers from S3 to EFS, or from S3 to self-managed NFS. The following diagram illustrates this type of transfer. For information about how S3 to in-cloud NFS works, see Transfer from S3 to In-Cloud NFS (p. 5).

Deploying the DataSync EC2 Agent to Write to your Destination Location

First, deploy the DataSync EC2 agent in the AWS Region and AWS account where the destination EFS or self-managed NFS resides.

**To deploy the agent**

- Launch the agent from the selected AMI by using the EC2 launch wizard. To do so, use the following URL.

  https://console.aws.amazon.com/ec2/v2/home?region=DESTINATION-EFS-or-NFS-REGION#LaunchInstanceWizard:ami=AMI-ID.

  In the URL, replace the AWS Region and AMI ID with your own. You are redirected to the **Choose an Instance Type** page on the EC2 console. For a list of AMI IDs by AWS Region, see Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21).

Creating a Task to Transfer Data from Amazon S3

Next, you create a task to transfer data.

**Note**  
Create the task in the AWS account and AWS Region where the source S3 bucket resides.

**To create a task that transfers data from S3 to EFS or a self-managed NFS or SMB**

1. Open the DataSync console in the AWS Region where your source S3 bucket is located.
2. Choose **Create task**, and choose the use case **AWS to on-premises**.
3. Choose Create agent.
4. If you set Auto-assign Public IP to Enable, choose your instance and note the public IP address in the Description tab. You use this IP address later to connect to your sync agent.

   If you set Auto-assign Public IP to Disable, launch or use an existing instance in your VPC to activate the agent. In this case, you use the private IP address of the sync agent to activate the agent from this instance in the VPC.
5. In the Create Agent wizard, for Agent address enter the EC2 instance's IP address (private or public, as explained in step 3), and then choose Get key.

   Your browser connects to this IP address to get a unique activation key from your agent. This key securely associates your agent with your AWS account. This IP address doesn't need to be accessible from outside your network, but must be accessible from your browser.
6. Choose an agent name that you can easily identify later. You can optionally add tags. When you're done, choose Create agent.
7. Choose AWS to on-premises, and choose Next.
8. Choose Create new location:
   - For Location type, choose Amazon S3 bucket.
   - For S3 bucket, choose your source S3 bucket.
   - For Folder, choose a folder prefix for the transfer, or you can keep it blank.

   DataSync can autogenerate an IAM role to access your bucket, or you can create on your own.
9. Choose Next. Choose Create new location, choose NFS or SMB for Location type, and choose the agent that you just created from the list.
10. a. If you are copying to EFS, do the following:
    - For NFS Server, enter the DNS name of your source EFS.
    - For Mount path, enter / (backslash) and choose Next.
    b. If you are copying to in-cloud NFS, do the following:
    - For NFS Server, enter the private DNS or IP address of your source NFS.
    - For Mount path, enter a path that is exported by your NFS server. For more information, see Create an NFS Location (p. 36).
11. Choose Next, and optionally name the task and add tags.
12. Choose or create a CloudWatch Logs log group at the bottom of the page, and choose Next. For more information on working with CloudWatch Logs, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).
13. Review the settings on the next page, and choose Create task.
14. Choose Start to run the task that you just created to transfer data, and then choose Start again on the Start Task page.

Configuring Task Settings

Following, you can find information on how to configure a task setting. You use these settings to control how a task execution behaves. These settings are available in the Options section.

These options control the behavior of a task execution. Behavior includes preserving metadata such as the user ID (UID) or group ID (GID), preserving file permissions, and data integrity verification. If you don't specify values for these options, DataSync uses a set of default options that can be overridden for a task execution.

Available options are as follows:
• **Verify data** – Set an optional verification to compare source and destination at the end of the transfer. As DataSync transfers data, it always performs data integrity checks during the transfer. You can enable additional verification to compare source and destination at the end of a transfer. As part of that, you can check either the entire dataset or just the files or objects that DataSync transferred.

Data verification options include the following:

- **Verify only the data transferred (recommended)** calculates the checksum of transferred files and metadata on the source. It then compares this checksum to the checksum calculated on those files at the destination at the end of the transfer. We recommend this option when transferring to S3 Glacier or S3 Glacier Deep Archive storage classes. For more information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

- **Verify all data in the destination** performs a scan at the end of the transfer of the entire source and entire destination to verify that source and destination are fully synchronized. This option isn't supported when transferring to S3 Glacier or S3 Glacier Deep Archive storage classes. For more information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

- **Check integrity during the transfer** doesn't run additional verification at the end of the transfer. All data transmissions are still integrity-checked with checksum verification during the transfer.

- **Copy ownership** – If you choose this option, DataSync copies file ownership such as the group ID of the file's owners and the user ID of the file's owner.

- **Copy permissions** – If you choose this option, DataSync copies file POSIX permissions from the source to the destination.

- **Copy timestamps** – If you choose this option, DataSync copies the timestamp metadata from the source to the destination.

- **Keep deleted files** – If you choose this option, DataSync keeps files in the destination that don't exist in the source file system.

If your task deletes objects, you might incur minimum storage duration charges for certain storage classes. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

- **Overwrite files** – If you choose this option, files at the destination are overwritten by files from the source. If you don't choose this option, the destination file isn't replaced by the source file, even if the destination file differs from the source file.

If your task overwrites objects, you might incur minimum storage duration charges for certain storage classes. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

- **Use available** or **Set bandwidth limit (MiB/s)** – If you choose Use available, DataSync uses all the network bandwidth that is available for the transfer. If you choose Set bandwidth limit (MiB/s), you limit the maximum bandwidth that you want DataSync to use for this task.

- **Queueing** – If you use a single agent to run multiple tasks, choose this option to make the tasks run in series (that is, first in, first out). For more information, see Queueing Task Executions (p. 57).

In the Filtering configuration - Optional section, enter a pattern to use as a filter. This pattern defines the criteria for specific files, folders, and objects to exclude from your transfer. You can add more patterns later by editing the task configuration. For more information, see Excluding Data from a Transfer (p. 59). You can include files, folders, and objects in the transfer when you start a task. For more information, see Start Your Task (p. 32).

**Note**

To use a pipe in your pattern, you must escape it. For examples, see Filtering the Data Transferred by AWS DataSync (p. 58).

- In the **Schedule (optional)** section, configure your task to run on a schedule that you specify.

  For **Frequency**, configure how frequently you want the task to run. For frequency configuration options, see Configuring a Task Schedule (p. 62).
• In the **Tags - optional** section, enter **Key** and **Value** to tag your task. A **tag** is a key-value pair that helps you manage, filter, and search for your tasks. We recommend that you create a name tag for your task.

• **Task logging - optional** – If you choose this option, DataSync publishes logs for individual files or objects to the CloudWatch log group that you specify.

To upload logs to your CloudWatch log group, DataSync requires a resource policy that grants sufficient permissions. For an example of such a policy, see [Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups](p. 47).

For more information about using log groups and streams, see [Working with Log Groups and Log Streams](in the Amazon CloudWatch Logs User Guide).

Use the **Log level** option to set the level of detail that is logged to CloudWatch Logs. Log level options include the following:

• **Log basic information such as transfer errors** – If you choose this option, only basic information such as transfer errors is published to CloudWatch.

• **Log all transferred objects, files, and folders** – If you choose this option, log records are published to CloudWatch Logs for all files or objects that the task copies and integrity-checks.

• **Do not send logs to CloudWatch** – If you choose this option, DataSync logs are not published to CloudWatch.

## Starting a Task

Starting a task creates a task execution. A **task execution** is an individual run of a task. Each task can have at most one task execution at a time. You can run a task with the DataSync options already configured on the task level when creating it. Alternatively, you can change the options for a specific task run and execution before you run the task. For instructions on how to start a task, see [Start Your Task](p. 32).

**Note**

Each agent can execute a single task at a time.

The time that AWS DataSync spends in the **PREPARING** status depends on the number of files in both the source and destination file systems. It also depends on the performance of these file systems. When a task starts, DataSync performs a recursive directory listing to discover all files and file metadata in the source and destination file system. These listings are used to identify differences and determine what to copy, and usually takes between a few minutes to a few hours.

## Queueing Task Executions

When you use the same agent to run multiple tasks, you can queue one task execution for each task. By using queueing, you can make tasks run in series (first in, first out) even if the agent is already running other tasks. You can set queueing either by using the DataSync console or the API.

To enable queueing on the DataSync console, choose **Enabled** for **Queueing** for the option when you configure task settings. If you enable queueing and the agent is running an execution from another task, the current task’s execution is automatically queued. After a task execution is completed, the agent runs the next queued execution. If you want to remove a task execution from the queue yourself, simply cancel the execution.

To enable queueing by using the DataSync API, set the **TaskQueueing** property to **ENABLED**.
Filtering the Data Transferred by AWS DataSync

When you transfer data from your source to your destination location, you can apply filters to transfer only a subset of the files in your source location. For example, if your source location includes temporary files that end with `.tmp`, you can create an exclude filter that ensures that these files are not transferred to the destination.

Filtering is an optional configuration to specify if you want to transfer a subset of your source files. You can leave the filter configuration empty if you want all files from the source to be transferred to the destination location.

Topics
- Filtering Terms, Definitions, and Syntax (p. 58)
- Excluding Data from a Transfer (p. 59)
- Including Data in a Transfer (p. 60)
- Sample Filters for Common Uses (p. 60)

Filtering Terms, Definitions, and Syntax

Following are some terms and definitions for use with filtering:

**Filter**

The whole string that makes up a particular filter, for example: `*.tmp | *.temp`

Filters are made up of patterns that are delimited with a `|` (pipe). A delimiter is not needed when you add patterns on the console because each pattern is added separately.

**Pattern**

A pattern within a filter. For example, `*.tmp` is a pattern that is part of the `*.tmp | *.temp` filter.

**Folders**

- All filters are relative to the source location path. For example, suppose that you specify `/my_source/` as the source path when you create the task and create the include filter `/transfer_this/`. In this case, only the directory `/my_source/transfer_this/` and its content are transferred.
- To specify a folder directly under the source location, include a backslash (`/`) in front of the folder name. In the example preceding, the pattern uses `/transfer_this`, not `transfer_this`.
- The following patterns are interpreted the same way and match both the folder and its content.

  ```
  /dir
  /dir/
  ```

- When you are transferring data from or to an Amazon S3 bucket, DataSync treats the `/` character in the object key as the equivalent of a folder on a file system.

**Special Characters**

Following are special characters for use with filtering.

<table>
<thead>
<tr>
<th>Special Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>*</code> (wildcard)</td>
<td>A character used to match zero or more characters. For example, <code>/movies_folder*</code> matches both <code>/movies_folder</code> and <code>/movies_folder1</code>.</td>
</tr>
</tbody>
</table>
### Special Character Description

<table>
<thead>
<tr>
<th>Special Character</th>
<th>Description</th>
</tr>
</thead>
</table>
| | (pipe delimiter) | A character used as a delimiter between patterns. It enables specifying multiple patterns, any of which can match the filter. For example, *.*tmp | *.*temp matches files ending with either tmp or temp.  
**Note**  
This delimiter is not needed when you add patterns on the console because each pattern is added on a separate line. |
| \ (backslash) | This character is used for escaping when a file or object names contains special characters (*, |, \).  
A double backslash (\\) is required when a backslash is part of a file name. Similarly, \\\ represents two consecutive backslashes in a file name.  
A backslash followed by a pipe (\|) is required when a pipe is part of a file name.  
\ followed by any other character, or at the end of a pattern, is ignored. |

## Excluding Data from a Transfer

*Exclude filters* define files, folders, and objects that are excluded when you transfer files from a source to a destination location. You can configure these filters when you create or edit a task.

To create a task with an exclude filter in the DataSync console, specify a list of patterns in the Filtering configuration – (optional) section in the Exclude patterns box. For example, to exclude temporary folders, you can specify */temp in the exclude patterns text box, choose Add patterns and then specify */tmp in the second text box. To add more patterns to the filter, choose Add pattern. When you are using the CLI, note that quotes are required around the filter and a | (pipe) is used as a delimiter. For this example, you would specify '**/temp|*/tmp'.

The following screenshot shows the Edit Task wizard with patterns that exclude temporary folders.

![Edit Task wizard with patterns that exclude temporary folders](image)

After you have created a task, you can edit the task configuration to add or remove patterns from the filter.

You can also use the AWS Command Line Interface (AWS CLI) to create an exclude filter. The following example shows such a CLI command.

```
aws datasync create-task
```
Including Data in a Transfer

Include filters define files, folders, and objects that are transferred when you run a task. You configure these filters as part of the configuration when you start a task.

To start a task with an include filter, specify a list of patterns to be included in the optional configuration when you start a task. To do this, use the Start with Overrides option in the DataSync console.

Files and folders matching the include filters are the only ones that are transferred. For example, to include only a subset of your source folders, you might specify /important_folder_1/important_folder_2.

You can also use the AWS CLI to create an include filter. The following example shows the CLI command. Take note of the quotes around the filter and the | (pipe) that is used as a delimiter.

```bash
aws datasync start-task-execution
  --task-arn 'arn:aws:datasync:region:account-id:task/task-id'
  --includes FilterType=SIMPLE_PATTERN,Value='/important_folder1|/important_folder2'
```

Note
Currently, include filters support the * character only as the rightmost character in a pattern. For example, /documents*/code* is supported but *.txt is not supported.

Sample Filters for Common Uses

Exclude some folders from your source location

In some cases, you might need to exclude folders in your source location to keep them from being copied to your destination. For example, you might have temporary work-in-progress folders. Or you might use NetApp and want to exclude NetApp backup folders. In these cases, you use the following filter.

*/*.snapshot

To exclude folders at any level in the file hierarchy, you can create a task to configure an exclude filter like the following.

*/folder-to-exclude-1|*/folder-to-exclude-2

To exclude folders at the top level of the source location, you can create a task to configure an exclude filter like the following.

/top-level-folder-to-exclude-1|/top-level-folder-to-exclude-2

Include only a subset of the folders on your source location
In some cases, your source location might be a large share, and you need to transfer only a subset of the folders under the root. To include specific folders, start a task execution with an include filter like the following.

/folder-to-transfer

**Exclude specific file types**

To exclude certain file types from the transfer, you can create a task execution with an exclude filter such as `.temp`.

**Transfer only individual files you specify**

To transfer a list of individual files, start a task execution with an include filter like the following: 

"/folder/subfolder/file1.txt[/folder/subfolder/file2.txt[/folder/subfolder/file2.txt"

**Note**

The string length is limited to 100,000 characters.

---

**Scheduling Your Task**

Using task scheduling in AWS DataSync, you can periodically execute a transfer task from your source storage system to the destination. A scheduled task automatically runs at a frequency that you configure. For example, the following screenshot shows a configuration that runs a task every Sunday and Wednesday at 12:00 PM UTC.

With DataSync, you can also execute a task schedule based on a cron expression. A cron expression should be specified in UTC time. For example, you can configure a task to run on every Sunday and Wednesday at 12:00 PM by using the following cron expression.

```
0 12 ? * SUN,WED *
```
Configuring a Task Schedule

You can configure the frequency of the task execution by using the DataSync console or API. When you create or edit a task, the following options are available for Frequency in the console:

- Choose Not Scheduled if you don’t want to schedule your task to run periodically.
- Choose Hourly and choose the minute in the hour that the task should run. The task runs every hour on the specified minute.
- Choose Daily and enter the UTC time that you want the task to run, in the format HH:MM. This task runs every day at the specified time.
- Choose Weekly and the day of the week and enter the UTC time the task should run, in the format HH:MM. This task runs every week on the specified day at the specified time.
- Choose Days of the week, choose the specific day or days, and enter the UTC time that the task should run in the format HH:MM. This task runs on the days and the time that you specified.
- Choose Custom if you want to use a custom cron expression to run your task. Then enter your expression in the Cron expression box.

For detailed information about schedule expressions syntax, see Schedule Expressions for Rules in the Amazon CloudWatch User Guide.

Editing a Task Schedule

You can configure scheduling when you initially create a task (p. 30), or you can edit a task schedule after a task is created. Use the following procedure to configure a schedule after you have created a task.

To edit a task schedule

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. On the navigation pane, choose Tasks, and then choose the task that you want to edit.
3. For Actions, choose Edit to open the Edit Tasks page and expand Schedule (optional).
4. In the Schedule (optional) section, configure your task to run on a schedule that you specify.
5. For Frequency, configure how frequently you want the task to run. For frequency configurations options, see Configuring a Task Schedule (p. 62).
Deleting a Task

If you no longer need a task, you can delete it from the DataSync Management Console.

To delete a task

1. In the navigation pane, choose Task.
2. For Actions, choose Delete.
3. In the Delete task dialog box that appears, choose Delete.

When you delete a task, you can't get it back.
Working with Locations

In this section, you can find information about how to create and configure locations.

A *location* defines the storage system or service that you want to read data from or write data to. AWS DataSync supports Network File System (NFS), Server Message Block (SMB), Amazon EFS, Amazon FSx for Windows File Server, and Amazon S3 as location types.

For NFS, SMB, Amazon FSx for Windows File Server, and Amazon EFS, the location is the export path where you want to mount your file system. For Amazon S3, the location is the prefix path that you want to mount and use as the root of the sync.

DataSync supports the following source and destination location combinations.

<table>
<thead>
<tr>
<th>Source (From)</th>
<th>Destination (To)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFS or SMB file system</td>
<td>Amazon S3</td>
</tr>
<tr>
<td>NFS or SMB file system</td>
<td>Amazon EFS</td>
</tr>
<tr>
<td>NFS or SMB file system</td>
<td>Amazon FSx for Windows File Server</td>
</tr>
<tr>
<td>Amazon EFS (configured as an NFS location)</td>
<td>Amazon EFS</td>
</tr>
<tr>
<td>Amazon EFS (configured as an NFS location)</td>
<td>Amazon S3</td>
</tr>
<tr>
<td>Amazon EFS (configured as an NFS location)</td>
<td>Amazon FSx for Windows File Server</td>
</tr>
<tr>
<td>Amazon FSx for Windows File Server (configured as an SMB location)</td>
<td>Amazon FSx for Windows File Server</td>
</tr>
<tr>
<td>Amazon FSx for Windows File Server (configured as an SMB location)</td>
<td>Amazon S3</td>
</tr>
<tr>
<td>Amazon FSx for Windows File Server (configured as an SMB location)</td>
<td>Amazon EFS</td>
</tr>
<tr>
<td>Amazon S3</td>
<td>NFS or SMB file system</td>
</tr>
<tr>
<td>Amazon EFS</td>
<td>NFS or SMB file system</td>
</tr>
<tr>
<td>Amazon FSx for Windows File Server</td>
<td>NFS or SMB file system</td>
</tr>
</tbody>
</table>

**Topics**

- Creating a Location for NFS (p. 65)
- Creating a Location for SMB (p. 66)
Creating a Location for NFS

DataSync supports the NFS v3, NFS v4.0, and NFS v4.1 protocols.

To create an NFS location

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose Locations. The locations that you previously created appear in the list of locations.
3. On the Locations page, choose Create location.
4. For Location type, choose NFS. You configure this location as a source or destination later.
5. For Agents, choose the agent that you want to use. If you have previously created agents, the agents appear in the list. The agent connects to your on-premises NFS server and makes it easier to securely transfer data between the on-premises location and AWS.
6. For NFS server, provide the DNS name or IP address of the NFS server.

DataSync automatically chooses the NFS version that it uses to read from an NFS location. To specify an NFS version, use the optional Version parameter for the NfsMountOptions (p. 220) API operation.
7. For Mount path, enter the mount path for your NFS location.
8. (Optional) For Key and Value, enter a key and value to tag your NFS location. A tag is a key-value pair that helps you manage, filter, and search for your locations.
9. When you are done, choose Create location.

For detailed information about these NFS location settings, see NFS Location Settings (p. 65).

NFS Location Settings

Following, you can find descriptions for the configuration settings for NFS locations in DataSync.

Agent

An agent is a VM that is deployed in your on-premises environment to connect to your on-premises location. An agent makes it easier to securely transfer data between the on-premises location and AWS. You can use an agent for more than one location.

If a task is using multiple agents, all the agents need to have the status Available for the task to run. If you use multiple agents for a source location, the status of all the agents must be Available for the task to run. Agents are automatically updated by AWS on a regular basis, using a mechanism that doesn't interrupt your tasks.

NFS server

The name of the NFS server, the IP address or DNS name of the NFS server. An agent that is installed on-premises uses this name to mount the NFS server in a network.

Mount path
The mount path for your NFS file system. This path must be a path that's exported by the NFS server, or a subdirectory of an exported path. This path should be such that it can be mounted by other NFS clients in your network. For information about how to resolve mount path issues, see Your Task Status Is Unavailable and Status Indicates a Mount Error (p. 114).

To transfer all the data in the folder you specified, DataSync needs to have permissions to read all the data. To ensure this, either configure the NFS export with `no_root_squash`, or ensure that the permissions for all of the files you want DataSync to allow read access for all users. Doing either enables the agent to read the files. For the agent to access directories, you must additionally enable all execute access.

**Tag**

A *tag* is a key-value pair that helps you manage, filter, and search for your location. Adding a tag is optional. We recommend using tags for naming your resources.

**Note**

DataSync supports the NFS v3, NFS v4.0, and NFS v4.1 protocols. DataSync automatically chooses the NFS version that it uses when reading from an NFS location. If you need to force DataSync to use a specific NFS version, see You Need DataSync to Use a Specific NFS or SMB Version to Mount Your Share (p. 113).

**Creating a Location for SMB**

When you use a location in a task, you configure it as the source or destination location. DataSync supports the SMB 2 and SMB 3 protocols. DataSync authenticates by using a user name and a password that you provide. This user can be a local user on your Windows File Server, or it can be a domain user defined in your Active Directory.

If the user is a domain user defined in your Active Directory, we recommend that you make this user a member of the built-in Active Directory group backup operators. Doing helps ensure sufficient permissions to files, folders, and file metadata.

**To create an SMB location**

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose Locations. The locations that you previously created appear in the list of locations.
3. On the Locations page, choose Create location.
4. For Location type, choose Server Message Block (SMB). You configure this location as a source or destination later.
5. For Agents, choose the agent that you want to use. The agent connects to your on-premises SMB server and makes it easier to securely transfer data between the on-premises location and AWS.
6. For SMB Server, provide the DNS name or IP address of the SMB server.
7. For Share name, enter the name of the share exported by your SMB server. You can include a folder from within this share. Specify the share by using slashes, for example `/path/to/folder`.

For **User**, enter the user who can mount the location and has the permissions to access files file share. To transfer data from or into the share you specified, DataSync needs to have permissions to read all the data or write to the share. To ensure this, we recommend that the user you configure is either a local Administrator or a member of the built-in Active Directory group (Backup Operators).
8. For **Password**, enter the password of the user who can mount the location and has the permissions to access the SMB file share.
9. (Optional) Expand the **Additional settings** section if you want to enter values for these settings:

   - (Optional) For **Domain**, enter the name of the domain the user is part of.
• (Optional) For **SMB version**, you can choose the SMB version to use. DataSync automatically chooses a version based on a negotiation with the SMB server. To use a specific SMB version, choose that version.

10. (Optional) For **Key** and **Value**, enter a key and value to tag your SMB location. A *tag* is a key-value pair that helps you manage, filter, and search for your locations.

11. When you are done, choose **Create location**.

**SMB Location Settings**

Following, you can find descriptions for the configuration settings for SMB locations in DataSync.

**Agent**

An *agent* is a VM that is deployed in your on-premises environment to connect to your on-premises location. An agent makes it easier to securely transfer data between the on-premises location and AWS. You can use an agent for more than one location.

If a task is using multiple agents, all the agents need to have the status **Available** for the task to run. If you use multiple agents for a source location, the status of all the agents must be **Available** for the task to run. Agents are automatically updated by AWS on a regular basis, using a mechanism that doesn't interrupt your tasks.

**SMB Server**

The name of the SMB server, the IP address or DNS name of the SMB server. An agent that is installed on-premises uses this name to mount the SMB server in a network.

**Share Name**

The name of the share exported by your SMB server. You can include a folder from within this share. Specify the share by using slashes, for example `/path/to/folder`.

**User**

The user who can mount the location and has the permissions to access SMB file share.

**Password**

The password of the user who can mount the location and has the permissions to access files and folders in the SMB file share.

**Domain**

The name of the domain that the user is part of.

**SMB Version**

DataSync automatically chooses the SMB version that it uses to read from an SMB location. If you need DataSync to use a specific SMB version, use this optional parameter.

**Tag**

A *tag* is a key-value pair that helps you manage, filter, and search for your location. Adding a tag is optional. We recommend using tags for naming your resources.

**Creating a Location for Amazon EFS**

A location for Amazon EFS is an endpoint for an Amazon EFS file system. If you don't have an Amazon EFS file system in the current AWS Region, create one. For information about how to create an Amazon
Considerations When Creating a Location for Amazon EFS

EFS file system, see Getting Started with Amazon Elastic File System in the Amazon Elastic File System User Guide.

Note
DataSync currently doesn't support transferring files to Amazon EFS volumes that are in virtual private clouds (VPCs) that have dedicated tenancy. For information about dedicated tenancy VPCs, see Creating a VPC with an Instance Tenancy of Dedicated in the Amazon EC2 User Guide for Linux Instances.

Topics
• Considerations When Creating a Location for Amazon EFS (p. 68)

To create an EFS location
1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose Locations. The locations that you previously created appear in the list of locations.
3. On the Create location page, choose EFS for Location type.
4. For File system, choose the EFS file system that you want to use as an endpoint. You configure this location as a source or destination later.
5. For Mount path, enter the mount path for your EFS file system. The path can include a subdirectory. If so, this is a subdirectory in the EFS file system that is used to read data from the EFS source or write data to the EFS destination. By default, DataSync uses the root directory.
6. For Subnet and Security Group, the DataSync console automatically chooses a subnet that includes a mount target for your Amazon EFS file system and this subnet’s default security group. We recommend using these default settings.

Note
DataSync uses the security group specified in this step to connect to your Amazon EFS file system. If the security group is configured to disallow connections from within itself, you have two options. One is to change this configuration to allow the security group to communicate within itself. The other is to choose a different subnet and security group, such that the following is true:

• The selected subnet contains a mount target for your EFS file system.
• The selected security group can communicate with that mount target's security group.

For detailed information about security groups and mount targets, see Security Groups for Amazon EC2 Instances and Mount Targets in the Amazon Elastic File System User Guide.
7. (Optional) Provide values for the Key and Value fields to tag the EFS file system. A tag is a key-value pair that helps you manage, filter, and search for your locations. We recommend using tags for naming your resources.
8. When you are done, choose Create location. The location that you just created appears in the list of locations.

Considerations When Creating a Location for Amazon EFS

Be sure to consider the following when creating a location for Amazon EFS:

• When you create an Amazon EFS file system in Bursting Throughput mode, you get an allocation of 2.1 TB worth of burst credits. All Amazon EFS file systems are able to burst up to 100 MB/s of throughput
when using Bursting Throughput mode. File systems that are larger than 1 TiB can burst to twice their baseline throughput.

DataSync consumes file system burst credits. This can have an impact on the performance of your applications. When using DataSync with a file system that has an active workload, consider using EFS Provisioned Throughput.

- Amazon EFS file systems that are in General Purpose performance mode have a limit of 7,000 file system operations per second. This limit can impact the maximum throughput DataSync can achieve when copying files.

For more information, see Amazon EFS Performance in the Amazon Elastic File System User Guide.

Creating a Location for Amazon FSx for Windows File Server

A location for Amazon FSx for Windows is an endpoint for an Amazon FSx for Windows File Server. AWS DataSync accesses your Amazon FSx for Windows File Server using the SMB protocol. It authenticates with the user name and password that you configure in the AWS DataSync console or AWS CLI.

To ensure sufficient permissions to files, folders, and file metadata, we recommend that you make this user a member of the file system administrators group. If you are using AWS Directory Service for Microsoft Active Directory with Amazon FSx for Windows File Server, the user must be a member of the AWS Delegated FSx Administrators. If you are using a self-managed Microsoft Active Directory with your Amazon FSx for Windows File Server, the user must be a member of one of two groups. These are the group of domain admins or the custom group you specified for file system administration when you created your file system.

The DataSync service mounts your file system from your virtual private cloud (VPC) from Elastic Network Interfaces managed by the DataSync service. DataSync fully manages the creation, the use, and the deletion of these network interfaces on your behalf.

If you don't have an Amazon FSx for Windows File Server in the current AWS Region, create one. For information about how to create an Amazon FSx for Windows File Server, see Getting Started with Amazon FSx in the Amazon FSx for Windows File Server User Guide.

**Note**

DataSync currently doesn't support transferring files to Amazon FSx for Windows File Server volumes that are in dedicated tenancy VPCs. For information about dedicated tenancy VPCs, see Creating a VPC with an Instance Tenancy of Dedicated in the Amazon EC2 User Guide for Linux Instances.

**To create an FSx location**

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose Locations. The locations that you previously created appear in the list of locations.
3. Choose Create Location to open the Create location page. For Location type, choose Amazon FSx for Windows File Server
4. For FSx for Windows File system, choose the Amazon FSx for Windows File Server that you want to use as an endpoint. You configure this location as a source or destination later.
5. For Share name, enter the mount path for your FSx file server. The path can include a subdirectory. If so, this is a subdirectory in the Amazon FSx for Windows File Server that is used to read data from the FSx location or write data to the FSx destination.

**Note**
The subdirectory must be specified with forward slashes, for example /path/to/folder.
6. For **Security Group**, the DataSync console automatically chooses the default security group of the subnet for the chosen Amazon FSx for Windows File Server. We recommend using these default settings.

   **Note**
   DataSync uses the security group specified in this step to connect to your Amazon FSx for Windows File Server. If the security group is configured to disallow connections from within itself, you have two options:
   
   - Change the security group configuration to allow the security group to communicate within itself.
   - Choose a different security group, so the selected security group can communicate with the mount target's security group.

7. In the **User settings** section, provide the information for Amazon FSx for Windows File Server:

   **User**
   The user that can mount the location and has the permissions to access FSx server.

   **Password**
   The password of the user that can mount the location and has the permissions to access files and folders in the Amazon FSx for Windows File Server.

   **Domain**
   (Optional) The name of the domain the Amazon FSx for Windows File Server belongs to.

8. (Optional) Provide values for the **Key** and **Value** fields to tag the Amazon FSx for Windows File Server. A tag is a key-value pair that helps you manage, filter, and search for your locations. We recommend using tags to name your resources.

9. When you are done, choose **Create location**. The location that you just created appears in the list of locations.

---

### Creating a Location for Amazon S3

A location for Amazon S3 is an endpoint for the Amazon S3 bucket that DataSync uses as a source or destination.

**Topics**
- Amazon S3 Location Settings (p. 71)
- Considerations When Working with S3 Storage Classes in DataSync (p. 72)
- Manually Configuring an IAM Role to Access Your S3 Bucket (p. 73)

**To create an S3 location**

2. In the navigation pane, choose **Locations**. The locations that you previously created appear in the list of locations.
3. On the **Create location** page, choose **Create location**.
4. For **Location type**, choose **Amazon S3 bucket**.
5. For **S3 Bucket**, choose the S3 bucket that you want to use as an endpoint. You configure this location as a source or destination later.
6. For **S3 storage class**, choose a storage class you want to transfer objects directly into. For information about storage classes, see Amazon S3 Storage Classes in the *Amazon Simple*...
Some storage classes have behaviors that can affect your S3 cost. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

7. For **Folder**, provide the name of a folder in S3. This is the S3 folder that DataSync uses, either to read data from for an S3 source location or write data to for an S3 destination.

8. For **IAM role**, choose Autogenerate for DataSync to automatically create a role with the required permissions.

   If DataSync has previously created such a role for this S3 bucket, that role is chosen as the default in the list. You can also create your own role and choose it from the list. For instructions on how to create an IAM role manually, see Manually Configuring an IAM Role to Access Your S3 Bucket (p. 73).

9. (Optional) For **Key** and **Value**, provide values to tag your S3 location. A **tag** is a key-value pair that helps you manage, filter, and search for your locations.

10. When you are done, choose **Create location**. The location that you just created appears in the list of locations.

**Note**

- When using object versioning, changes to object metadata create a new version of the object. A single DataSync task execution might create more than one version of an Amazon S3 object.

- In addition to the IAM policies that grant DataSync permissions, we recommend creating a multipart upload bucket policy for your S3 buckets. Doing this can help you control your storage costs. For more information, see the blog post S3 Lifecycle Management Update – Support for Multipart Uploads and Delete Markers.

**Amazon S3 Location Settings**

If the location you want to use as a source or a destination is an Amazon S3 bucket, you configure the following settings.

**S3 bucket**

The Amazon S3 bucket that you want to use as a source or destination location.

**Storage class**

This enables you to transfer objects directly into the Amazon S3 storage class that you choose. For information about storage classes, see Amazon S3 Storage Classes in the Amazon Simple Storage Service Developer Guide. Some storage classes have specific behaviors that can affect your S3 storage cost. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

**Folder**

This is the S3 folder that DataSync uses, either to read data from for an S3 source location or write data to for an S3 destination.

**IAM role**

The AWS Identity and Access Management (IAM) role that has permissions to access the S3 bucket.

For DataSync to access a destination S3 bucket, it requires access to your Amazon S3 bucket. To get this access, DataSync assumes the IAM role that you provide. The role requires an IAM policy and a security token service trust (STS) relationship. The policy determines which actions the role can perform. DataSync can create the role on your behalf. For instructions, see Creating a Location for Amazon
You can also create the role manually and choose it from the list in the console. For instructions, see Manually Configuring an IAM Role to Access Your S3 Bucket (p. 73).

**Tag**

A key-value pair that identifies the S3 location. By default, the DataSync console prepopulates a name value with the task or location name.

**Considerations When Working with S3 Storage Classes in DataSync**

DataSync can transfer objects directly into the Amazon S3 storage class that you choose. For more information about Amazon S3 storage classes, see Amazon S3 Storage Classes. Some storage classes have behaviors that can affect your Amazon S3 storage cost. For more information, see Amazon S3 Pricing.

Following, you can find some considerations for how each Amazon S3 storage class works with DataSync.

<table>
<thead>
<tr>
<th>S3 Storage Class</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>Choose Standard to store your frequently accessed files redundantly in multiple Availability Zones that are geographically separated. This is the default if you don't specify a storage class.</td>
</tr>
</tbody>
</table>
| Intelligent-Tiering | Choose Intelligent-Tiering to optimize storage costs by automatically moving data to the most cost-effective storage access tier.  
                   | Objects stored in the Intelligent-Tiering storage class can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. 
                   | Objects less than 128 KB are not eligible for autotiering in the Intelligent-Tiering storage class. These objects are stored in Standard. |
| Standard-IA      | Choose Standard-IA to store your infrequently accessed files redundantly in multiple Availability Zones that are geographically separated. 
                   | Objects stored in the Standard-IA storage class can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. 
                   | Objects less than 128 KB are smaller than the minimum capacity charge per object in the Standard-IA storage class. These objects are stored in Standard. |
| One Zone-IA      | Choose Standard-IA to store your infrequently accessed files in a single Availability Zone. 
                   | Objects stored in the One Zone-IA storage class can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. 
<pre><code>               | Objects less than 128 KB are smaller than the minimum capacity charge per object in the One Zone-IA storage class. These objects are stored in Standard. |
</code></pre>
<table>
<thead>
<tr>
<th>S3 Storage Class</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glacier</td>
<td>Choose S3 Glacier to archive the files for more active archives. Objects stored in S3 Glacier can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Objects less than 40 KB are smaller than the minimum capacity charge per object in the S3 Glacier storage class. These objects are stored in Standard. When using S3 Glacier as an Amazon S3 storage class, choose <strong>Verify only the data transferred</strong> to compare data and metadata checksums at the end of the transfer. <strong>Verify all data in the destination</strong> isn't an available option for this storage class, because it requires retrieving all existing objects from the destination.</td>
</tr>
<tr>
<td>Glacier Deep Archive</td>
<td>Choose S3 Glacier Deep Archive to archive your files for long-term data retention and digital preservation where data is accessed once or twice a year. Objects stored in S3 Glacier Deep Archive can incur additional charges for overwriting, deleting, or retrieving. Consider how often these objects change, how long you plan to keep these objects, and how often you need to access them. Objects less than 40 KB are smaller than the minimum capacity charge per object in the S3 Glacier Deep Archive storage class. These objects are stored in Standard. When using S3 Glacier Deep Archive as an Amazon S3 storage class, choose <strong>Verify only the data transferred</strong> to compare data and metadata checksums at the end of the transfer. <strong>Verify all data in the destination</strong> isn't an available option for this storage class, because it requires retrieving all existing objects from the destination.</td>
</tr>
</tbody>
</table>

**Manually Configuring an IAM Role to Access Your S3 Bucket**

When you use the DataSync Management Console to create an Amazon S3 location, DataSync automatically creates an IAM role that has the required permissions for you. If you want to create the IAM role manually, use the following procedure.

**To manually configure an IAM role to access your S3 bucket**

1. Open the IAM Management Console.
2. On the navigation pane, choose Roles, and then choose Create role to open the Create role page.
3. In the Select type of trusted entity section, make sure that AWS service is selected.
4. Under Choose the service that will use this role, choose DataSync, or manually configure it as shown in the following example.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
```
5. Choose Next: Permissions.
6. Choose AmazonS3FullAccess. You can also manually configure a more restricted policy. For an example of such a policy, see Amazon S3 Location Settings (p. 71).
7. (Optional) Choose Next: Tags to create tags for the role.
8. Choose Next: Review, choose the role name, and then choose Create role.
9. Open the DataSync Management Console.
10. Choose the refresh button on the right side of the IAM role list, and then choose the role that you just created.

How DataSync Handles Metadata and Special Files

DataSync saves metadata and special files when copying to and from file systems. The following section describes how DataSync handles metadata and special files (links and directories) for different storage systems.

Topics
- Metadata Copied by DataSync (p. 74)
- Links and Directories Copied by DataSync (p. 76)

Metadata Copied by DataSync

DataSync preserves metadata between storage systems that have similar metadata structures.

When you copy between an NFS server and Amazon EFS – In this case, DataSync copies the following metadata:
- File and folder modification timestamps
- User ID and group ID
- POSIX permissions

When you copy between SMB file share and Amazon FSx for Windows File Server – In this case, DataSync copies the following metadata:
- File timestamps: access time, modification time, and creation time
- File owner and file group security identifiers (SIDs)
- Standard file attributes:
  - Read-only (R)
  - Archive (A)
  - System (S)
  - Hidden (H)
  - Compressed (C)
  - Not content indexed (N)
• Encrypted (E)
• Temporary (T)
• Offline (O)

NTFS discretionary access lists (DACLs): access control entries (ACEs) that determine whether to grant access to an object

COMING SOON — NTFS system access control lists (SACLs), which are used by administrators to log attempts to access a secured object. Copying SACLs will be added later in 2020. For more information, contact us through AWS Support.

When copying between an NFS server and Amazon S3 – In this case, the following metadata is stored as Amazon S3 user metadata:

• File and folder modification timestamps
• User ID and group ID
• POSIX permissions

The file metadata that is stored in Amazon S3 user metadata is interoperable with file gateways in AWS Storage Gateway. A file gateway enables you to have on-premises file-based access to data stored in Amazon S3 by DataSync. The metadata is also interoperable with Amazon FSx for Lustre.

When DataSync copies objects that contain this metadata back to an NFS server, the file metadata is restored. Restoring metadata requires granting elevated permissions to the NFS server when you are Creating a Location for NFS (p. 65).

When you copy between storage systems that don’t have similar metadata structure – In this case, DataSync sets metadata using the following rules.

<table>
<thead>
<tr>
<th>If You Copy This Way</th>
<th>This Happens to Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>From an SMB share to Amazon EFS or Amazon S3</td>
<td>Default POSIX metadata is set for all files on the target NFS server or Amazon EFS, or stored in the Amazon S3 object’s metadata. This approach includes using the default POSIX user ID and group ID values.</td>
</tr>
<tr>
<td>From Amazon FSx for Windows to an NFS share</td>
<td>Ownership is set based on the Windows user that was specified in DataSync to access the Amazon FSx or SMB share. Default Windows permissions are assigned based on that file owner.</td>
</tr>
</tbody>
</table>

Default POSIX Metadata Applied by DataSync

When the source and destination don’t have a similar metadata structure, or when source metadata is missing, DataSync applies default POSIX metadata.

Specifically, DataSync applies this metadata in these situations:

• When transferring files from an Amazon S3 location to an Amazon EFS or NFS location, in cases where Amazon S3 objects don't have DataSync POSIX metadata
• When transferring from an SMB share to an NFS, Amazon S3, or Amazon EFS location
The following table shows the default POSIX metadata and permissions that DataSync applies.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>UID</td>
<td>65534</td>
</tr>
<tr>
<td>GID</td>
<td>65534</td>
</tr>
<tr>
<td>Folder Permission</td>
<td>0755</td>
</tr>
<tr>
<td>File Permission</td>
<td>0755</td>
</tr>
</tbody>
</table>

Links and Directories Copied by DataSync

The following section describes how DataSync handles working with copied hard links, symbolic links, and directories in different storage locations.

Hard links

Hard links are preserved when copying between an NFS server and Amazon EFS.

When copying to Amazon S3, each hard link is only transferred once. Separate Amazon S3 objects are created for each copy. If a hard link is unchanged in Amazon S3, it's correctly restored upon transfer to an NFS server or Amazon EFS.

Hard links aren't supported for copying between SMB file shares and Amazon FSx for Windows File Server. If DataSync encounters hard links in such a copy, they are skipped and logged to Amazon CloudWatch Logs. For more information about how DataSync works with CloudWatch Logs, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).

Symbolic links

Symbolic links are preserved when copying between an NFS server and Amazon EFS.

When copying to Amazon S3, the link target path is stored in the Amazon S3 object. The link is correctly restored upon transfer to an NFS server or Amazon EFS.

Symbolic links aren't supported for copying between SMB file shares and Amazon FSx for Windows File Server. If DataSync encounters symbolic links in such a copy, they are skipped and logged to CloudWatch Logs. For more information about how DataSync works with CloudWatch Logs, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).

Directories

When copying to or from Amazon S3 buckets, directories are represented as empty objects ending with "/".

Deleting a Location

Use the following procedure to delete any type of location.

To delete a location

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. On the navigation pane, choose Locations.
3. On the Locations page, choose the location that you want to delete.
4. Choose **Delete**, note the location ID that appears, enter `delete` in the text box, and choose **Delete**.
Working with Agents

An agent is a virtual machine (VM) that is owned by the user, and is used to read or write data from an on-premises storage system. To create an agent to read from an on-premises source location, you download the AWS DataSync .ova image and deploy it in a VMware ESXi hypervisor and activate it.

**Note**
The default credentials for the OVA are login: admin, password: password. You can change the password on the VM local console. You don't need to log in to the VM for basic DataSync functionality. The credentials are needed for troubleshooting tasks such as running a connectivity test or opening a support channel with AWS. The credentials include network-specific settings such as for setting up a static IP address. For information about how to access the VM local console, see Configuring Your Agent on the VM Local Console (p. 84).

**Topics**
- Creating and Activating an Agent (p. 78)
- Using AWS DataSync in a Virtual Private Cloud (p. 79)
- Editing Your Agent's Properties (p. 82)
- Using Multiple Agents for a Location (p. 82)
- Understanding Agent Statuses (p. 83)
- Deleting an Agent (p. 83)
- Configuring Your Agent for Multiple NICs (p. 83)

Creating and Activating an Agent

After you deploy an agent into your VMware ESXi or Amazon EC2 environment, activate the agent. Activate your agent in the AWS Region where the Amazon S3 bucket, Amazon EFS, or Amazon FSx for Windows File Server file system that you plan to use with DataSync resides. The activation process associates your agent with your AWS account in the most secure way available. After the agent is successfully activated, DataSync closes the agent's port 80.

All data transferred between the agent and AWS is encrypted with Transport Layer Security (TLS). A DataSync agent can communicate with AWS by using one of the following:

- **Connecting to a public service endpoint** – Data is transferred over the public internet.
- **Connecting to a private virtual private cloud (VPC) endpoint** – Data is transferred within your VPC, without crossing the public internet, to increase the security of the copied data.

For more information about activating an agent with a private VPC endpoint, see Using AWS DataSync in a Virtual Private Cloud (p. 79).

- **Connecting to a Federal Information Processing Standard (FIPS) endpoint** – Data is transferred over the public internet using processes that comply with Federal Information Processing Standards (FIPS).

Your agent is managed by AWS, and AWS automatically updates it without interrupting your tasks. To access the agent's local console, see Logging In to the AWS DataSync Local Console (p. 84). For
security reasons, you can't use Secure Shell (SSH) to access the agent or remotely connect to the agent's local console.

For the agent to work properly, make sure that your network is configured properly. For information on network requirements, see Network Requirements for DataSync (p. 10). You can use the VM's local console to test for internet connectivity. For more information, see Testing Your Agent Connection to the Internet (p. 87).

In some cases, an agent is activated but isn't functioning properly. This issue can come from problems with a network partition, firewall misconfiguration, or other events that mean the agent VM can't connect to AWS. For information about how to troubleshoot connectivity and activation issues, see Testing Your Agent Connection to the Internet (p. 87).

For instructions on how to create an agent on a VMware ESXi host, see Deploy Your DataSync Agent on VMware (p. 21).

For instructions on how to create an agent on an Amazon EC2 instance, see Deploy Your Agent as an EC2 Instance to Read Files from In-Cloud (p. 21).

### Using AWS DataSync in a Virtual Private Cloud

You can deploy AWS DataSync in your virtual private cloud (VPC) based on the Amazon VPC service by using VPC endpoints. With this feature, the connection between an agent and the DataSync service doesn't cross the public internet and doesn't require public IP addresses. These connection restrictions increase the security of your data by keeping network traffic within your VPC.

VPC endpoints for DataSync are powered by VPC Endpoint Services (AWS PrivateLink). AWS PrivateLink is a highly available, scalable AWS service that enables you to privately connect your VPC to supported AWS services. For more information, see VPC Endpoint Services (AWS PrivateLink) in the Amazon VPC User Guide.

To use VPC endpoints, you can transfer files using AWS Direct Connect or a virtual private network (VPN). With this kind of transfer, you use private IP addresses that are accessible only from inside your VPC.

### How DataSync Works with VPC Endpoints

The DataSync agent transfers data between on-premises storage and AWS. You deploy the agent as a virtual machine in the same local network as your source storage. This approach minimizes network overhead associated with transferring data using network protocols such as Network File System (NFS) and Server Message Block (SMB).

When you use DataSync with a private VPC endpoint, the DataSync agent can communicate directly with AWS without the need to cross the public internet.

### Configuring DataSync to Use Private IP Addresses for Data Transfer

In the following procedure, you can find the steps to configure a DataSync agent and a task that communicate with AWS by using VPC endpoints.

The diagram following illustrates the setup process.
Configuring DataSync to Use Private IP Addresses for Data Transfer

To configure a DataSync agent and task to communicate with AWS by using VPC endpoints

1. Choose the VPC and subnet where you want to set up the DataSync private IP addresses.

   The VPC should extend to your local environment, where your SMB or NFS storage is located, by using routing rules over AWS Direct Connect or VPN. This setup ensures that all communications between the DataSync agent and the DataSync service remain within the VPC.

2. Deploy a DataSync agent close to your local storage. The agent must be able to access your source storage location by using NFS or SMB. You can download the .ova file for the DataSync agent from the DataSync console. The agent doesn't need a public IP address. For more information about downloading and deploying an .ova image, see Step 1: Create an Agent (p. 33).

   **Note**

   You can use one agent for only one type of endpoint—private, public, or Federal Information Processing Standard (FIPS). If you already have an agent configured for transferring data over the public internet, deploy a new agent to transfer data to private DataSync endpoints. For detailed instructions, see Deploy an AWS DataSync Agent (p. 20).

3. In the VPC that you chose in step 1, create a security group to ensure access to the private IP addresses that DataSync uses. These addresses include one VPC endpoint for control traffic and four elastic network interfaces to use for data transfer. You use this security group to manage access to these private IP addresses and ensure that your agent can route to them.

   The agent needs to be able to establish connections to these IP addresses. In the security group attached to the endpoints, configure inbound rules to allow the agent's private IP address to connect to these endpoints.
4. Create a VPC endpoint for the DataSync service.

To do this, open the Amazon VPC console at https://console.aws.amazon.com/vpc/, and choose Endpoints from the navigation pane at left. Choose Create Endpoint.

For Service category, choose AWS service. For Service Name, choose DataSync in your AWS Region (for example, com.amazonaws.us-east-1.datasync). Then choose the VPC and security group that you chose in steps 1 and 3. Make sure that you clear the Enable Private DNS Name check box.

For additional details on creating VPC endpoints, see Creating an Interface Endpoint in Amazon VPC User Guide.

5. When your new VPC endpoint becomes available, make sure that the network configuration for your on-premises environment allows agent activation.

Activation is a one-time operation that securely associates the agent with your AWS account. To activate the agent, use a computer that can reach the agent by using port 80. After activation, this access can be revoked. The agent should be able to reach the private IP address of the VPC endpoint that you created in step 4.

To find this IP address, open the Amazon VPC console at https://console.aws.amazon.com/vpc/, and choose Endpoints from the navigation pane at left. Choose the DataSync endpoint, and check the Subnets list for the private IP address for the subnet that you chose. This is the IP address of your VPC endpoint.

Note Make sure to allow outbound traffic from the agent to the VPC endpoint by using ports 443, 1024–1064, and port 22. Port 22 is optional and is used for the AWS Support channel.

6. Activate the agent. If you have a computer that can route to the agent by using port 80 and that can access the DataSync console, open the console and choose Create Agent. In the service endpoint section, choose VPC endpoints using AWS PrivateLink.

Choose the VPC endpoint from step 4, the subnet from step 1, and the security group from step 3. Enter the agent's IP address.

If you can't access the agent and the DataSync console using the same computer, activate the agent using the command line from a computer that can reach the agent's port 80. For more information, see Step 1: Create an Agent (p. 33).

7. Choose Get Key, optionally enter an agent name and tags, and choose Create agent. Your new agent now appears on the Agents tab of the DataSync console. The green VPC Endpoint banner indicates that all tasks performed with this agent use private endpoints, without crossing the public internet.

8. Create your task by configuring a source and a destination for your data transfer. For more information on choosing endpoints, see Choose a Service Endpoint (p. 25).

To make transfer easier by using private IP addresses, your task creates four elastic network interfaces in the VPC and subnet that you chose.

9. Make sure that your agent can reach the four elastic network interfaces and related IP addresses that your task creates.

To find these IP addresses, open the Amazon EC2 console at https://console.aws.amazon.com/ec2/, and choose Network Interfaces on the dashboard. Enter the task ID into the search filter to see the four elastic network interfaces for the task. These are the elastic network interfaces used by your VPC endpoint. Make sure that you allow outbound traffic from the agent to these interfaces by using port 443.

You can now start your task. For each additional task that uses this agent, repeat step 9 to allow the task's traffic through port 443.
Editing Your Agent's Properties

You can get detailed information about your agent and edit the agent's properties on the agent's details page.

To edit your agent's properties

1. On the navigation pane, choose Agent to open the Agents page.
2. In the Agent ID column, choose the agent that you want to edit. The agent Details page opens. For agents that are activated with a private VPC endpoint, details about the VPC endpoint appear on this page.

3. Choose Edit and make the changes you want.

   **Important**
   You can't change the service endpoint from a public endpoint to a VPC endpoint, or from a VPC endpoint to a public endpoint.

Using Multiple Agents for a Location

For most workloads, we recommend that you activate one agent for each on-premises location. However, there are a couple of exceptions:

- Some workloads have tens of millions of small files. In these cases, we recommend up to four agents for each on-premises location.
- In some environments, each agent is on a network link less than 2.5 Gbps in bandwidth. For such environments, or environments that are otherwise network-limited, we recommend four agents for each on-premises location.

If you use multiple agents for a source location, the status of all the agents must be available for the task to run.

   **Note**
   If one of the source agents becomes unavailable, locations that include that unavailable agent can't be used, even if the other agents are available.
Understanding Agent Statuses

The following table describes each agent status, and if and when you should take action based on the status. When an agent is in use, it has running status all or most of the time.

<table>
<thead>
<tr>
<th>Agent Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONLINE</td>
<td>The agent is configured properly and is available to use. The ONLINE status is the normal running status for an agent.</td>
</tr>
<tr>
<td>OFFLINE</td>
<td>The agent's VM is turned off or the agent is in an unhealthy state. When the issue that caused the unhealthy state is resolved, the agent returns to ONLINE status.</td>
</tr>
</tbody>
</table>

Deleting an Agent

When you delete an agent, DataSync disassociates the agent from your AWS account. However, it doesn't delete the agent virtual machine (VM) from your on-premises environment.

**Note**
After you delete an agent, you can’t reactivate it.

To delete an agent
1. Choose Agents from the left navigation menu.
2. Choose the agent you want to delete.
3. Choose Delete, type delete in the text box that appears and choose Delete.

Configuring Your Agent for Multiple NICs

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

- **Maximizing throughput** – You might want to maximize throughput to an agent when network adapters are a bottleneck.
- **Network isolation** – Your NFS or SMB server might reside on a virtual LAN (VLAN) that lacks internet connectivity for security reasons.

In a typical multiple-adapter use case, one adapter is configured as the route by which the agent communicates with AWS (that is, as the default agent). Except for this one adapter, NFS or SMB locations must be in the same subnet as the adapter that connects to them. Otherwise, communication with the intended NFS or SMB locations might not be possible. In some cases, you might configure an NFS or SMB location on the same adapter that is used for communication with AWS. In these cases, NFS or SMB traffic for that server and AWS traffic flows through the same adapter.

In some cases, you might configure one adapter to connect to the DataSync console and then add a second adapter. In such a case, DataSync automatically configures the route table to use the second adapter as the preferred route.
Configuring Your Agent on the VM Local Console

For a DataSync setup that is deployed on-premises, you can perform the following maintenance tasks using the VM host's local console.

Topics

- Logging In to the AWS DataSync Local Console (p. 84)
- Configuring Your Agent Network Settings (p. 85)
- Testing Your Agent Connection to the Internet (p. 87)
- Viewing Your Agent System Resource Status (p. 89)
- Configuring a Network Time Protocol (NTP) Server (p. 90)
- Running AWS DataSync Commands on the Local Console (p. 92)
- Enabling AWS Support to Help Troubleshoot Your Running On-Premises Agent (p. 93)

Logging In to the AWS DataSync Local Console

When the VM is ready for you to log in, the login screen is displayed. If this is your first time logging in to the local console, you use the default user name and password to log in. These default login credentials give you access to menus where you can configure the agent's network settings and change the password from the local console. You don't need to know the default password to set a new password.

For security reasons, you can't ssh to the agent or remotely connect to it.
To log in to the agent's local console

- If this is your first time logging in to the local console, log in to the agent VM with the default credentials. The default user name is admin and the password is password. Otherwise, use your credentials to log in.

  Note
  We recommend changing the default password. You do this by running the passwd command from the local console menu (item 6 on the main menu). For information about how to run the command, see Running AWS DataSync Commands on the Local Console (p. 92).

Configuring Your Agent Network Settings

The default network configuration for the agent is Dynamic Host Configuration Protocol (DHCP). With DHCP, your agent is automatically assigned an IP address. In some cases, you might need to manually assign your agent's IP as a static IP address, as described following.

To configure your agent to use static IP addresses

1. Log in to your agent's local console:

2. On the AWS Appliance Activation - Configuration main menu, enter 2 to begin configuring your network.

3. On the Network Configuration menu, choose one of the following options.
### AWS Appliance Activation - Network Configuration

1: Describe Adapter  
2: Configure DHCP  
3: Configure Static IP  
4: Reset all to DHCP  
5: Set Default Adapter  
6: Edit DNS Configuration  
7: View DNS Configuration  
8: View Routes  

Press “x” to exit  

Enter command: _

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
</table>
| Get information about your network adapter | Enter 1.  
A list of adapter names appears, and you are prompted to enter an adapter name—for example, _eth0_. If the adapter you specify is in use, the following information about the adapter is displayed:  
- Media access control (MAC) address  
- IP address  
- Netmask  
- Agent IP address  
- DHCP enabled status  
You use the same adapter name when you configure a static IP address (option 3) as when you set your agent's default route adapter (option 5). |
| Configure DHCP                          | Enter 2.  
You are prompted to configure the network interface to use DHCP. |
| Configure a static IP address for your agent | Enter 3.  
You are prompted to enter the Network adapter name.  
**Important**  
If your agent has already been activated, you must shut it down and restart it from the DataSync console for the settings to take effect. |
### Testing Your Agent Connectivity to the Internet

You can use your agent's local console to test your internet connection. This test can be useful when you are troubleshooting network issues with your agent.

**To test your agent's connection to the internet**

1. Log in to your agent's local console:
2. On the AWS Appliance Activation - Configuration main menu, enter 3 to begin testing network connectivity.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reset all your agent's network configuration to DHCP</td>
<td>Enter 4. All network interfaces are set to use DHCP. <strong>Important</strong> If your agent has already been activated, you must shut down and restart your agent from the DataSync console for the settings to take effect.</td>
</tr>
<tr>
<td>Set your agent's default route adapter</td>
<td>Enter 5. The available adapters for your agent are shown, and you are prompted to choose one of the adapters—for example, <strong>eth0</strong>.</td>
</tr>
<tr>
<td>Edit your agent's DNS configuration</td>
<td>Enter 6. The available adapters of the primary and secondary DNS servers are displayed. You are prompted to provide the new IP address.</td>
</tr>
<tr>
<td>View your agent's DNS configuration</td>
<td>Enter 7. The available adapters of the primary and secondary DNS servers are displayed. <strong>Note</strong> For some versions of the VMware hypervisor, you can edit the adapter configuration in this menu.</td>
</tr>
<tr>
<td>View routing tables</td>
<td>Enter 8. The default route of your agent is displayed.</td>
</tr>
</tbody>
</table>
3. Choose option 3 for DataSync.

The console displays the available AWS Regions for DataSync.

4. Choose the AWS Region that you want to test. For information about AWS Regions and endpoints, see AWS Regions and Endpoints (p. 8).

Each endpoint in the selected AWS Region displays either a PASSED or FAILED message, as shown following.
For information about network and firewall requirements, see Network Requirements for DataSync (p. 10).

Viewing Your Agent System Resource Status

When your agent starts, it checks its virtual CPU cores, root volume size, and RAM. It then determines whether these system resources are sufficient for your agent to function properly. You can view the results of this check on the agent's local console.

To view the status of a system resource check

1. Log in to your agent's local console:
2. In the AWS Appliance Activation - Configuration main menu, enter 4 to view the results of a system resource check.

The console displays an [OK], [WARNING], or [FAIL] message for each resource as described in the table following.
### Configuring a Network Time Protocol (NTP) Server

You can view Network Time Protocol (NTP) server configurations and synchronize the VM time on your agent with your hypervisor host.

**To manage system time**

1. Log in to your agent's local console.
2. In the *AWS Appliance Activation - Configuration* main menu, enter 5 to manage your system's time.
3. In the **System Time Management** menu, enter 1.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>View and synchronize your VM time with NTP server time.</td>
<td>Enter 1.</td>
</tr>
<tr>
<td></td>
<td>The current time of your agent is displayed. Your agent determines the time difference from your agent VM, and your NTP server time and prompts you to synchronize the agent time with NTP time.</td>
</tr>
<tr>
<td></td>
<td>After your agent is deployed and running, in some scenarios the agent's time can drift. For example, suppose that there is a prolonged network outage and your hypervisor host and agent don't get time updates. In this case, the agent's time is different from the true time. When there is a time drift, a discrepancy occurs between the stated</td>
</tr>
</tbody>
</table>
Running AWS DataSync Commands on the Local Console

The VM local console in AWS DataSync helps provide a secure environment for configuring and diagnosing issues with your agent. Using the local console commands, you can perform maintenance tasks such as saving routing tables, connecting to AWS Support, and so on.

**To run a configuration or diagnostic command**

1. Log in to your agent's local console.
2. On the *AWS Appliance Activation - Configuration* main menu, enter 6 for *Command Prompt*.

3. On the *AWS Appliance Activation - Command Prompt* console, enter h, and then press the Return key.

<table>
<thead>
<tr>
<th><strong>To</strong></th>
<th><strong>Do This</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>times when operations such as snapshots occur and the actual times that the operations occur.</td>
<td></td>
</tr>
<tr>
<td>Edit your NTP server configuration</td>
<td>Enter 2. You are prompted to provide a preferred and a secondary NTP server.</td>
</tr>
<tr>
<td>View your NTP server configuration</td>
<td>Enter 3. Your NTP server configuration is displayed.</td>
</tr>
</tbody>
</table>
The console displays the **AVAILABLE COMMANDS** menu with what the commands do, as shown in the following screenshot.

4. At the command prompt, enter the command that you want to use and follow the instructions.

   To learn about a command, enter the command name at the command prompt.

---

**Enabling AWS Support to Help Troubleshoot Your Running On-Premises Agent**

You can allow AWS Support to access your agent, and assist you with troubleshooting agent issues. By default, AWS Support access to your DataSync is disabled. You enable this access through the host's local console. To give AWS Support access to your DataSync, you first log in to the local console for the host then connect to the support server.

**To enable AWS Support access to AWS DataSync**

1. Log in to your host's local console. Use the user name admin and the password password.

   The local console looks like the following.
2. At the prompt, type \texttt{6} to open the help menu.
3. Type \texttt{h} to open the \texttt{AVAILABLE COMMANDS} window.
4. In the \texttt{AVAILABLE COMMANDS} window, type \texttt{open-support-channel} to connect to customer support. You must allow TCP port 22 to initiate a support channel to AWS. When you connect to customer support, DataSync assigns you a support number. Make a note of your support number.

\begin{verbatim}
AWS DataSync - Configuration

# Currently connected network adapters:
#
# eth0: [IP Address]
#
1: Network Configuration
2: Test Network Connectivity
3: View System Resource Check (0 Errors)
4: Command Prompt

Press "x" to exit session

Enter command: 
\end{verbatim}

**Note**
The channel number is not a Transmission Control Protocol/User Datagram Protocol (TCP/UDP) port number. Instead, it makes a Secure Shell (SSH) (TCP 22) connection to servers and provides the support channel for the connection.

5. Once the support channel is established, provide your support service number to AWS Support so AWS Support can provide troubleshooting assistance.
6. When the support session is completed, type \texttt{q} to end it.
7. Type \texttt{exit} to log out of the DataSync local console.
8. Follow the prompts to exit the local console.
Security in AWS DataSync

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The 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 AWS DataSync, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using DataSync. The following topics show you how to configure DataSync to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your DataSync resources.

**Topics**
- Data Protection in AWS DataSync (p. 96)
- Identity and Access Management in AWS DataSync (p. 97)
- Logging AWS DataSync API Calls with AWS CloudTrail (p. 109)
- Compliance Validation for AWS DataSync (p. 111)
- Resilience in AWS DataSync (p. 111)
- Infrastructure Security in AWS DataSync (p. 111)

Data Protection in AWS DataSync

You can configure AWS DataSync to meet your security and compliance objectives, including protecting your data.

**Data Encryption**

AWS DataSync uses Transport Layer Security (TLS) to encrypt data that is transferred between your agent and AWS.

For data stored in Amazon S3, DataSync supports S3-managed encryption keys (SSE-S3) and Amazon EFS file system encryption for data stored in Amazon EFS. When reading and writing data from Amazon S3, DataSync works with Amazon S3 Default Encryption for S3 Buckets, as described in the Amazon Simple Storage Service Developer Guide. When reading and writing data from Amazon EFS, DataSync works with Amazon EFS file system encryption of data at rest, as described in Amazon Elastic File System User Guide.
Identity and Access Management in AWS DataSync

AWS uses security credentials to identify you and to grant you access to your AWS resources. You can use features of AWS Identity and Access Management (IAM) to allow other users, services, and applications to use your AWS resources fully or in a limited way, without sharing your security credentials.

By default, IAM identities (users, groups, and roles) don't have permission to create, view, or modify AWS resources. To allow user, groups and roles to access AWS DataSync resources and interact with the DataSync console and API, you should use an IAM policy that grants permission to use the specific resources and API actions they will need. You then attach the policy to the IAM identity that requires access. For an overview of the basic elements for a policy, see Overview of Managing Access Permissions for DataSync (p. 104).

Topics
• Using Identity-Based Policies (IAM Policies) for DataSync (p. 97)
• DataSync API Permissions: Actions, Resources (p. 100)
• Overview of Managing Access Permissions for DataSync (p. 104)

The following sections provide details on how you can use AWS Identity and Access Management (IAM) and DataSync to help secure your resources by controlling who can access them:

• Authentication (p. 107)
• Permissions (p. 108)

We recommend that you read the these sections.

Using Identity-Based Policies (IAM Policies) for DataSync

Account administrator can attach identity-based policies to IAM identities, users, groups, roles, services, and resources.

This topic provides examples of identity-based policies in which an account administrator can attach permissions policies to IAM identities (that is, users, groups, and roles).

Important
We recommend that you first review the introductory topics that explain the basic concepts and options available for you to manage access to your DataSync resources. For more information, see Overview of Managing Access Permissions for DataSync (p. 104).

The sections in this topic cover the following:
• AWS Managed Policies for DataSync (p. 98)
• Permissions Required to Use the DataSync Console (p. 98)
• Customer Managed Policy Examples (p. 100)

The following shows an example of a permissions policy.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowsSpecifiedActionsOnAllTasks",
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```
Using Identity-Based Policies (IAM Policies)

The policy has one statement (note the `Action` and `Resource` elements in the statements):

- The statement grants permissions for two DataSync actions (`datasync:DescribeTask` and `datasync:ListTasks`) on a task resource using the Amazon Resource Name (ARN) for the task. The ARN specifies a wildcard character (*) because the user is allowed to perform the two actions on tasks. To limit permissions for the actions to a specific task, create a separate statement for that action in the policy and specify the task ID instead of the wildcard in that statement.

### AWS Managed Policies for DataSync

AWS addresses many common use cases by providing standalone IAM policies that are created and administered by AWS. Managed policies grant necessary permissions for common use cases so you can avoid having to investigate what permissions are needed. For more information about AWS managed policies, see AWS Managed Policies in the IAM User Guide.

The managed policies that are created by AWS grant the required permissions for common use cases. You can attach these policies to your IAM users, groups, and roles, based on the access that they need to DataSync:

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

**Note**

You can review these permissions policies by signing in to the IAM console and searching for specific policies there.

You can also create your own custom IAM policies to allow permissions for AWS DataSync API actions. You can attach these custom policies to the IAM users or groups that require those permissions. For more information about AWS managed policies, see AWS Managed Policies in the IAM User Guide.

### Permissions Required to Use the DataSync Console

To use the DataSync console, you require AWSDataSyncFullAccess permissions.

The following is an example policy that grants these permissions. This is an AWS managed policy that provides read-only access to DataSync.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [ "datasync:*" ]
        }
    ]
}
```
"ec2:CreateNetworkInterface",
"ec2:CreateNetworkInterfacePermission",
"ec2:DeleteNetworkInterface",
"ec2:DescribeNetworkInterfaces",
"ec2:DescribeSecurityGroups",
"ec2:DescribeSubnets",
"ec2:ModifyNetworkInterfaceAttribute",
"elasticfilesystem:DescribeFileSystems",
"elasticfilesystem:DescribeMountTargets",
"iam:GetRole",
"iam:ListRoles",
"logs:CreateLogGroup",
"logs:DescribeLogGroups",
"s3:ListAllMyBuckets",
"s3:ListBucket"
],
"Resource": "**",
{
  "Effect": "Allow",
  "Action": [
    "iam:PassRole"
  ],
  "Resource": "**",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": [
        "datasync.amazonaws.com"
      ]
    }
  }
}]
}
Customer Managed Policy Examples

In this section, you can find example user policies that grant permissions for various DataSync actions. These policies work when you are using AWS SDKs and the AWS CLI. When you are using the console, you need to grant additional permissions specific to the console, which is discussed in Permissions Required to Use the DataSync Console (p. 98).

**Note**

All examples use the US West (Oregon) Region (us-west-2) and contain fictitious account IDs and resource IDs.

**Topics**

- Example 1: Create a Trust Relationship That Allows DataSync to Access Your Amazon S3 Bucket (p. 100)
- Example 2: Allow DataSync to Read and Write to Your Amazon S3 Bucket (p. 100)
- Example 3: Allow DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 100)

**Example 1: Create a Trust Relationship That Allows DataSync to Access Your Amazon S3 Bucket**

The following is an example of a trust policy that allows DataSync to assume an IAM role. This role allows DataSync to access an S3 bucket.

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

**Example 2: Allow DataSync to Read and Write to Your Amazon S3 Bucket**

You provide the required policy that grants DataSync the minimal permissions to read and write data to your S3 bucket.

For an example of such a policy, see Amazon S3 Location Settings (p. 71).

**Example 3: Allow DataSync to Upload Logs to Amazon CloudWatch Log Groups**

Data Sync requires permissions to be able to upload logs to your CloudWatch Log Groups. You can use CloudWatch Log Groups to monitor and debug your tasks.

For an example a IAM policy that grants such permissions, see Allowing DataSync to Upload Logs to Amazon CloudWatch Log Groups (p. 47).

**DataSync API Permissions: Actions, Resources**

When you are setting up Permissions (p. 108) and writing permissions policies that you can attach to an IAM identity (identity-based policies), you can use the following as a reference. We list each AWS DataSync API operation, the corresponding actions for which you can grant permissions to perform the
action, and the AWS resource for which you can grant the permissions. You specify the actions in the policy's Action field, and you specify the resource value in the policy's Resource field.

You can use AWS-wide condition keys in your DataSync policies to express conditions. For a complete list of AWS-wide keys, see Available Keys in the IAM User Guide.

Note
To specify an action, use the `datasync:` prefix followed by the API operation name (for example, `datasync:CreateTask`).

For a list of DataSync resources with the ARN format, see DataSync Resources and Operations (p. 104).

DataSync API Operations and Required Permissions for Actions

CancelTaskExecution

**Action(s):** `datasync:CancelTaskExecution`

**Resource:**
`arn:aws:datasync:region:account-id:task/task-id/execution/exec-id`

CreateAgent

**Action(s):** `datasync:CreateAgent`

**Resource:** None

CreateLocationEfs

**Action(s):** `datasync:CreateLocationEfs`

**Resource:**

and
`arn:aws:ec2:region:account-id:subnet/subnet-id` and

CreateLocationFSxWindows

**Action(s):** `datasync:CreateLocationFSxWindows`

**Resource:**

and
`arn:aws:ec2:region:account-id:subnet/subnet-id` and

CreateLocationNfs

**Action(s):** `datasync:CreateLocationNfs`

**Resource:**
`arn:aws:datasync:region:account-id:agent/agent-id`

CreateLocationS3

**Action(s):** `datasync:CreateLocationS3`

**Resource:**
`arn:aws:s3:::bucket-name` and `arn:aws:iam::account-id:role/role-name`

CreateLocationSmb

**Action(s):** `datasync:CreateLocationSmb`

**Resource:**
`arn:aws:datasync:region:account-id:agent/agent-id`
CreateTask

Action(s): datasync:CreateTask
Resource: arn:aws:datasync:*:*:location/*

DeleteAgent

Action(s): datasync:DeleteAgent
Resource: arn:aws:datasync:*:*:agent/*

DeleteLocation

Action(s): datasync:DeleteLocation
Resource: arn:aws:datasync:*:*:location/*

DeleteTask

Action(s): datasync:DeleteTask
Resource: arn:aws:datasync:*:*:task/*

DescribeAgent

Action(s): datasync:DescribeAgent
Resource: arn:aws:datasync:*:*:agent/*

DescribeLocationEFS

Action(s): datasync:DescribeLocationEFS
Resource: arn:aws:datasync:*:*:location/*

DescribeLocationNFS

Action(s): datasync:DescribeLocationNFS
Resource: arn:aws:datasync:*:*:location/*

DescribeLocationS3

Action(s): datasync:DescribeLocationS3
Resource: arn:aws:datasync:*:*:location/*

DescribeLocationSMB

Action(s): datasync:DescribeLocationSMB
Resource: arn:aws:datasync:*:*:location/*

DescribeTask

Action(s): datasync:DescribeTask
Resource: arn:aws:datasync:*:*:task/*

DescribeTaskExecution

Action(s): datasync:DescribeTaskExecution
Resource: arn:aws:datasync:*:*:task/*/execution/*

ListAgents

Action(s): datasync:ListAgents
**Resource:** None

**ListLocations**

**Action(s):** datasync:ListLocations

**Resource:** None

**ListTagsForResource**

**Action(s):** datasync:ListTagsForResource

**Resource:** arn:aws:datasync::\[region\]:\[account-id\]:task/\[task-id\]

or

arn:aws:datasync::\[region\]:\[account-id\]:task/\[task-id\]/execution/\[exec-id\]

or

arn:aws:datasync::\[region\]:\[account-id\]:agent/\[agent-id\]

or

arn:aws:datasync::\[region\]:\[account-id\]:location/\[location-id\]

**ListTaskExecutions**

**Action(s):** datasync:ListTaskExecutions

**Resource:** None

**ListTasks**

**Action(s):** datasync:ListTasks

**Resource:** None

**StartTaskExecution**

**Action(s):** datasync:StartTaskExecution

**Resource:** arn:aws:datasync::\[region\]:\[account-id\]:task/\[task-id\]

**TagResource**

**Action(s):** datasync:TagResource

**Resource:** arn:aws:datasync::\[region\]:\[account-id\]:task/\[task-id\]

or

arn:aws:datasync::\[region\]:\[account-id\]:task/\[task-id\]/execution/\[exec-id\]

or

arn:aws:datasync::\[region\]:\[account-id\]:agent/\[agent-id\]

or

arn:aws:datasync::\[region\]:\[account-id\]:location/\[location-id\]

**UntagResource**

**Action(s):** datasync:UntagResource

**Resource:** arn:aws:datasync::\[region\]:\[account-id\]:task/\[task-id\]
or

arn:aws:datasync:region:account-id:task/task-id/execution/exec-id

or

arn:aws:datasync:region:account-id:agent/agent-id

or

arn:aws:datasync:region:account-id:location/location-id

UpdateAgent

**Action(s):** datasync:UpdateAgent

**Resource:** arn:aws:datasync:region:account-id:agent/agent-id

UpdateTask

**Action(s):** datasync:UpdateTask

**Resource:** arn:aws:datasync:region:account-id:task/task-id

Related Topics

- Permissions (p. 108)
- Customer Managed Policy Examples (p. 100)

## Overview of Managing Access Permissions for DataSync

Every AWS resource is owned by an AWS account, and permissions to create or access a resource are governed by permissions policies. An account administrator can attach permissions policies to IAM identities (that is, users, groups, and roles), and some services (such as AWS Lambda) also support attaching permissions policies to resources.

**Note**

An **account administrator** (or administrator user) is a user with administrator privileges. For more information, see IAM Best Practices in the IAM User Guide.

When granting permissions, you decide who is getting the permissions, the resources they get permissions for, and the specific actions that you want to allow on those resources.

**Topics**

- DataSync Resources and Operations (p. 104)
- Understanding Resource Ownership (p. 105)
- Managing Access to Resources (p. 105)
- Specifying Policy Elements: Actions, Effects, Resources, and Principals (p. 106)
- Specifying Conditions in a Policy (p. 107)
- Controlling Access (p. 107)

## DataSync Resources and Operations

In DataSync, the primary resources are task, location, agent and task execution.
These resources have unique Amazon Resource Names (ARNs) associated with them as shown in the following table.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>ARN Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task ARN</td>
<td>arn:aws:datasync:region:account-id:task/task-id</td>
</tr>
<tr>
<td>Location ARN</td>
<td>arn:aws:datasync:region:account-id:location/location-id</td>
</tr>
<tr>
<td>Agent ARN</td>
<td>arn:aws:datasync:region:account-id:agent/agent-id</td>
</tr>
<tr>
<td>Task Execution ARN</td>
<td>arn:aws:datasync:region:account-id:task/task-id/execution/exec-id</td>
</tr>
</tbody>
</table>

To grant permissions for specific API operations, such as creating a task, DataSync defines a set of actions that you can specify in a permissions policy to grant permissions for specific API operations. An API operation can require permissions for more than one action. For a list of all the DataSync API actions and the resources they apply to, see DataSync API Permissions: Actions, Resources (p. 100).

Understanding Resource Ownership

A resource owner is the AWS account that created the resource. That is, the resource owner is the AWS account of the principal entity (the root account, an IAM user, or an IAM role) that authenticates the request that creates the resource. The following examples illustrate how this works:

- If you use the root account credentials of your AWS account to create a task, your AWS account is the owner of the resource (in DataSync, the resource is the task).
- If you create an IAM user in your AWS account and grant permissions to the CreateTask action to that user, the user can create a task. However, your AWS account, to which the user belongs, owns the task resource.
- If you create an IAM role in your AWS account with permissions to create a task, anyone who can assume the role can create a task. Your AWS account, to which the role belongs, owns the task resource.

Managing Access to Resources

A permissions policy describes who has access to what. The following section explains the available options for creating permissions policies.

Note

This section discusses using IAM in the context of DataSync. It doesn't provide detailed information about the IAM service. For complete IAM documentation, see What Is IAM? in the IAM User Guide. For information about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the IAM User Guide.

Policies attached to an IAM identity are referred to as identity-based policies (IAM polices) and policies attached to a resource are referred to as resource-based policies. DataSync supports only identity-based policies (IAM policies).

Topics

- Identity-Based Policies (IAM Policies) (p. 106)
- Resource-Based Policies (p. 106)
Identity-Based Policies (IAM Policies)

You can attach policies to IAM identities. For example, you can do the following:

- **Attach a permissions policy to a user or a group in your account** – An account administrator can use a permissions policy that is associated with a particular user to grant permissions for that user to create a DataSync resource, such as a task, location, agent, or task execution.

- **Attach a permissions policy to a role (grant cross-account permissions)** – You can attach an identity-based permissions policy to an IAM role to grant cross-account permissions. For example, the administrator in Account A can create a role to grant cross-account permissions to another AWS account (for example, Account B) or an AWS service as follows:
  1. Account A administrator creates an IAM role and attaches a permissions policy to the role that grants permissions on resources in Account A.
  2. Account A administrator attaches a trust policy to the role identifying Account B as the principal who can assume the role.
  3. Account B administrator can then delegate permissions to assume the role to any users in Account B. Doing this allows users in Account B to create or access resources in Account A. The principal in the trust policy can also be an AWS service principal if you want to grant an AWS service permissions to assume the role.

For more information about using IAM to delegate permissions, see Access Management in the IAM User Guide.

The following is an example policy that grants permissions to all List* actions on all resources. This action is a read-only action. Thus, the policy doesn't allow the user to change the state of the resources.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllListActionsOnAllResources",
            "Effect": "Allow",
            "Action": [
                "datasync:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

For more information about using identity-based policies with DataSync, see Using Identity-Based Policies (IAM Policies) for DataSync (p. 97). For more information about users, groups, roles, and permissions, see Identities (Users, Groups, and Roles in the IAM User Guide).

Resource-Based Policies

Other services, such as Amazon S3, support resource-based permissions policies. For example, you can attach a policy to an Amazon S3 bucket to manage access permissions to that bucket. DataSync doesn't support resource-based policies.

Specifying Policy Elements: Actions, Effects, Resources, and Principals

For each DataSync resource (see DataSync API Permissions: Actions, Resources (p. 100)), the service defines a set of API operations (see Actions). To grant permissions for these API operations, DataSync defines a set of actions that you can specify in a policy. For example, for the DataSync resource, the
The following actions are defined: CreateTask, DeleteTask, and DescribeTask. Note that, performing an API operation can require permissions for more than one action.

The following are the most basic policy elements:

- **Resource** – In a policy, you use an Amazon Resource Name (ARN) to identify the resource to which the policy applies. For DataSync resources, you can use the wildcard character (*) in IAM policies. For more information, see DataSync Resources and Operations (p. 104).

- **Action** – You use action keywords to identify resource operations that you want to allow or deny. For example, depending on the specified Effect, the datasync:CreateTask permission allows or denies the user permissions to perform the DataSync CreateTask operation.

- **Effect** – You specify the effect when the user requests the specific action—this can be either allow or deny. If you don't explicitly grant access to (allow) a resource, access is implicitly denied. You can also explicitly deny access to a resource, which you might do to make sure that a user cannot access it, even if a different policy grants access.

- **Principal** – In identity-based policies (IAM policies), the user that the policy is attached to is the implicit principal. For resource-based policies, you specify the user, account, service, or other entity that you want to receive permissions (applies to resource-based policies only). DataSync doesn't support resource-based policies.

To learn more about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the IAM User Guide.

For a table showing all of the DataSync API actions, see DataSync API Permissions: Actions, Resources (p. 100).

**Specifying Conditions in a Policy**

When you grant permissions, you can use the IAM policy language to specify the conditions when a policy should take effect when granting permissions. For example, you might want a policy to be applied only after a specific date. For more information about specifying conditions in a policy language, see Condition in the IAM User Guide.

To express conditions, you use predefined condition keys. There are no condition keys specific to DataSync. However, there are AWS-wide condition keys that you can use as appropriate. For a complete list of AWS-wide keys, see Available Keys in the IAM User Guide.

**Controlling Access**

In this section, you can find information about how to control access to AWS Resources.

**Authentication**

You can access AWS as any of the following types of identities:

- **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 user** – An IAM user is an identity within your AWS account that has specific custom permissions (for example, permissions to create a task in DataSync). You can use an IAM user name and password to sign in to secure AWS webpages like the AWS Management Console, AWS Discussion Forums, or the AWS Support Center.
In addition to a user name and password, you can also generate access keys for each user. You can use these keys when you access AWS services programmatically, either through one of the several SDKs or by using the AWS Command Line Interface (CLI). The SDK and CLI tools use the access keys to cryptographically sign your request. If you don't use AWS tools, you must sign the request yourself. DataSync supports 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.

- **IAM role** – An IAM role is an IAM identity that you can create in your account that has specific permissions. An IAM role is similar to an IAM user in that it is an AWS identity with permissions policies that determine what the identity can and cannot do in AWS. However, instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it. Also, a role does not have standard long-term credentials such as a password or access keys associated with it. Instead, when you assume a role, it provides you with temporary security credentials for your role session. IAM roles with temporary credentials are useful in the following situations:

- **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.

- **AWS service access** – A service role is an IAM role that a service assumes to perform actions in your account on your behalf. When you set up some AWS service environments, you must define a role for the service to assume. This service role must include all the permissions that are required for the service to access the AWS resources that it needs. Service roles vary from service to service, but many allow you to choose your permissions as long as you meet the documented requirements for that service. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. You can create, modify, and delete a service role from within IAM. For example, you can create a role that allows Amazon Redshift to access an Amazon S3 bucket on your behalf and then load data from that bucket into an Amazon Redshift cluster. For more information, see Creating a Role to Delegate Permissions to an AWS Service in the IAM User Guide.

- **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.

**Permissions**

You can have valid credentials to authenticate your requests, but unless you have permissions you cannot create or access DataSync resources. For example, you must have permissions to create a task in DataSync.

The following sections provides and overview and describe how to manage permissions for DataSync.

- **Overview of Managing Access Permissions for DataSync (p. 104)**
• Identity-Based Policies (IAM Policies) (p. 106)

Logging AWS DataSync API Calls with AWS CloudTrail

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

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for AWS DataSync. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in Event history. Using the information collected by CloudTrail, you can determine the request that was made to AWS DataSync, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Working with AWS DataSync Information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in AWS DataSync, that activity is recorded in a CloudTrail event along with other AWS service events in Event history. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for AWS DataSync, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all AWS Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

• Overview for Creating a Trail
• CloudTrail Supported Services and Integrations
• Configuring Amazon SNS Notifications for CloudTrail
• Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

All AWS DataSync actions are logged by CloudTrail and are documented in the https://docs.aws.amazon.com/datasync/latest/userguide/API_Operations.html. For example, calls to the CreateAgent, CreateTask and ListLocations actions generate entries in the CloudTrail log files.

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

• Whether the request was made with root or AWS Identity and Access Management (IAM) 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.
Understanding AWS DataSync Log File Entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don’t appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the CreateTask action.

```json
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAJOERGY7LS5PKXTMXO",
    "arn": "arn:aws:iam::123456789012:user/user1",
    "accountId": "123456789012",
    "accessKeyId": "access key",
    "userName": "user1",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-12-13T14:56:46Z"
      }
    },
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2018-12-13T14:57:02Z",
  "eventSource": "datasync.amazonaws.com",
  "eventName": "CreateTask",
  "awsRegion": "ap-southeast-1",
  "sourceIPAddress": "12.345.123.45",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {
    "name": "MyTask-NTIzMzY1",
    "tags": [],
    "destinationLocationArn": "arn:aws:datasync:ap-southeast-1:123456789012:location/loc-020c33c5d9966f40a",
    "options": {
      "bytesPerSecond": -1,
      "verifyMode": "POINT_IN_TIME_CONSISTENT",
      "uid": "INT_VALUE",
      "posixPermissions": "PRESERVE",
      "mtime": "PRESERVE",
      "gid": "INT_VALUE",
      "preserveDevices": "NONE",
      "preserveDeletedFiles": "REMOVE",
      "atime": "BEST_EFFORT"
    },
    "sourceLocationArn": "arn:aws:datasync:ap-southeast-1:123456789012:location/loc-04aaa9c609b12135d"
  },
  "responseElements": {
    "taskArn": "arn:aws:datasync:ap-southeast-1:123456789012:task/task-00e5db3f3f41f6cd2"
  },
  "requestID": "5890e03c-fee7-11e8-8b63-0b409054d4dc",
  "eventID": "e5f59b6a-05e6-4412-bd56-440d872e90e9",
  "eventType": "AwsApiCall"
}
```
Compliance Validation for AWS DataSync

Third-party auditors assess the security and compliance of AWS DataSync as part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

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 DataSync is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. If your use of DataSync is subject to compliance with standards such as HIPAA, PCI, or FedRAMP, AWS provides resources to help:

- **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.
- **AWS Config** – This AWS service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- **AWS Security Hub** – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in AWS DataSync

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

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

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

Infrastructure Security in AWS DataSync

As a managed service, AWS DataSync is protected by the AWS global network security procedures that are described in the Amazon Web Services: Overview of Security Processes whitepaper.

You use AWS published API calls to access DataSync through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support
cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.
Troubleshooting AWS DataSync Issues

Following, you can find information on how to troubleshoot AWS DataSync issues.

Topics

- You Need DataSync to Use a Specific NFS or SMB Version to Mount Your Share (p. 113)
- You Get a "Failed to Retrieve Agent Activation Key" Error (p. 114)
- You Can't Activate an Agent Created Using a VPC Endpoint (p. 114)
- Your Task Status Is Unavailable and Status Indicates a Mount Error (p. 114)
- Your Task Execution Fails with an Input/Output Error Message (p. 115)
- Your Task Execution is Stuck in Launching Status (p. 115)
- Your Task Execution Fails with a Permissions Denied Error Message (p. 115)
- Preparing Status for a Task Execution Takes Longer Than Expected to Complete (p. 116)
- Verifying Status for a Task Execution Takes Longer Than Expected to Complete (p. 116)
- Your Storage Cost Is Higher Than Expected (p. 116)
- How to Enable AWS Support to Help Troubleshoot Your Running On-Premises Agent (p. 117)

You Need DataSync to Use a Specific NFS or SMB Version to Mount Your Share

DataSync automatically selects the Network File System (NFS) or Server Message Block (SMB) version that is used to access your location. If you need DataSync to use a specific version, use the DataSync API or the AWS CLI. For SMB, you also can choose the version from the DataSync console.

Action to Take

To make sure that DataSync chooses a specific version, do the following:

- For NFS, use the optional Version parameter for the CreateLocationNfs (p. 139) API operation.
- For SMB, use the optional Version parameter for the CreateLocationSmb (p. 148) API operation.

The following AWS CLI commands create an NFS source location and cause DataSync to use NFS version 4.0. Specify the subdirectory option with forward slashes, for example /path/to/folder.

```sh
$ aws datasync create-location-nfs --server-hostname your-server-address --on-prem-config AgentArns=your-agent-arns --subdirectory nfs-export-path --mount-options Version=NFS4_0
```

The following AWS CLI commands create an SMB source location and cause DataSync to use SMB version 3. Specify the subdirectory option with forward slashes, for example /path/to/folder.

```sh
$ aws datasync create-location-smb --server-hostname
```
You Get a "Failed to Retrieve Agent Activation Key" Error

When you are activating your DataSync agent, the agent connects to the specified endpoint to request an activation key. You can get this error in non-VPC endpoint use cases. For example, when your agent is deployed on-premises and your firewall settings block the connection. You can also get this error if your agent is deployed as an Amazon EC2 instance and the security groups are locked down.

**Action to Take**

Verify that your security group is set up to allow your agent to connect to the VPC endpoint and that you have allowed the required ports. For information about required ports, see Network Requirements for DataSync (p. 10).

Also, check your firewall and router settings and make sure that they allow communication with endpoints in AWS. For information about endpoint communication, see Network Requirements When Using Public Service Endpoints or FIPS Endpoints (p. 14).

You Can't Activate an Agent Created Using a VPC Endpoint

If you are having issues when you are activating an agent that is created using a VPC endpoint, open a support channel against your VPC endpoint ENI. For information about Support Channel, see Enabling AWS Support to Help Troubleshoot Your Running On-Premises Agent (p. 93).

Your Task Status Is Unavailable and Status Indicates a Mount Error

When you create a task, your task status might transition from CREATING to UNAVAILABLE when the agent that you chose can't mount the location that you specified during configuration.

**Action to Take**

First, make sure that the NFS server and export that you specified are both valid. If they aren't, delete the task, create a new one using the correct NFS server, and then export. For information more information, see Create an NFS Location (p. 36).

If the NFS server and export are both valid, it generally indicates one of two things. Either a firewall is preventing the agent from mounting the NFS server, or the NFS server isn't configured to allow the agent to mount it.

Make sure that there is no firewall between the agent and the NFS server. Then make sure that the NFS server is configured to allow the agent to mount the export end specified in the task. For information about network and firewall requirements, see Network Requirements for DataSync (p. 10).
Your Task Execution Fails with an Input/Output Error Message

You can get an Input/Output Error error message if your NFS server fails I/O requests that are issued by the DataSync agent. This can occur for reasons such as disk failure on the NFS server, changes in firewall configuration, and a network router failure.

**Action to Take**

First, check your NFS server's logs and metrics to determine if the problem started on the NFS server. If yes, resolve the issue that you discover.

Next, check that your network configuration hasn't changed. To check if the NFS server is configured correctly and accessible to DataSync through the network, do the following:

1. Set up another NFS client on the same network subnet as the DataSync agent.
2. Mount your share on that client.
3. Validate that the client can read and write to the share successfully.

Your Task Execution is Stuck in Launching Status

Your task execution can become stuck in Launching status when DataSync can't instruct the specified source agent to begin a task. This issue usually occurs because the agent either is powered off or has lost network connectivity.

**Action to Take**

Make sure that the agent is connected and the status is **ONLINE**. If the status is **OFFLINE**, then the agent is not connected. For information about how to test network connectivity, see Testing Your Agent Connection to the Internet (p. 87).

Next, make sure that your agent is powered on. If it isn't, power it on.

If the agent is powered on and the task is still stuck in Launching status, then a network connectivity problem between the agent and DataSync is the most likely issue. Check your network and firewall settings to make sure that the agent can connect to DataSync.

If you perform these actions and the issue isn't resolved, open a support channel and engage AWS Support. For information about how to open a support channel, see Enabling AWS Support to Help Troubleshoot Your Running On-Premises Agent (p. 93).

Your Task Execution Fails with a Permissions Denied Error Message

You can get a "permissions denied" error message if you configure your NFS server with **root_squash** or **all_squash** enabled and your files don't have all read access.
Preparing Status for a Task Execution Takes Longer Than Expected to Complete

The time DataSync spends in the `PREPARING` status depends on the number of files in both the source and destination file systems, and the performance of these file systems. When a task starts, DataSync performs a recursive directory listing to discover all files and file metadata in the source and destination file system. These listings are used to identify differences and determine what to copy. This process usually takes between a few minutes to a few hours. For more information, see Starting a Task (p. 57).

**Action to Take**

You don't need to take any action. Wait for the `PREPARING` status to complete and status changes to `TRANSFERRING`. If the status doesn't change to `TRANSFERRING` status, contact AWS Support.

Verifying Status for a Task Execution Takes Longer Than Expected to Complete

The time DataSync spends in the `VERIFYING` status depends on a number of factors. These are the number of files, the total size of all files in the source and destination file systems, and the performance of these file systems. By default, Verification mode is enabled in the options setting. The verification DataSync performs includes a SHA256 checksum on all file content and an exact comparison of all file metadata.

**Action to Take**

You don't need to take any action. Wait for the `VERIFYING` status to complete. If the `VERIFYING` status doesn't complete, contact AWS Support.

Your Storage Cost Is Higher Than Expected

If your storage cost is higher than expected, it might be due to one or more of the following reasons:

- DataSync uses the Amazon S3 multipart upload feature to upload objects to Amazon S3. This approach can result in unexpected storage charges for uploads that don't successfully complete.
- Object versioning might be enabled on your S3 bucket. Object versioning results in Amazon S3 storing multiple copies of objects that have the same name.
Action to Take

In these cases, you can take the following steps:

- If the issue relates to multipart uploads, configure a policy for multipart uploads for your S3 bucket to clean up incomplete multipart uploads to reduce storage cost. For more information, see the AWS blog post [S3 Lifecycle Management Update – Support for Multipart Uploads and Delete Markers](https://aws.amazon.com/blogs/aws/s3-lifecycle-management-update-support-for-multipart-uploads-and-delete-markers/).
- If the issue relates to object versioning, verify whether object versioning is enabled for your Amazon S3 bucket. If versioning is enabled, turn it off.

If you perform these actions and the issue isn't resolved, contact AWS Support. For information about how to contact AWS Support, see [Getting Started with AWS Support](https://aws.amazon.com/support/).

How to Enable AWS Support to Help Troubleshoot Your Running On-Premises Agent

You can allow AWS Support to access your agent and assist you with troubleshooting agent issues. By default, AWS Support access to your DataSync is not enabled. You enable this access through the host's local console. To give AWS Support access to your DataSync, you first log in to the local console for the host then connect to the support server.

For instructions on how to open a support channel, see [Enabling AWS Support to Help Troubleshoot Your Running On-Premises Agent](https://aws.amazon.com/support/).
Troubleshooting an EC2 Agent

Following, you can find information on how to troubleshoot EC2 agent issues.

Enabling AWS Support to Help Troubleshoot Your DataSync Running on Amazon EC2

DataSync provides a local console that you can use to perform several maintenance tasks. These tasks include enabling AWS Support to access your DataSync agent to assist you with troubleshooting. By default, AWS Support access to your agent is disabled.

You enable this access through the EC2 local console. You log in to the EC2 local console through Secure Shell (SSH). To successfully log in through SSH, your instance's security group must have a rule that opens TCP port 22.

**Note**

If you add a new rule to an existing security group, the new rule applies to all instances that use that security group. For more information about security groups and how to add a security group rule, see Amazon EC2 Security Groups for Linux Instances in the Amazon EC2 User Guide for Linux Instances.

Connecting to the EC2 Agent's Local Console

Log in to the local console for your DataSync EC2 agent. For instructions, see Connect to Your Instance in the Amazon EC2 User Guide for Linux Instances.

You can use the following command to log in to the EC2 instance's local console. The user name is admin.

```
ssh --i PRIVATE-KEY admin@AGENT-PUBLIC-DNS-NAME
```

In this command, the `PRIVATE-KEY` value is the .pem file containing the private certificate of the EC2 key pair that you used to launch the EC2 instance. For more information, see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html#retrieving-the-public-keyin the Amazon EC2 User Guide for Linux Instances.

The `AGENT-PUBLIC-DNS-NAME` value is the public DNS name of your agent. You get this public DNS name by choosing the EC2 instance in the EC2 console and choosing the Description tab.

For instructions about using the local console menu to enable AWS Support access to your agent and to perform other operations, see Configuring Your Agent on the VM Local Console (p. 84).
AWS DataSync Quotas

Following, you can find information on AWS DataSync resources and their quotas.

Topics
- Quotas for Tasks (p. 119)
- Quotas for Task Executions (p. 119)
- Quotas for DataSync Filters (p. 120)

Quotas for Tasks

Following are the quotas on tasks for each customer account in an AWS Region.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of tasks you can create in account per AWS Region</td>
<td>100</td>
</tr>
<tr>
<td>Maximum number of files per task</td>
<td>50 million</td>
</tr>
<tr>
<td>Important For tasks that transfer more than 20 million files, make sure that you allocate a minimum of 64 GB of RAM to the VM. For minimum resource requirements for DataSync, see Virtual Machine Requirements (p. 9).</td>
<td></td>
</tr>
<tr>
<td>Maximum throughput per task</td>
<td>10 Gbps</td>
</tr>
</tbody>
</table>

Note
You can make a request for these quotas to be increased.

You can take the following steps to request an increase for these quotas. These increases are not granted right away, so it might take a couple of days for your increase to take effect.

To request a quotas increase
1. Open the AWS Support Center page, sign in if necessary, and then choose Create Case.
2. For Regarding, choose Service Limit Increase.
3. For Limit Type, choose the type of quotas to increase, fill in the necessary fields in the form, and then choose your preferred method of contact.

Quotas for Task Executions

Following are the quotas on tasks executions for each customer account in an AWS Region.
Quotas for DataSync Filters

Following are the quotas on DataSync filters per task or task execution.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of characters in a filter string</td>
<td>100,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of days task execution history is retained</td>
<td>30</td>
</tr>
</tbody>
</table>

Quotas for DataSync Filters

Following are the quotas on DataSync filters per task or task execution.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of characters in a filter string</td>
<td>100,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of days task execution history is retained</td>
<td>30</td>
</tr>
</tbody>
</table>
Additional Resources

In this section, you can find additional information about and resources for AWS DataSync.

Topics
- Transferring Data from an On-Premises Storage Array (p. 121)
- Other Use Cases (p. 121)

Transferring Data from an On-Premises Storage Array

You might want to transfer data from an on-premises enterprise storage array to Amazon EFS. In this case, files in the source file system might be modified by another application while the files are being transferred from Network File System (NFS) or Server Message Block (SMB) file share to Amazon EFS.

To ensure that DataSync successfully performs a transfer with full consistency verification, we recommend that the source location point to a read-only snapshot. This setup ensures that files at the source location can't be modified while the files are being transferred, and makes sure that verification works.

For information about how to take a snapshot in an enterprise storage array, see one of the following:
- EMC VNX: How to create a VNX snapshot and attach it to a server
- EMC VMAX: EMC TimeFinder Product Description Guide
- NetApp: Snapshot management
- HPE 3PAR: Snapshots and copy data management
- HDS: Hitachi Copy-on-Write Snapshot User Guide

Other Use Cases

In this section, you can find information about use cases that are not common to most users.

Topics
- Transferring Files in Opposite Directions (p. 121)
- Using Multiple Tasks to Write to the Same Amazon S3 Bucket (p. 122)
- Allowing Amazon S3 Access From a Private VPC Endpoint (p. 122)

Transferring Files in Opposite Directions

Transferring data in opposite directions allows for workflows where the active application moves between locations. AWS DataSync doesn't support workflows where multiple active applications write to both locations at the same time. Use the steps in the following procedure to configure DataSync to transfer data in opposite directions.

To configure DataSync to data transfers in opposite directions

1. Create a location and name it Location A.
2. Create a second location and name it **Location B**.
3. Create a task, name it **Task A-B**, and then configure **Location A** as the source location and **Location B** as the destination location.
4. Create a second task, name it **Task B-A**, and then configure **Location B** as the source location and **Location A** as the destination location.
5. To update **Location B** with data from **Location A**, run **Task A-B**.

To update **Location A** with data from **Location B**, run **Task B-A**.

Don't run these two tasks concurrently. DataSync can transfer files in opposite directions periodically. However, it doesn't support workflows where multiple active applications write to both **Location A** and **Location B** simultaneously.

### Using Multiple Tasks to Write to the Same Amazon S3 Bucket

In certain use cases, you might want different tasks to write to the same Amazon S3 bucket. In this case, you create different folders in the S3 bucket for each of the task. This approach prevents file name conflicts between the tasks, and also means that you can set different permissions for each of folders.

For example, you might have three tasks: `task1`, `task2`, and `task3` write to an S3 bucket named `MyBucket`.

You create three folders in the bucket:

- `s3://MyBucket/task1`
- `s3://MyBucket/task2`
- `s3://MyBucket/task3`

For each task, you choose the folder in `MyBucket` that corresponds to the task as the destination, and set different permissions for each of the three folders.

### Allowing Amazon S3 Access From a Private VPC Endpoint

In certain cases, you might want to only allow Amazon S3 access from a private endpoint. In that case, you create an IAM policy that allows that access and attach it to the S3 bucket. If you need a policy that restricts your S3 bucket's access to DataSync VPC endpoints, contact [AWS DataSync Support](https://www.aws.amazon.com/datasync) to get the DataSync VPC endpoint for your AWS Region.

The following is a sample policy that only allows Amazon S3 access from a private endpoint.

```json
{
  "Version": "2012-10-17",
  "Id": "Policy1415115909152",
  "Statement": [
    {
      "Sid": "Access-to-specific-VPCE-only",
      "Principal": "",
      "Action": "s3:",
      "Effect": "Deny",
      "Resource": ["arn:aws:s3:::examplebucket",
                   "arn:aws:s3:::examplebucket/*"]
    }
  ]
}
```
"Condition": {
  "StringNotEquals": {
    "aws:sourceVpce": "vpce-your vpc endpoint",
    "aws:sourceVpce": "vpce-DataSync vpc endpoint for your region"
  }
}

For more information, see Example Bucket Policies for VPC Endpoints for Amazon S3 in the Amazon Simple Storage Service Developer Guide.
In addition to using the console, you can use the AWS DataSync API to programmatically configure and manage DataSync and its resources. This section describes the AWS DataSync operations and data types and contains the API Reference documentation for AWS DataSync.

Topics
- Actions (p. 124)
- Data Types (p. 214)
- Common Errors (p. 235)
- Common Parameters (p. 237)

Actions

The following actions are supported:

- CancelTaskExecution (p. 126)
- CreateAgent (p. 128)
- CreateLocationEfs (p. 132)
- CreateLocationFsxWindows (p. 136)
- CreateLocationNfs (p. 139)
- CreateLocationS3 (p. 143)
- CreateLocationSmb (p. 148)
- CreateTask (p. 152)
- DeleteAgent (p. 157)
- DeleteLocation (p. 159)
- DeleteTask (p. 161)
- DescribeAgent (p. 163)
- DescribeLocationEfs (p. 166)
- DescribeLocationFsxWindows (p. 169)
- DescribeLocationNfs (p. 172)
- DescribeLocationS3 (p. 175)
- DescribeLocationSmb (p. 178)
- DescribeTask (p. 181)
- DescribeTaskExecution (p. 186)
- ListAgents (p. 191)
- ListLocations (p. 193)
- ListTagsForResource (p. 195)
- ListTaskExecutions (p. 198)
- ListTasks (p. 201)
- StartTaskExecution (p. 203)
- TagResource (p. 206)
- UntagResource (p. 208)
- UpdateAgent (p. 210)
• UpdateTask (p. 212)
CancelTaskExecution

Cancels execution of a task.

When you cancel a task execution, the transfer of some files is abruptly interrupted. The contents of files that are transferred to the destination might be incomplete or inconsistent with the source files. However, if you start a new task execution on the same task and you allow the task execution to complete, file content on the destination is complete and consistent. This applies to other unexpected failures that interrupt a task execution. In all of these cases, AWS DataSync successfully complete the transfer when you start the next task execution.

Request Syntax

```
{
   "TaskExecutionArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**TaskExecutionArn (p. 126)**

The Amazon Resource Name (ARN) of the task execution to cancel.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateAgent

Activates an AWS DataSync agent that you have deployed on your host. The activation process associates your agent with your account. In the activation process, you specify information such as the AWS Region that you want to activate the agent in. You activate the agent in the AWS Region where your target locations (in Amazon S3 or Amazon EFS) reside. Your tasks are created in this AWS Region.

You can activate the agent in a VPC (virtual private cloud) or provide the agent access to a VPC endpoint so you can run tasks without going over the public Internet.

You can use an agent for more than one location. If a task uses multiple agents, all of them need to have status AVAILABLE for the task to run. If you use multiple agents for a source location, the status of all the agents must be AVAILABLE for the task to run.

For more information, see Activating an Agent in the AWS DataSync User Guide.

Agents are automatically updated by AWS on a regular basis, using a mechanism that ensures minimal interruption to your tasks.

Request Syntax

```json
{
   "ActivationKey": "string",
   "AgentName": "string",
   "SecurityGroupArns": [ "string" ],
   "SubnetArns": [ "string" ],
   "Tags": [ {
      "Key": "string",
      "Value": "string"
   } ],
   "VpcEndpointId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**ActivationKey** (p. 128)

Your agent activation key. You can get the activation key either by sending an HTTP GET request with redirects that enable you to get the agent IP address (port 80). Alternatively, you can get it from the AWS DataSync console.

The redirect URL returned in the response provides you the activation key for your agent in the query string parameter activationKey. It might also include other activation-related parameters; however, these are merely defaults. The arguments you pass to this API call determine the actual configuration of your agent.

For more information, see Activating an Agent in the AWS DataSync User Guide.

Type: String

Length Constraints: Maximum length of 29.
CreateAgent

Pattern: `[A-Z0-9]{5}(-[A-Z0-9]{5}){4}`

Required: Yes

**AgentName (p. 128)**

The name you configured for your agent. This value is a text reference that is used to identify the agent in the console.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `^[a-zA-Z0-9\s+=._:@/-]+$`

Required: No

**SecurityGroupArns (p. 128)**

The ARNs of the security groups used to protect your data transfer task subnets. See CreateAgent:SubnetArns (p. 129).

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]*:[0-9]{12}:security-group/.*$`

Required: No

**SubnetArns (p. 128)**

The Amazon Resource Names (ARNs) of the subnets in which DataSync will create elastic network interfaces for each data transfer task. The agent that runs a task must be private. When you start a task that is associated with an agent created in a VPC, or one that has access to an IP address in a VPC, then the task is also private. In this case, DataSync creates four network interfaces for each task in your subnet. For a data transfer to work, the agent must be able to route to all these four network interfaces.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]*:[0-9]{12}:subnet/.*$`

Required: No

**Tags (p. 128)**

The key-value pair that represents the tag that you want to associate with the agent. The value can be an empty string. This value helps you manage, filter, and search for your agents.

**Note**

Valid characters for key and value are letters, spaces, and numbers representable in UTF-8 format, and the following special characters: `+ - = . _ : / @`

Type: Array of TagListEntry (p. 230) objects
Array Members: Minimum number of 0 items. Maximum number of 55 items.

Required: No

VpcEndpointId (p. 128)

The ID of the VPC (virtual private cloud) endpoint that the agent has access to. This is the client-side VPC endpoint, also called a PrivateLink. If you don't have a PrivateLink VPC endpoint, see Creating a VPC Endpoint Service Configuration in the Amazon VPC User Guide.

VPC endpoint ID looks like this: vpce-01234d5aff67890e1.

Type: String

Pattern: ^vpce-[0-9a-f]{17}$

Required: No

Response Syntax

```json
{
   "AgentArn": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

AgentArn (p. 130)

The Amazon Resource Name (ARN) of the agent. Use the ListAgents operation to return a list of agents for your account and AWS Region.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400
Examples

Example

The following example creates an agent and activates it using an activation key.

Sample Request

```json
{
    "ActivationKey": "AAAAA-7AAAA-GG7MC-3I9R3-27COD",
    "AgentName": "MyAgent",
    "Tags": [
        {
            "Key": "Job",
            "Value": "TransferJob-1"
        }
    ]
}
```

Example

The response returns the Amazon Resource Name (ARN) of the activated agent.

Sample Response

```json
{
    "AgentArn": "arn:aws:datasync:us-east-2:11122233444:agent/agent-0b0addbeef44baca3"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateLocationEfs

Creates an endpoint for an Amazon EFS file system.

Request Syntax

```json
{
  "Ec2Config": {
    "SecurityGroupArns": [ "string" ],
    "SubnetArn": "string"
  },
  "EfsFileSystemArn": "string",
  "Subdirectory": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**Ec2Config (p. 132)**

The subnet and security group that the Amazon EFS file system uses. The security group that you provide needs to be able to communicate with the security group on the mount target in the subnet specified.

The exact relationship between security group M (of the mount target) and security group S (which you provide for DataSync to use at this stage) is as follows:

- Security group M (which you associate with the mount target) must allow inbound access for the Transmission Control Protocol (TCP) on the NFS port (2049) from security group S. You can enable inbound connections either by IP address (CIDR range) or security group.

- Security group S (provided to DataSync to access EFS) should have a rule that enables outbound connections to the NFS port on one of the file system's mount targets. You can enable outbound connections either by IP address (CIDR range) or security group.

For information about security groups and mount targets, see [Security Groups for Amazon EC2 Instances and Mount Targets](#) in the *Amazon EFS User Guide*.

Type: Ec2Config (p. 216) object

Required: Yes

**EfsFileSystemArn (p. 132)**

The Amazon Resource Name (ARN) for the Amazon EFS file system.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):elasticfilesystem:[a-z\-0-9]*:[0-9]{12}:file-system/fs-.*$
Subdirectory (p. 132)

A subdirectory in the location's path. This subdirectory in the EFS file system is used to read data from the EFS source location or write data to the EFS destination. By default, AWS DataSync uses the root directory.

**Note**

Subdirectory must be specified with forward slashes. For example, `/path/to/folder`.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: `^[a-zA-Z0-9_\-+./\(\)\p{Zs}]*$`

Required: No

Tags (p. 132)

The key-value pair that represents a tag that you want to add to the resource. The value can be an empty string. This value helps you manage, filter, and search for your resources. We recommend that you create a name tag for your location.

Type: Array of TagListEntry (p. 230) objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

Required: No

Response Syntax

```json
{
  "LocationArn": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

LocationArn (p. 133)

The Amazon Resource Name (ARN) of the Amazon EFS file system location that is created.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).
InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example creates an endpoint for an Amazon EFS file system.

Sample Request

```
{
    "Ec2Config": {
        "SubnetArn": "arn:aws:ec2:us-east-2:11122233344:subnet/subnet-f45a0e678",
    },
    "Subdirectory": "/MySubdirectory",
    "Tags": [
        {
            "Key": "Name",
            "Value": "ElasticFileSystem-1"
        }
    ]
}
```

Example

The response returns the Amazon Resource Name (ARN) of the EFS location.

Sample Response

```
{
    "LocationArn": "arn:aws:datsync:us-east-2:111222333444:location/loc-07db7abfc326c50fb"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
CreateLocationFsxWindows

Creates an endpoint for an Amazon FSx for Windows file system.

Request Syntax

```json
{
  "Domain": "string",
  "FsxFilesystemArn": "string",
  "Password": "string",
  "SecurityGroupArns": [ "string" ],
  "Subdirectory": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "User": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**Domain (p. 136)**

The name of the Windows domain that the FSx for Windows server belongs to.

Type: String

Length Constraints: Maximum length of 253.

Pattern: ^([A-Za-z0-9]+[A-Za-z0-9-]*[A-Za-z0-9]*[A-Za-z0-9-]*[A-Za-z0-9]+$

Required: No

**FsxFilesystemArn (p. 136)**

The Amazon Resource Name (ARN) for the FSx for Windows file system.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-isog|aws-iso|aws-iso-b):fsx:[a-z\-0-9]*[0-9]{12}:file-system/fs-.*$

Required: Yes

**Password (p. 136)**

The password of the user who has the permissions to access files and folders in the FSx for Windows file system.

Type: String

Length Constraints: Maximum length of 104.
Pattern: ^\{0,104}\$

Required: Yes

**SecurityGroupArns (p. 136)**

The Amazon Resource Names (ARNs) of the security groups that are to use to configure the FSx for Windows file system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]*:[0-9]{12}:security-group/.*$

Required: Yes

**Subdirectory (p. 136)**

A subdirectory in the location's path. This subdirectory in the Amazon FSx for Windows file system is used to read data from the Amazon FSx for Windows source location or write data to the FSx for Windows destination.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[a-zA-Z0-9\-\_\-/\(\)\$\p{Zs}]+$  

Required: No

**Tags (p. 136)**

The key-value pair that represents a tag that you want to add to the resource. The value can be an empty string. This value helps you manage, filter, and search for your resources. We recommend that you create a name tag for your location.

Type: Array of TagListEntry (p. 230) objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

Required: No

**User (p. 136)**

The user who has the permissions to access files and folders in the FSx for Windows file system.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^[^\x5B\x5D\x5D\x5D\x5D::=,+*?]{1,104}$

Required: Yes

**Response Syntax**

```json
{
  "LocationArn": "string"
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LocationArn** *(p. 137)*

The Amazon Resource Name (ARN) of the FSx for Windows file system location that is created.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$`

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 235)](#).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
CreateLocationNfs

Defines a file system on a Network File System (NFS) server that can be read from or written to.

Request Syntax

```json
{
    "MountOptions": {
        "Version": "string"
    },
    "OnPremConfig": {
        "AgentArns": [ "string" ]
    },
    "ServerHostname": "string",
    "Subdirectory": "string",
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }
    ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

MountOptions (p. 139)

The NFS mount options that DataSync can use to mount your NFS share.

Type: NfsMountOptions (p. 220) object

Required: No

OnPremConfig (p. 139)

Contains a list of Amazon Resource Names (ARNs) of agents that are used to connect to an NFS server.

Type: OnPremConfig (p. 221) object

Required: Yes

ServerHostname (p. 139)

The name of the NFS server. This value is the IP address or Domain Name Service (DNS) name of the NFS server. An agent that is installed on-premises uses this host name to mount the NFS server in a network.

Note

This name must either be DNS-compliant or must be an IP version 4 (IPv4) address.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^((a-zA-Z0-9\-\[a-zA-Z0-9\])\.)*((A-Za-z0-9\-\[A-Za-z0-9\])$
**Subdirectory (p. 139)**

The subdirectory in the NFS file system that is used to read data from the NFS source location or write data to the NFS destination. The NFS path should be a path that's exported by the NFS server, or a subdirectory of that path. The path should be such that it can be mounted by other NFS clients in your network.

To see all the paths exported by your NFS server, run "showmount -e nfs-server-name" from an NFS client that has access to your server. You can specify any directory that appears in the results, and any subdirectory of that directory. Ensure that the NFS export is accessible without Kerberos authentication.

To transfer all the data in the folder you specified, DataSync needs to have permissions to read all the data. To ensure this, either configure the NFS export with `no_root_squash`, or ensure that the permissions for all of the files that you want DataSync allow read access for all users. Doing either enables the agent to read the files. For the agent to access directories, you must additionally enable all execute access.

For information about NFS export configuration, see 18.7. The `/etc/exports` Configuration File in the Red Hat Enterprise Linux documentation.

**Tags (p. 139)**

The key-value pair that represents the tag that you want to add to the location. The value can be an empty string. We recommend using tags to name your resources.

**Response Syntax**

```json
{
   "LocationArn": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LocationArn (p. 140)**

The Amazon Resource Name (ARN) of the source NFS file system location that is created.

Type: String
Length Constraints: Maximum length of 128.

Pattern: \^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}\$

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example creates an endpoint for an NFS file system using the specified NFS version as a mount option.

**Sample Request**

```json
{
  "MountOptions": {
    "Version": "NFS4_0"
  },
  "OnPremConfig": {
    "AgentArn": ["arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44b3nfs" ],
    "ServerHostname": "MyServer@amazon.com",
    "Subdirectory": "/MyFolder",
    "Tags": [
      { "Key": "Name",
        "Value": "ElasticFileSystem-1"
      }
    ]
  }
}
```

**Example**

The response returns the Amazon Resource Name (ARN) of the NFS location.

**Sample Response**

```json
{
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location[loc-07db7abfc326c50aa"
}
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateLocationS3

CreateLocationS3

Creates an endpoint for an Amazon S3 bucket.

For AWS DataSync to access a destination S3 bucket, it needs an AWS Identity and Access Management (IAM) role that has the required permissions. You can set up the required permissions by creating an IAM policy that grants the required permissions and attaching the policy to the role. An example of such a policy is shown in the examples section.

For more information, see Configuring Amazon S3 Location Settings in the AWS DataSync User Guide.

Request Syntax

```json
{
  "S3BucketArn": "string",
  "S3Config": {
    "BucketAccessRoleArn": "string"
  },
  "S3StorageClass": "string",
  "Subdirectory": "string",
  "Tags": [
    { "Key": "string", "Value": "string" }
  ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

S3BucketArn (p. 143)

The Amazon Resource Name (ARN) of the Amazon S3 bucket.

Type: String

Length Constraints: Maximum length of 76.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3:::([^/]*$)

Required: Yes

S3Config (p. 143)

The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that is used to access an Amazon S3 bucket.

For detailed information about using such a role, see Creating a Location for Amazon S3 in the AWS DataSync User Guide.

Type: S3Config (p. 228) object

Required: Yes
S3StorageClass (p. 143)

The Amazon S3 storage class that you want to store your files in when this location is used as a task destination. For more information about S3 storage classes, see Amazon S3 Storage Classes in the Amazon Simple Storage Service Developer Guide. Some storage classes have behaviors that can affect your S3 storage cost. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

Type: String

Valid Values: STANDARD | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE

Required: No

Subdirectory (p. 143)

A subdirectory in the Amazon S3 bucket. This subdirectory in Amazon S3 is used to read data from the S3 source location or write data to the S3 destination.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[a-zA-Z0-9-\._/-]+\./\(\)\p{Zs}]*$

Required: No

Tags (p. 143)

The key-value pair that represents the tag that you want to add to the location. The value can be an empty string. We recommend using tags to name your resources.

Type: Array of TagListEntry (p. 230) objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

Required: No

Response Syntax

```json
{
  "LocationArn": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

LocationArn (p. 144)

The Amazon Resource Name (ARN) of the source Amazon S3 bucket location that is created.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:location/loc-[0-9a-z]{17}$
Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

The following example shows the simplest policy that grants the required permissions for AWS DataSync to access a destination Amazon S3 bucket and attaches it to an IAM role.

Step 1. Allow AWS DataSync to assume the IAM role required to write to the bucket

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

```json
"Role": {
    "Path": "/",
    "RoleName": "MyBucketAccessRole",
    "RoleId": "role-id",
    "Arn": "arn:aws:iam::account-id:role/MyBucketAccessRole",
    "CreateDate": "2018-07-27T02:49:23.117Z",
    "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Principal": {
                    "Service": "datasync.amazonaws.com"
                },
                "Action": "sts:AssumeRole"
            }
        ]
    }
}
```
Step 2. Allow the created IAM role to write to the bucket

Attach a policy that has sufficient permissions to access the bucket to the role. An example of such policy is the AWSDataSyncFullAccess managed policy.

For more information, see AWSDataSyncFullAccess in the IAM console.

You don't need to create this policy. It's managed by AWS, so all that you need to do is specify its ARN in the attach-role-policy command.

IAM_POLICY_ARN='arn:aws:iam::aws:policy/AWSDataSyncFullAccess'

Examples

Example

The following example creates an endpoint for an Amazon S3 bucket.

Sample Request

```json
{
  "S3BucketArn": "arn:aws:s3:::MyBucket",
  "S3Config": {
    "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/MyBucketAccessRole",
  },
  "S3StorageClass": "STANDARD",
  "Subdirectory": "/MyFolder",
  "Tags": [
    {
      "Key": "Name",
      "Value": "s3Bucket-1"
    }
  ]
}
```

Example

The following response returns the Amazon Resource Name (ARN) for the Amazon S3 location.

Sample Response

```json
{
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50s3"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateLocationSmb

Defines a file system on a Server Message Block (SMB) server that can be read from or written to.

**Request Syntax**

```json
{
   "AgentArns": [ "string" ],
   "Domain": "string",
   "MountOptions": {
      "Version": "string"
   },
   "Password": "string",
   "ServerHostname": "string",
   "Subdirectory": "string",
   "Tags": [
      { "Key": "string", "Value": "string" }
   ],
   "User": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**AgentArns (p. 148)**

The Amazon Resource Names (ARNs) of agents to use for a Simple Message Block (SMB) location.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 64 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: Yes

**Domain (p. 148)**

The name of the Windows domain that the SMB server belongs to.

Type: String

Length Constraints: Maximum length of 253.

Pattern: ^([A-Za-z0-9]+[A-Za-z0-9-\.]*)*[A-Za-z0-9-]*[A-Za-z0-9]$  

Required: No

**MountOptions (p. 148)**

The mount options used by DataSync to access the SMB server.

Type: SmbMountOptions (p. 229) object
Required: No

Password (p. 148)

The password of the user who can mount the share, has the permissions to access files and folders in the SMB share.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^.{0,104}$

Required: Yes

ServerHostname (p. 148)

The name of the SMB server. This value is the IP address or Domain Name Service (DNS) name of the SMB server. An agent that is installed on-premises uses this hostname to mount the SMB server in a network.

Note

This name must either be DNS-compliant or must be an IP version 4 (IPv4) address.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^((((a-zA-Z0-9-\*][a-zA-Z0-9-\*])\.)*)\*([A-Za-z0-9-\*][A-Za-z0-9-\*])$\n
Required: Yes

Subdirectory (p. 148)

The subdirectory in the SMB file system that is used to read data from the SMB source location or write data to the SMB destination. The SMB path should be a path that's exported by the SMB server, or a subdirectory of that path. The path should be such that it can be mounted by other SMB clients in your network.

Note

Subdirectory must be specified with forward slashes. For example, /path/to/folder.

To transfer all the data in the folder you specified, DataSync needs to have permissions to mount the SMB share, as well as to access all the data in that share. To ensure this, either ensure that the user/password specified belongs to the user who can mount the share, and who has the appropriate permissions for all of the files and directories that you want DataSync to access, or use credentials of a member of the Backup Operators group to mount the share. Doing either enables the agent to access the data. For the agent to access directories, you must additionally enable all execute access.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[a-zA-Z0-9-_\+\./\(\)\$\p\{Zs\}]$+

Required: Yes

Tags (p. 148)

The key-value pair that represents the tag that you want to add to the location. The value can be an empty string. We recommend using tags to name your resources.

Type: Array of TagListEntry (p. 230) objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

Required: No
User (p. 148)
The user who can mount the share, has the permissions to access files and folders in the SMB share.
Type: String
Length Constraints: Maximum length of 104.
Pattern: ^[^\x5B\x5D\/:;|=,\+\*?]{1,104}#
Required: Yes

Response Syntax

```
{
   "LocationArn": "string"
}
```

Response Elements
If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

LocationArn (p. 150)
The Amazon Resource Name (ARN) of the source SMB file system location that is created.
Type: String
Length Constraints: Maximum length of 128.
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}#

Errors
For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException
This exception is thrown when an error occurs in the AWS DataSync service.
HTTP Status Code: 500

InvalidRequestException
This exception is thrown when the client submits a malformed request.
HTTP Status Code: 400

Examples
Example
Sample Request
```
{
}
```
"AgentArns": [
    agent-0b0addbeef44b3nfs,arn:aws:datasync:us-east-2:111222333444:agent/
    agent-2345noo35nnee1123ovo3",
    "Domain": "AMAZON",
    "MountOptions": {
        "Version": "SMB3"
    },
    "Password": "string",
    "ServerHostname": "MyServer.amazon.com",
    "Subdirectory": "share",
    "Tags": [
        {
            "Key": "department",
            "Value": "finance"
        }
    ],
    "User": "user-1"
}

Example

Sample Response

{"arn:aws:datasync:us-east-1:111222333444:location/loc-0f01451b40b2af49"}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateTask

Creates a task. A task is a set of two locations (source and destination) and a set of Options that you use to control the behavior of a task. If you don’t specify Options when you create a task, AWS DataSync populates them with service defaults.

When you create a task, it first enters the CREATING state. During CREATING AWS DataSync attempts to mount the on-premises Network File System (NFS) location. The task transitions to the AVAILABLE state without waiting for the AWS location to become mounted. If required, AWS DataSync mounts the AWS location before each task execution.

If an agent that is associated with a source (NFS) location goes offline, the task transitions to the UNAVAILABLE status. If the status of the task remains in the CREATING status for more than a few minutes, it means that your agent might be having trouble mounting the source NFS file system. Check the task’s ErrorCode and ErrorDetail. Mount issues are often caused by either a misconfigured firewall or a mistyped NFS server host name.

Request Syntax

```json
{
    "CloudWatchLogGroupArn": "string",
    "DestinationLocationArn": "string",
    "Excludes": [
        {
            "FilterType": "string",
            "Value": "string"
        }
    ],
    "Name": "string",
    "Options": {
        "Atime": "string",
        "BytesPerSecond": number,
        "Gid": "string",
        "LogLevel": "string",
        "Mtime": "string",
        "OverwriteMode": "string",
        "PosixPermissions": "string",
        "PreserveDeletedFiles": "string",
        "PreserveDevices": "string",
        "TaskQueueing": "string",
        "Uid": "string",
        "VerifyMode": "string"
    },
    "Schedule": {
        "ScheduleExpression": "string"
    },
    "SourceLocationArn": "string",
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }
    ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).
The request accepts the following data in JSON format.

**CloudWatchLogGroupArn (p. 152)**

The Amazon Resource Name (ARN) of the Amazon CloudWatch log group that is used to monitor and log events in the task.

For more information about how to use CloudWatch Logs with DataSync, see Monitoring Your Task in the *AWS DataSync User Guide*.

For more information about these groups, see Working with Log Groups and Log Streams in the *Amazon CloudWatch Logs User Guide*.

Type: String

Length Constraints: Maximum length of 562.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):logs:[a-z\-0-9]*:[0-9]{12}:log-group:(\[^:\*\]*)$`

Required: No

**DestinationLocationArn (p. 152)**

The Amazon Resource Name (ARN) of an AWS storage resource's location.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$`

Required: Yes

**Excludes (p. 152)**

A list of filter rules that determines which files to exclude from a task. The list should contain a single filter string that consists of the patterns to exclude. The patterns are delimited by "|" (that is, a pipe), for example, "/folder1|/folder2"

Type: Array of FilterRule (p. 217) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

**Name (p. 152)**

The name of a task. This value is a text reference that is used to identify the task in the console.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `^[a-zA-Z0-9-\s+=._:@/-]+$`

Required: No

**Options (p. 152)**

The set of configuration options that control the behavior of a single execution of the task that occurs when you call StartTaskExecution. You can configure these options to preserve metadata such as user ID (UID) and group ID (GID), file permissions, data integrity verification, and so on.
For each individual task execution, you can override these options by specifying the `OverrideOptions` before starting the task execution. For more information, see the `StartTaskExecution (p. 203)` operation.

**Type:** Options (p. 222) object

**Required:** No

**Schedule (p. 152)**

Specifies a schedule used to periodically transfer files from a source to a destination location. The schedule should be specified in UTC time. For more information, see Scheduling Your Task (p. 61).

**Type:** TaskSchedule (p. 235) object

**Required:** No

**SourceLocationArn (p. 152)**

The Amazon Resource Name (ARN) of the source location for the task.

**Type:** String

**Length Constraints:** Maximum length of 128.

**Pattern:** `arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$`

**Required:** Yes

**Tags (p. 152)**

The key-value pair that represents the tag that you want to add to the resource. The value can be an empty string.

**Type:** Array of TagListEntry (p. 230) objects

**Array Members:** Minimum number of 0 items. Maximum number of 55 items.

**Required:** No

**Response Syntax**

```
{
  "TaskArn": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**TaskArn (p. 154)**

The Amazon Resource Name (ARN) of the task.

**Type:** String

**Length Constraints:** Maximum length of 128.

**Pattern:** `arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:task/task-[0-9a-f]{17}$`
Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example creates a task using a source and destination locations.

**Sample Request**

```json
{
    "Options": {
        "Atime": "BEST_EFFORT",
        "Gid": "NONE",
        "Mtime": "PRESERVE",
        "PosixPermissions": "PRESERVE",
        "PreserveDevices": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "Uid": "NONE",
        "VerifyMode": "POINT_IN_TIME_CONSISTENT",
    },
    "Schedule": {
        "ScheduleExpression": "0 12 ? * SUN,WED *"
    },
    "DestinationLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50fb",
    "Name": "MyTask",
    "SourceLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-0f01451b140b2af49",
    "Tags": [
        {
            "Key": "Name",
            "Value": "Task-1"
        }
    ]
}
```

**Example**

The following response returns the Amazon Resource Name (ARN) of the task.

**Sample Response**

```json
{
    "TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e66977f960f26"
}
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteAgent

Deletes an agent. To specify which agent to delete, use the Amazon Resource Name (ARN) of the agent in your request. The operation disassociates the agent from your AWS account. However, it doesn't delete the agent virtual machine (VM) from your on-premises environment.

Request Syntax

```json
{
   "AgentArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**AgentArn** (p. 157)

The Amazon Resource Name (ARN) of the agent to delete. Use the ListAgents operation to return a list of agents for your account and AWS Region.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteLocation

Deletes the configuration of a location used by AWS DataSync.

Request Syntax

```json
{
  "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**LocationArn (p. 159)**

The Amazon Resource Name (ARN) of the location to delete.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteTask

Deletes a task.

Request Syntax

```json
{
   "TaskArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**TaskArn (p. 161)**

The Amazon Resource Name (ARN) of the task to delete.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:0-9\{12\}:task/task-[0-9a-f\{17\}$

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
AWS SDK for C++
AWS SDK for Go
AWS SDK for Java
AWS SDK for JavaScript
AWS SDK for PHP V3
AWS SDK for Python
AWS SDK for Ruby V3
DescribeAgent

Returns metadata such as the name, the network interfaces, and the status (that is, whether the agent is running or not) for an agent. To specify which agent to describe, use the Amazon Resource Name (ARN) of the agent in your request.

Request Syntax

```json
{
  "AgentArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**AgentArn (p. 163)**

The Amazon Resource Name (ARN) of the agent to describe.

- **Type:** String
- **Length Constraints:** Maximum length of 128.
- **Pattern:** ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$
- **Required:** Yes

Response Syntax

```json
{
  "AgentArn": "string",
  "CreationTime": number,
  "EndpointType": "string",
  "LastConnectionTime": number,
  "Name": "string",
  "PrivateLinkConfig": {
    "PrivateLinkEndpoint": "string",
    "SecurityGroupArns": [ "string" ],
    "SubnetArns": [ "string" ],
    "VpcEndpointId": "string"
  },
  "Status": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**AgentArn (p. 163)**

The Amazon Resource Name (ARN) of the agent.
DescribeAgent

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$

CreationTime (p. 163)

The time that the agent was activated (that is, created in your account).

Type: Timestamp

EndpointType (p. 163)

The type of endpoint that your agent is connected to. If the endpoint is a VPC endpoint, the agent is not accessible over the public internet.

Type: String

Valid Values: PUBLIC | PRIVATE_LINK | FIPS

LastConnectionTime (p. 163)

The time that the agent last connected to DataSync.

Type: Timestamp

Name (p. 163)

The name of the agent.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=._:@/-]+$

PrivateLinkConfig (p. 163)

The subnet and the security group that DataSync used to access a VPC endpoint.

Type: PrivateLinkConfig (p. 226) object

Status (p. 163)

The status of the agent. If the status is ONLINE, then the agent is configured properly and is available to use. The Running status is the normal running status for an agent. If the status is OFFLINE, the agent's VM is turned off or the agent is in an unhealthy state. When the issue that caused the unhealthy state is resolved, the agent returns to ONLINE status.

Type: String

Valid Values: ONLINE | OFFLINE

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500
InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example returns information about the agent specified in the sample request.

Sample Request

```
{
    "AgentArn": "arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44baca3"
}
```

Example

Sample Response

```
{
    "AgentArn": "arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44baca3",
    "CreationTime": "1532660733.39",
    "LastConnectionTime": "1532660733.39",
    "Name": "MyAgent",
    "Status": "ONLINE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeLocationEfs

Returns metadata, such as the path information about an Amazon EFS location.

Request Syntax

```json
{
    "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**LocationArn (p. 166)**

The Amazon Resource Name (ARN) of the EFS location to describe.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: Yes

Response Syntax

```json
{
    "CreationTime": number,
    "Ec2Config": {
        "SecurityGroupArns": [ "string" ],
        "SubnetArn": "string"
    },
    "LocationArn": "string",
    "LocationUri": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CreationTime (p. 166)**

The time that the EFS location was created.

Type: Timestamp

**Ec2Config (p. 166)**

The subnet and the security group that DataSync uses to access target EFS file system. The subnet must have at least one mount target for that file system. The security group that you provide needs to be able to communicate with the security group on the mount target in the subnet specified.
DescribeLocationEfs

Type: Ec2Config (p. 216) object

LocationArn (p. 166)

The Amazon Resource Name (ARN) of the EFS location that was described.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

LocationUri (p. 166)

The URL of the EFS location that was described.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9.\-]+$  

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example returns information about the Amazon EFS location specified in the sample request.

Sample Request

```
{
        "LocationArn": "arn:aws:datasync:us-east-2:1112223344:location/loc-07db7abfc326c50fb"
}
```

Example

Sample Response

```
{
        "CreationTime": "",
        "Ec2Config": {
```
| }, |
| "LocationUri" : "us-east-2.fs-abcd1234. ", |

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeLocationFsxWindows

Returns metadata, such as the path information about an Amazon FSx for Windows location.

Request Syntax

```
{
   "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**LocationArn (p. 169)**

The Amazon Resource Name (ARN) of the FSx for Windows location to describe.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$`

Required: Yes

Response Syntax

```
{
   "CreationTime": number,
   "Domain": "string",
   "LocationArn": "string",
   "LocationUri": "string",
   "SecurityGroupArns": [ "string" ],
   "User": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CreationTime (p. 169)**

The time that the FSx for Windows location was created.

Type: Timestamp

**Domain (p. 169)**

The name of the Windows domain that the FSx for Windows server belongs to.
DescribeLocationFsxWindows

Type: String

Length Constraints: Maximum length of 253.

Pattern: ^([A-Za-z0-9-][A-Za-z0-9-]*[*][A-Za-z0-9-]*[A-Za-z0-9])$

**LocationArn (p. 169)**

The Amazon Resource Name (ARN) of the FSx for Windows location that was described.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

**LocationUri (p. 169)**

The URL of the FSx for Windows location that was described.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9.\-]+$

**SecurityGroupArns (p. 169)**

The Amazon Resource Names (ARNs) of the security groups that are configured for the FSx for Windows file system.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):ec2:[a-z\-0-9]+:[0-9]{12}:security-group/.*$

**User (p. 169)**

The user who has the permissions to access files and folders in the FSx for Windows file system.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^[^\x5B\x5D\x7C\x7E\x37\x38|=,+*?]\{1,104}\$

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeLocationNfs

Returns metadata, such as the path information, about an NFS location.

Request Syntax

```json
{
    "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**LocationArn (p. 172)**

The Amazon Resource Name (ARN) of the NFS location to describe.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: Yes

Response Syntax

```json
{
    "CreationTime": number,
    "LocationArn": "string",
    "LocationUri": "string",
    "MountOptions": {
        "Version": "string"
    },
    "OnPremConfig": {
        "AgentArns": [ "string" ]
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CreationTime (p. 172)**

The time that the NFS location was created.

Type: Timestamp
**LocationArn (p. 172)**

The Amazon Resource Name (ARN) of the NFS location that was described.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

**LocationUri (p. 172)**

The URL of the source NFS location that was described.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9.\-]+$

**MountOptions (p. 172)**

The NFS mount options that DataSync used to mount your NFS share.

Type: NfsMountOptions (p. 220) object

**OnPremConfig (p. 172)**

A list of Amazon Resource Names (ARNs) of agents to use for a Network File System (NFS) location.

Type: OnPremConfig (p. 221) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**Examples**

**Example**

The following example returns information about the NFS location specified in the sample request.

**Sample Request**

```json
{
  "LocationArn": "arn:aws:datasync:us-east-2:11122233444:location/loc-07db7abfc326c50aa"
}
```
Example

Sample Response

```
{
  "CreationTime": 1532660733.39,
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50aa",
  "LocationUri": "hostname.amazon.com",
  "OnPremConfig": {
    "AgentArns": [ "arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44b3nfs" ]
  }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeLocationS3

Returns metadata, such as bucket name, about an Amazon S3 bucket location.

Request Syntax

```
{
   "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

LocationArn (p. 175)

The Amazon Resource Name (ARN) of the Amazon S3 bucket location to describe.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$`

Required: Yes

Response Syntax

```
{
   "CreationTime": number,
   "LocationArn": "string",
   "LocationUri": "string",
   "S3Config": {
      "BucketAccessRoleArn": "string"
   },
   "S3StorageClass": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CreationTime (p. 175)

The time that the Amazon S3 bucket location was created.

Type: Timestamp

LocationArn (p. 175)

The Amazon Resource Name (ARN) of the Amazon S3 bucket location.
DescribeLocationS3

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-zA-Z0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

LocationUri (p. 175)

The URL of the Amazon S3 location that was described.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9.\-\]+$  

S3Config (p. 175)

The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that is used to access an Amazon S3 bucket.

For detailed information about using such a role, see Creating a Location for Amazon S3 in the AWS DataSync User Guide.

Type: S3Config (p. 228) object

S3StorageClass (p. 175)

The Amazon S3 storage class that you chose to store your files in when this location is used as a task destination. For more information about S3 storage classes, see Amazon S3 Storage Classes in the Amazon Simple Storage Service Developer Guide. Some storage classes have behaviors that can affect your S3 storage cost. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72).

Type: String

Valid Values: STANDARD | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example returns information about the S3 location specified in the sample request.
Sample Request

```json
{
   "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50s3"
}
```

Example

Sample Response

```json
{
   "CreationTime": 1532660733.39,
   "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50s3",
   "LocationUri": "MyBucket.",
   "S3Config": {
      "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/MyBucketAccessRole",
   },
   "S3StorageClass": "STANDARD"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeLocationSmb

Returns metadata, such as the path and user information about an SMB location.

Request Syntax

```json
{
   "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**LocationArn (p. 178)**

The Amazon Resource Name (ARN) of the SMB location to describe.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: Yes

Response Syntax

```json
{
   "AgentArns": [ "string" ],
   "CreationTime": number,
   "Domain": "string",
   "LocationArn": "string",
   "LocationUri": "string",
   "MountOptions": {
      "Version": "string"
   },
   "User": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**AgentArns (p. 178)**

The Amazon Resource Name (ARN) of the source SMB file system location that is created.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 64 items.
DescribeLocationSmb

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$

**CreationTime (p. 178)**

The time that the SMB location was created.

Type: Timestamp

**Domain (p. 178)**

The name of the Windows domain that the SMB server belongs to.

Type: String

Length Constraints: Maximum length of 253.

Pattern: ^([A-Za-z0-9]+[A-Za-z0-9-\.]*)*[A-Za-z0-9-]*[A-Za-z0-9-]*[A-Za-z0-9-]*[A-Za-z0-9-]*$ LocationArn (p. 178)

The Amazon Resource Name (ARN) of the SMB location that was described.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:location/loc-[0-9a-z]{17}$ LocationUri (p. 178)

The URL of the source SMB location that was described.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9-]+\+ MountOptions (p. 178)

The mount options that are available for DataSync to use to access an SMB location.

Type: SmbMountOptions (p. 229) object

**User (p. 178)**

The user who can mount the share, has the permissions to access files and folders in the SMB share.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^[^\x00\x0B\x0D\x20;|=,*?]{1,104}$

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.
HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

Sample Request

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

Example

Sample Response

```
{
  "AgentArns": [
    "arn:aws:datasync:us-east-2:111222333444:agent/agent-0bc3b3dc9bb0c15145",
    "arn:aws:datasync:us-east-2:111222333444:agent/agent-04b3fe3d261a18c8f"
  ],
  "CreationTime": "1532660733.39",
  "Domain": "AMAZON",
  "LocationArn": "arn:aws:datasync:us-east-1:111222333444:location/loc-0f01451b140b2af49",
  "LocationUri": "smb://hostname.amazon.com/share",
  "MountOptions": {
    "Version": "SMB3"
  },
  "User": "user-1"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeTask

Returns metadata about a task.

Request Syntax

```json
{
    "TaskArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**TaskArn (p. 181)**

The Amazon Resource Name (ARN) of the task to describe.

- **Type:** String
- **Length Constraints:** Maximum length of 128.
- **Pattern:** ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}$
- **Required:** Yes

Response Syntax

```json
{
    "CloudWatchLogGroupArn": "string",
    "CreationTime": number,
    "CurrentTaskExecutionArn": "string",
    "DestinationLocationArn": "string",
    "DestinationNetworkInterfaceArns": [ "string" ],
    "ErrorCode": "string",
    "ErrorDetail": "string",
    "Excludes": [ {
        "FilterType": "string",
        "Value": "string"
    } ],
    "Name": "string",
    "Options": { 
        "Atime": "string",
        "BytesPerSecond": number,
        "Gid": "string",
        "LogLevel": "string",
        "Mtime": "string",
        "OverwriteMode": "string",
        "PosixPermissions": "string",
        "PreserveDeletedFiles": "string",
        "PreserveDevices": "string",
        "TaskQueueing": "string",
        "Uid": "string",
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CloudWatchLogGroupArn (p. 181)

The Amazon Resource Name (ARN) of the Amazon CloudWatch log group that was used to monitor and log events in the task.

For more information on these groups, see Working with Log Groups and Log Streams in the Amazon CloudWatch User Guide.

Type: String
Length Constraints: Maximum length of 562.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):logs:[a-z\-0-9]*:[0-9]{12}:log-group:\[^:]*$  

CreationTime (p. 181)

The time that the task was created.

Type: Timestamp

CurrentTaskExecutionArn (p. 181)

The Amazon Resource Name (ARN) of the task execution that is syncing files.

Type: String
Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$

DestinationLocationArn (p. 181)

The Amazon Resource Name (ARN) of the AWS storage resource's location.

Type: String
Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:location/loc-[0-9a-zA-Z]{17}$

DestinationNetworkInterfaceArns (p. 181)

The Amazon Resource Name (ARN) of the destination ENIs (Elastic Network Interface) that was created for your subnet.
Type: Array of strings

Length Constraints: Maximum length of 128.

Pattern: ^arn:aws[\-a-z]{0,}:ec2:[a-z\-0-9]*:[0-9]{12}:network-interface/eni-[0-9a-f]+$

**ErrorCode (p. 181)**

Errors that AWS DataSync encountered during execution of the task. You can use this error code to help troubleshoot issues.

Type: String

**ErrorDetail (p. 181)**

Detailed description of an error that was encountered during the task execution. You can use this information to help troubleshoot issues.

Type: String

**Excludes (p. 181)**

A list of filter rules that determines which files to exclude from a task. The list should contain a single filter string that consists of the patterns to exclude. The patterns are delimited by "|" (that is, a pipe), for example: "/folder1|/folder2"

Type: Array of FilterRule (p. 217) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

**Name (p. 181)**

The name of the task that was described.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\-\s+=._:@/-]+$,

**Options (p. 181)**

The set of configuration options that control the behavior of a single execution of the task that occurs when you call `StartTaskExecution`. You can configure these options to preserve metadata such as user ID (UID) and group (GID), file permissions, data integrity verification, and so on.

For each individual task execution, you can override these options by specifying the overriding `OverrideOptions` value to `StartTaskExecution (p. 203)` operation.

Type: Options (p. 222) object

**Schedule (p. 181)**

The schedule used to periodically transfer files from a source to a destination location.

Type: TaskSchedule (p. 235) object

**SourceLocationArn (p. 181)**

The Amazon Resource Name (ARN) of the source file system's location.

Type: String

Length Constraints: Maximum length of 128.
Pattern: \^arn:\(\(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b\):datasync:[:a-z\-0-9\+]:\[0-9\]{12}:location/loc-[0-9a-z]{17}\$\n
**SourceNetworkInterfaceArns (p. 181)**

The Amazon Resource Name (ARN) of the source ENIs (Elastic Network Interface) that was created for your subnet.

Type: Array of strings

Length Constraints: Maximum length of 128.

Pattern: \^arn:aws[\-a-z]{0,}:ec2[\-a-z\-0-9\*]:[0-9\]{12}:network-interface/eni-[0-9a-f]+\$

**Status (p. 181)**

The status of the task that was described.

For detailed information about task execution statuses, see Understanding Task Statuses in the AWS DataSync User Guide.

Type: String

Valid Values: AVAILABLE | CREATING | QUEUED | RUNNING | UNAVAILABLE

**TaskArn (p. 181)**

The Amazon Resource Name (ARN) of the task that was described.

Type: String

Length Constraints: Maximum length of 128.

Pattern: \^arn:\(\(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b\):datasync:[:a-z\-0-9\*]:\[0-9\]{12}:task/task-[0-9a-f]{17}\$\n
**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**Examples**

**Example**

The following example returns information about the task specified in the sample request.

**Sample Request**

```
{
```
"TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026"
}

Example

Sample Response

`
{
  "CreationTime": 1532660733.39,
  "CurrentTaskExecutionArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f",
  "Options": {
    "Atime": "BEST_EFFORT",
    "BytesPerSecond": 1000,
    "Gid": "NONE",
    "Mtime": "PRESERVE",
    "PosixPermissions": "PRESERVE",
    "PreserveDevices": "NONE",
    "PreserveDeletedFiles": "PRESERVE",
    "Uid": "NONE",
    "VerifyMode": "POINT_IN_TIME_CONSISTENT"
  },
  "DestinationLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50fb",
  "ErrorCode": "??????",
  "ErrorDetail": "??????",
  "Name": "MyTask",
  "SourceLocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50aa",
  "Status": "CREATING",
  "TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026"
}
``

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeTaskExecution

Returns detailed metadata about a task that is being executed.

Request Syntax

```
{
   "TaskExecutionArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

TaskExecutionArn (p. 186)

The Amazon Resource Name (ARN) of the task that is being executed.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$`

Required: Yes

Response Syntax

```
{
   "BytesTransferred": number,
   "BytesWritten": number,
   "EstimatedBytesToTransfer": number,
   "EstimatedFilesToTransfer": number,
   "Excludes": [
      { "FilterType": "string", "Value": "string" }
   ],
   "FilesTransferred": number,
   "Includes": [
      { "FilterType": "string", "Value": "string" }
   ],
   "Options": {
      "Atime": "string",
      "BytesPerSecond": number,
      "Gid": "string",
      "LogLevel": "string",
      "Mtime": "string",
      "OverwriteMode": "string",
      "PosixPermissions": "string",
      "TaskExecutionArn": "string"
   }
}
```

186
"PreserveDeletedFiles": "string",
"PreserveDevices": "string",
"TaskQueueing": "string",
"Uid": "string",
"VerifyMode": "string"
}

"Result": {
  "ErrorCode": "string",
  "ErrorDetail": "string",
  "PrepareDuration": number,
  "PrepareStatus": "string",
  "TotalDuration": number,
  "TransferDuration": number,
  "TransferStatus": "string",
  "VerifyDuration": number,
  "VerifyStatus": "string"
},

"StartTime": number,

"Status": "string",

"TaskExecutionArn": "string"

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

BytesTransferred (p. 186)

The physical number of bytes transferred over the network.

Type: Long

BytesWritten (p. 186)

The number of logical bytes written to the destination AWS storage resource.

Type: Long

EstimatedBytesToTransfer (p. 186)

The estimated physical number of bytes that is to be transferred over the network.

Type: Long

EstimatedFilesToTransfer (p. 186)

The expected number of files that is to be transferred over the network. This value is calculated during the PREPARING phase, before the TRANSFERRING phase. This value is the expected number of files to be transferred. It's calculated based on comparing the content of the source and destination locations and finding the delta that needs to be transferred.

Type: Long

Excludes (p. 186)

A list of filter rules that determines which files to exclude from a task. The list should contain a single filter string that consists of the patterns to exclude. The patterns are delimited by "|" (that is, a pipe), for example: "/folder1/folder2"

Type: Array of FilterRule (p. 217) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.
**FilesTransferred (p. 186)**

The actual number of files that was transferred over the network. This value is calculated and updated on an ongoing basis during the TRANSFERRING phase. It’s updated periodically when each file is read from the source and sent over the network.

If failures occur during a transfer, this value can be less than EstimatedFilesToTransfer. This value can also be greater than EstimatedFilesTransferred in some cases. This element is implementation-specific for some location types, so don’t use it as an indicator for a correct file number or to monitor your task execution.

Type: Long

**Includes (p. 186)**

A list of filter rules that determines which files to include when running a task. The list should contain a single filter string that consists of the patterns to include. The patterns are delimited by ”|” (that is, a pipe), for example: ”/folder1|/folder2"

Type: Array of FilterRule (p. 217) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

**Options (p. 186)**

Represents the options that are available to control the behavior of a StartTaskExecution (p. 203) operation. Behavior includes preserving metadata such as user ID (UID), group ID (GID), and file permissions, and also overwriting files in the destination, data integrity verification, and so on.

A task has a set of default options associated with it. If you don’t specify an option in StartTaskExecution (p. 203), the default value is used. You can override the defaults options on each task execution by specifying an overriding options value to StartTaskExecution (p. 203).

Type: Options (p. 222) object

**Result (p. 186)**

The result of the task execution.

Type: TaskExecutionResultDetail (p. 232) object

**StartTime (p. 186)**

The time that the task execution was started.

Type: Timestamp

**Status (p. 186)**

The status of the task execution.

For detailed information about task execution statuses, see Understanding Task Statuses.

Type: String

Valid Values: QUEUED | LAUNCHING | PREPARING | TRANSFERRING | VERIFYING | SUCCESS | ERROR

**TaskExecutionArn (p. 186)**

The Amazon Resource Name (ARN) of the task execution that was described. TaskExecutionArn is hierarchical and includes TaskArn for the task that was executed.

For example, a TaskExecution value with the ARN arn:aws:datasync:us-east-1:112233445:task/task-020875f79cedf4a2/execution/
exec-08ef1e88ec491019b executed the task with the ARN arn:aws:datasync:us-east-1:111222333444:task/task-0208075f79cedf4a2.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Example

The following example returns information about the TaskExecution value specified in the sample request.

Sample Request

```
{
  "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f"
}
```

Example

Sample Response

```
{
  "BytesTransferred": "5000",
  "BytesWritten": "5000",
  "EstimatedBytesToTransfer": "5000",
  "EstimatedFilesToTransfer": "100",
  "FilesTransferred": "100",
  "Result": {
    "ErrorCode": "??????",
    "ErrorDetail": "??????",
    "PrepareDuration": "100",
    "PrepareStatus": "SUCCESS",
    "TransferDuration": "60",
    "TransferStatus": "AVAILABLE",
    "VerifyDuration": "30",
```
"VerifyStatus": "SUCCESS",
"StartTime": "1532660733.39",
"Status": "SUCCESS",
"OverrideOptions": {
    "Atime": "BEST_EFFORT",
    "BytesPerSecond": "1000",
    "Gid": "NONE",
    "Mtime": "PRESERVE",
    "PosixPermissions": "PRESERVE",
    "PreserveDevices": "NONE",
    "PreserveDeletedFiles": "PRESERVE",
    "Uid": "NONE",
    "VerifyMode": "POINT_IN_TIME_CONSISTENT"
},
"TaskExecutionArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de66697796f026/execution/exec-04ce9d516d69bd52f"

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListAgents

Returns a list of agents owned by an AWS account in the AWS Region specified in the request. The returned list is ordered by agent Amazon Resource Name (ARN).

By default, this operation returns a maximum of 100 agents. This operation supports pagination that enables you to optionally reduce the number of agents returned in a response.

If you have more agents than are returned in a response (that is, the response returns only a truncated list of your agents), the response contains a marker that you can specify in your next request to fetch the next page of agents.

Request Syntax

```
{
    "MaxResults": number,
    "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**MaxResults (p. 191)**

The maximum number of agents to list.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

**NextToken (p. 191)**

An opaque string that indicates the position at which to begin the next list of agents.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=_-]+

Required: No

Response Syntax

```
{
    "Agents": [
        {
            "AgentArn": "string",
            "Name": "string",
            "Status": "string"
        }
    ],
```
"NextToken": "string"
}

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**Agents (p. 191)**

A list of agents in your account.

Type: Array of AgentListEntry (p. 215) objects

**NextToken (p. 191)**

An opaque string that indicates the position at which to begin returning the next list of agents.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: `[a-zA-Z0-9=_-]+`

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListLocations

Returns a list of source and destination locations.

If you have more locations than are returned in a response (that is, the response returns only a truncated list of your agents), the response contains a token that you can specify in your next request to fetch the next page of locations.

Request Syntax

```json
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

MaxResults (p. 193)

The maximum number of locations to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

NextToken (p. 193)

An opaque string that indicates the position at which to begin the next list of locations.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=_-]+

Required: No

Response Syntax

```json
{
   "Locations": [
      {
         "LocationArn": "string",
         "LocationUri": "string"
      }
   ],
   "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**Locations (p. 193)**

An array that contains a list of locations.

Type: Array of LocationListEntry (p. 218) objects

**NextToken (p. 193)**

An opaque string that indicates the position at which to begin returning the next list of locations.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=_-]+

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListTagsForResource

Returns all the tags associated with a specified resource.

Request Syntax

```
{
  "MaxResults": number,
  "NextToken": "string",
  "ResourceArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**MaxResults (p. 195)**

The maximum number of locations to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

**NextToken (p. 195)**

An opaque string that indicates the position at which to begin the next list of locations.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=._-]+

Required: No

**ResourceArn (p. 195)**

The Amazon Resource Name (ARN) of the resource whose tags to list.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:(agent|task|location)/(agent|task|loc)-[0-9a-z]{17}$

Required: Yes

Response Syntax

```
{
  "NextToken": "string",
}
```
"Tags": [
  {
    "Key": "string",
    "Value": "string"
  }
]

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

NextToken (p. 195)

An opaque string that indicates the position at which to begin returning the next list of resource tags.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [\w\-]+

Tags (p. 195)

Array of resource tags.

Type: Array of TagListEntry (p. 230) objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
ListTaskExecutions

Returns a list of executed tasks.

Request Syntax

```json
{
   "MaxResults": number,
   "NextToken": "string",
   "TaskArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

MaxResults (p. 198)

The maximum number of executed tasks to list.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

NextToken (p. 198)

An opaque string that indicates the position at which to begin the next list of the executed tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9-_]+

Required: No

TaskArn (p. 198)

The Amazon Resource Name (ARN) of the task whose tasks you want to list.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*: [0-9]{12}:task/task-[0-9a-f]{17}$

Required: No

Response Syntax

```json
{
   "NextToken": "string",
   "TaskExecutions": [ ]
}
```
{
    "Status": "string",
    "TaskExecutionArn": "string"
}
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 198)**

An opaque string that indicates the position at which to begin returning the next list of executed tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=\-_]+

**TaskExecutions (p. 198)**

A list of executed tasks.

Type: Array of TaskExecutionListEntry (p. 231) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
ListTasks

Returns a list of all the tasks.

Request Syntax

```json
{
    "MaxResults": number,
    "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**MaxResults (p. 201)**

The maximum number of tasks to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

**NextToken (p. 201)**

An opaque string that indicates the position at which to begin the next list of tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9-_]+

Required: No

Response Syntax

```json
{
    "NextToken": "string",
    "Tasks": [
        {
            "Name": "string",
            "Status": "string",
            "TaskArn": "string"
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**NextToken (p. 201)**

An opaque string that indicates the position at which to begin returning the next list of tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9=_-]+

**Tasks (p. 201)**

A list of all the tasks that are returned.

Type: Array of TaskListEntry (p. 234) objects

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
StartTaskExecution

Starts a specific invocation of a task. A TaskExecution value represents an individual run of a task. Each task can have at most one TaskExecution at a time.

TaskExecution has the following transition phases: INITIALIZING | PREPARING | TRANSFERRING | VERIFYING | SUCCESS/Failure.

For detailed information, see Task Execution in Components and Terminology in the AWS DataSync User Guide.

Request Syntax

```json
{
   "Includes": [
   {
      "FilterType": "string",
      "Value": "string"
   }
   ],
   "OverrideOptions": {
      "Atime": "string",
      "BytesPerSecond": number,
      "Gid": "string",
      "LogLevel": "string",
      "Mtime": "string",
      "OverwriteMode": "string",
      "PosixPermissions": "string",
      "PreserveDeletedFiles": "string",
      "PreserveDevices": "string",
      "TaskQueueing": "string",
      "Uid": "string",
      "VerifyMode": "string"
   },
   "TaskArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

Includes (p. 203)

A list of filter rules that determines which files to include when running a task. The pattern should contain a single filter string that consists of the patterns to include. The patterns are delimited by "|" (that is, a pipe). For example: "/folder1|/folder2"

Type: Array of FilterRule (p. 217) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

OverrideOptions (p. 203)

Represents the options that are available to control the behavior of a StartTaskExecution (p. 203) operation. Behavior includes preserving metadata such as user ID (UID), group ID (GID), and file permissions, and also overwriting files in the destination, data integrity verification, and so on.
A task has a set of default options associated with it. If you don't specify an option in StartTaskExecution (p. 203), the default value is used. You can override the defaults options on each task execution by specifying an overriding options value to StartTaskExecution (p. 203).

Type: Options (p. 222) object

Required: No

**TaskArn (p. 203)**

The Amazon Resource Name (ARN) of the task to start.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*: [0-9]{12}:task/task-[0-9a-f]{17}$

Required: Yes

**Response Syntax**

```
{
  "TaskExecutionArn": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**TaskExecutionArn (p. 204)**

The Amazon Resource Name (ARN) of the specific task execution that was started.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*: [0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400
Examples

Example

The following example starts a task execution using the default options and the specified task.

Sample Request

```json
{
    "OverrideOptions": {
        "Atime": "BEST_EFFORT",
        "BytesPerSecond": 1000,
        "Gid": "NONE",
        "Mtime": "PRESERVE",
        "PosixPermissions": "PRESERVE",
        "PreserveDevices": "NONE",
        "PreserveDeletedFiles": "PRESERVE",
        "Uid": "NONE",
        "VerifyMode": "POINT_IN_TIME_CONSISTENT"
    },
    "TaskArn": "arn:aws:datasync:us-east-2:1112233444:task/task-08de6e6697796f026"
}
```

Example

Sample Response

```json
{
    "TaskExecutionArn": "arn:aws:datasync:us-east-2:1112233444:task/task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
TagResource

Applies a key-value pair to an AWS resource.

Request Syntax

```json
{
"ResourceArn": "string",
"Tags": [
{
"Key": "string",
"Value": "string"
}
]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**ResourceArn (p. 206)**

The Amazon Resource Name (ARN) of the resource to apply the tag to.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:(agent|task|location)/(agent|task|loc)-[0-9a-z]{17}$

Required: Yes

**Tags (p. 206)**

The tags to apply.

Type: Array of TagListEntry (p. 230) objects

Array Members: Minimum number of 0 items. Maximum number of 55 items.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.
HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UntagResource

Removes a tag from an AWS resource.

Request Syntax

```
{
  "Keys": [ "string" ],
  "ResourceArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common
Parameters (p. 237).

The request accepts the following data in JSON format.

**Keys (p. 208)**

The keys in the key-value pair in the tag to remove.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `^[a-zA-Z0-9\s+=._:/-]+$`

Required: Yes

**ResourceArn (p. 208)**

The Amazon Resource Name (ARN) of the resource to remove the tag from.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:
[0-9]{12}:(agent|task|location)/(agent|task|loc)[-0-9a-z]{17}$`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500
InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateAgent

Updates the name of an agent.

Request Syntax

```json
{
  "AgentArn": "string",
  "Name": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

AgentArn (p. 210)

The Amazon Resource Name (ARN) of the agent to update.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:a-z\-0-9+:0-9\{12\}:agent/agent-[0-9a-z]\{17\}$

Required: Yes

Name (p. 210)

The name that you want to use to configure the agent.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-ZÀ-Ž0-9\s=._:@/-]+$

Required: No

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500
InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**UpdateTask**

Updates the metadata associated with a task.

**Request Syntax**

```json
{
  "CloudWatchLogGroupArn": "string",
  "Excludes": [
    {
      "FilterType": "string",
      "Value": "string"
    }
  ],
  "Name": "string",
  "Options": {
    "Atime": "string",
    "BytesPerSecond": number,
    "Gid": "string",
    "LogLevel": "string",
    "Mtime": "string",
    "OverwriteMode": "string",
    "PosixPermissions": "string",
    "PreserveDeletedFiles": "string",
    "PreserveDevices": "string",
    "TaskQueueing": "string",
    "Uid": "string",
    "VerifyMode": "string"
  },
  "Schedule": {
    "ScheduleExpression": "string"
  },
  "TaskArn": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 237).

The request accepts the following data in JSON format.

**CloudWatchLogGroupArn (p. 212)**

The Amazon Resource Name (ARN) of the resource name of the CloudWatch LogGroup.

Type: String

Length Constraints: Maximum length of 562.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):logs:[a-z\-0-9]*:[0-9]{12}:log-group:([^:]*$)

Required: No

**Excludes (p. 212)**

A list of filter rules that determines which files to exclude from a task. The list should contain a single filter string that consists of the patterns to exclude. The patterns are delimited by "|" (that is, a pipe), for example: "/folder1|/folder2"
UpdateTask

Type: Array of FilterRule (p. 217) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

Name (p. 212)

The name of the task to update.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\-\s+=._:@/-]+$

Required: No

Options (p. 212)

Represents the options that are available to control the behavior of a StartTaskExecution (p. 203) operation. Behavior includes preserving metadata such as user ID (UID), group ID (GID), and file permissions, and also overwriting files in the destination, data integrity verification, and so on.

A task has a set of default options associated with it. If you don’t specify an option in StartTaskExecution (p. 203), the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution (p. 203).

Type: Options (p. 222) object

Required: No

Schedule (p. 212)

Specifies a schedule used to periodically transfer files from a source to a destination location. You can configure your task to execute hourly, daily, weekly or on specific days of the week. You control when in the day or hour you want the task to execute. The time you specify is UTC time. For more information, see Scheduling Your Task (p. 61).

Type: TaskSchedule (p. 235) object

Required: No

TaskArn (p. 212)

The Amazon Resource Name (ARN) of the resource name of the task to update.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}$

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 235).
**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

**Data Types**

The following data types are supported:

- AgentListEntry (p. 215)
- Ec2Config (p. 216)
- FilterRule (p. 217)
- LocationListEntry (p. 218)
- NfsMountOptions (p. 220)
- OnPremConfig (p. 221)
- Options (p. 222)
- PrivateLinkConfig (p. 226)
- S3Config (p. 228)
- SmbMountOptions (p. 229)
- TagListEntry (p. 230)
- TaskExecutionListEntry (p. 231)
- TaskExecutionResultDetail (p. 232)
- TaskListEntry (p. 234)
- TaskSchedule (p. 235)
AgentListEntry

Represents a single entry in a list of agents. AgentListEntry returns an array that contains a list of agents when the ListAgents (p. 191) operation is called.

Contents

AgentArn

The Amazon Resource Name (ARN) of the agent.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:\[(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:\[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: No

Name

The name of the agent.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\-0-9\s+=._:@/-]+$

Required: No

Status

The status of the agent.

Type: String

Valid Values: ONLINE | OFFLINE

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Ec2Config

The subnet and the security group that DataSync uses to access target EFS file system. The subnet must have at least one mount target for that file system. The security group that you provide needs to be able to communicate with the security group on the mount target in the subnet specified.

Contents

SecurityGroupArns

The Amazon Resource Names (ARNs) of the security groups that are configured for the Amazon EC2 resource.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-is|aws-is-b):ec2:[a-z\-0-9]*:[0-9]{12}:security-group/.*$

Required: Yes

SubnetArn

The ARN of the subnet and the security group that DataSync uses to access the target EFS file system.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-is|aws-is-b):ec2:[a-z\-0-9]*:[0-9]{12}:subnet/.*$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
FilterRule

Specifies which files, folders and objects to include or exclude when transferring files from source to destination.

Contents

FilterType

The type of filter rule to apply. AWS DataSync only supports the SIMPLE_PATTERN rule type.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^[A-Z0-9_]+$  

Valid Values: SIMPLE_PATTERN

Required: No

Value

A single filter string that consists of the patterns to include or exclude. The patterns are delimited by "|" (that is, a pipe), for example: /folder1|/folder2

Type: String

Length Constraints: Maximum length of 409600.

Pattern: ^[^\x00]+$  

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LocationListEntry

Represents a single entry in a list of locations. LocationListEntry returns an array that contains a list of locations when the ListLocations (p. 193) operation is called.

Contents

**LocationArn**

The Amazon Resource Name (ARN) of the location. For Network File System (NFS) or Amazon EFS, the location is the export path. For Amazon S3, the location is the prefix path that you want to mount and use as the root of the location.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: No

**LocationUri**

Represents a list of URLs of a location. LocationUri returns an array that contains a list of locations when the ListLocations (p. 193) operation is called.

Format: TYPE://GLOBAL_ID/SUBDIR.

TYPE designates the type of location. Valid values: NFS | EFS | S3.

GLOBAL_ID is the globally unique identifier of the resource that backs the location. An example for EFS is us-east-2.fs-abcd1234. An example for Amazon S3 is the bucket name, such as myBucket. An example for NFS is a valid IPv4 address or a host name compliant with Domain Name Service (DNS).

SUBDIR is a valid file system path, delimited by forward slashes as is the *nix convention. For NFS and Amazon EFS, it's the export path to mount the location. For Amazon S3, it's the prefix path that you mount to and treat as the root of the location.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9-\-]++$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NfsMountOptions

Represents the mount options that are available for DataSync to access an NFS location.

Contents

Version

The specific NFS version that you want DataSync to use to mount your NFS share. If the server refuses to use the version specified, the sync will fail. If you don't specify a version, DataSync defaults to AUTOMATIC. That is, DataSync automatically selects a version based on negotiation with the NFS server.

You can specify the following NFS versions:

- **NFSv3** - stateless protocol version that allows for asynchronous writes on the server.
- **NFSv4.0** - stateful, firewall-friendly protocol version that supports delegations and pseudo filesystems.
- **NFSv4.1** - stateful protocol version that supports sessions, directory delegations, and parallel data processing. Version 4.1 also includes all features available in version 4.0.

Type: String

Valid Values: AUTOMATIC | NFS3 | NFS4_0 | NFS4_1

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
OnPremConfig

A list of Amazon Resource Names (ARNs) of agents to use for a Network File System (NFS) location.

Contents

AgentArns

ARNs of the agents to use for an NFS location.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 64 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Options

Represents the options that are available to control the behavior of a StartTaskExecution (p. 203) operation. Behavior includes preserving metadata such as user ID (UID), group ID (GID), and file permissions, and also overwriting files in the destination, data integrity verification, and so on.

A task has a set of default options associated with it. If you don't specify an option in StartTaskExecution (p. 203), the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution (p. 203).

Contents

Atime

A file metadata value that shows the last time a file was accessed (that is, when the file was read or written to). If you set Atime to BEST_EFFORT, DataSync attempts to preserve the original Atime attribute on all source files (that is, the version before the PREPARING phase). However, Atime's behavior is not fully standard across platforms, so AWS DataSync can only do this on a best-effort basis.

Default value: BEST_EFFORT.

BEST_EFFORT: Attempt to preserve the per-file Atime value (recommended).

NONE: Ignore Atime.

   Note
   If Atime is set to BEST_EFFORT, Mtime must be set to PRESERVE.
   If Atime is set to NONE, Mtime must also be NONE.

Type: String

Valid Values: NONE | BEST_EFFORT

Required: No

BytesPerSecond

A value that limits the bandwidth used by AWS DataSync. For example, if you want AWS DataSync to use a maximum of 1 MB, set this value to 1048576 (=1024*1024).

Type: Long

Valid Range: Minimum value of -1.

Required: No

Gid

The group ID (GID) of the file's owners.

Default value: INT_VALUE. This preserves the integer value of the ID.

INT_VALUE: Preserve the integer value of user ID (UID) and GID (recommended).

NONE: Ignore UID and GID.

Type: String

Valid Values: NONE | INT_VALUE | NAME | BOTH
Required: No

LogLevel

A value that determines the type of logs that DataSync publishes to a log stream in the Amazon CloudWatch log group that you provide. For more information about providing a log group for DataSync, see CloudWatchLogGroupArn. If set to OFF, no logs are published. BASIC publishes logs on errors for individual files transferred, and TRANSFER publishes logs for every file or object that is transferred and integrity checked.

Type: String

Valid Values: OFF | BASIC | TRANSFER

Required: No

Mtime

A value that indicates the last time that a file was modified (that is, a file was written to) before the PREPARING phase.

Default value: PRESERVE.

PRESERVE: Preserve original Mtime (recommended)

NONE: Ignore Mtime.

Note

If Mtime is set to PRESERVE, Atime must be set to BEST_EFFORT.
If Mtime is set to NONE, Atime must also be set to NONE.

Type: String

Valid Values: NONE | PRESERVE

Required: No

OverwriteMode

A value that determines whether files at the destination should be overwritten or preserved when copying files. If set to NEVER a destination file will not be replaced by a source file, even if the destination file differs from the source file. If you modify files in the destination and you sync the files, you can use this value to protect against overwriting those changes.

Some storage classes have specific behaviors that can affect your S3 storage cost. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72) in the AWS DataSync User Guide.

Type: String

Valid Values: ALWAYS | NEVER

Required: No

PosixPermissions

A value that determines which users or groups can access a file for a specific purpose such as reading, writing, or execution of the file.

Default value: PRESERVE.

PRESERVE: Preserve POSIX-style permissions (recommended).

NONE: Ignore permissions.
**Options**

**Note**
AWS DataSync can preserve extant permissions of a source location.

Type: String

Valid Values: NONE | PRESERVE

Required: No

**PreserveDeletedFiles**

A value that specifies whether files in the destination that don't exist in the source file system should be preserved. This option can affect your storage cost. If your task deletes objects, you might incur minimum storage duration charges for certain storage classes. For detailed information, see Considerations When Working with S3 Storage Classes in DataSync (p. 72) in the AWS DataSync User Guide.

Default value: PRESERVE.

PRESERVE: Ignore such destination files (recommended).

REMOVE: Delete destination files that aren't present in the source.

Type: String

Valid Values: PRESERVE | REMOVE

Required: No

**PreserveDevices**

A value that determines whether AWS DataSync should preserve the metadata of block and character devices in the source file system, and recreate the files with that device name and metadata on the destination.

**Note**
AWS DataSync can't sync the actual contents of such devices, because they are nonterminal and don't return an end-of-file (EOF) marker.

Default value: NONE.

NONE: Ignore special devices (recommended).

PRESERVE: Preserve character and block device metadata. This option isn't currently supported for Amazon EFS.

Type: String

Valid Values: NONE | PRESERVE

Required: No

**TaskQueueing**

A value that determines whether tasks should be queued before executing the tasks. If set to ENABLED, the tasks will be queued. The default is ENABLED.

If you use the same agent to run multiple tasks, you can enable the tasks to run in series. For more information, see Queueing Task Executions (p. 57).

Type: String

Valid Values: ENABLED | DISABLED
Required: No

**Uid**

The user ID (UID) of the file's owner.

Default value: INT_VALUE. This preserves the integer value of the ID.

INT_VALUE: Preserve the integer value of UID and group ID (GID) (recommended).

NONE: Ignore UID and GID.

Type: String

Valid Values: NONE | INT_VALUE | NAME | BOTH

Required: No

**VerifyMode**

A value that determines whether a data integrity verification should be performed at the end of a task execution after all data and metadata have been transferred. For more information, see [Configure Task Settings](#) (p. 30)

Default value: POINT_IN_TIME_CONSISTENT.

ONLY_FILES_TRANSFERRED (recommended): Perform verification only on files that were transferred.

POINT_IN_TIME_CONSISTENT: Scan the entire source and entire destination at the end of the transfer to verify that source and destination are fully synchronized. This option isn't supported when transferring to S3 Glacier or S3 Glacier Deep Archive storage classes.

NONE: No additional verification is done at the end of the transfer, but all data transmissions are integrity-checked with checksum verification during the transfer.

Type: String

Valid Values: POINT_IN_TIME_CONSISTENT | ONLY_FILES_TRANSFERRED | NONE

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
PrivateLinkConfig

The VPC endpoint, subnet, and security group that an agent uses to access IP addresses in a VPC (Virtual Private Cloud).

Contents

PrivateLinkEndpoint

The private endpoint that is configured for an agent that has access to IP addresses in a PrivateLink. An agent that is configured with this endpoint will not be accessible over the public internet.

Type: String


Pattern: \A(25[0-5]|2[0-4]\d|[0-1]?\d?\d)(\.25[0-5]|2[0-4]\d|[0-1]?\d?\d)\{3}\z

Required: No

SecurityGroupArns

The Amazon Resource Names (ARNs) of the security groups that are configured for the EC2 resource that hosts an agent activated in a VPC or an agent that has access to a VPC endpoint.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-isob):ec2:\[a-z\-0-9]*:[0-9] \{12\}:security-group/.*$

Required: No

SubnetArns

The Amazon Resource Names (ARNs) of the subnets that are configured for an agent activated in a VPC or an agent that has access to a VPC endpoint.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-isob):ec2:\[a-z\-0-9]*:[0-9] \{12\}:subnet/.*$

Required: No

VpcEndpointId

The ID of the VPC endpoint that is configured for an agent. An agent that is configured with a VPC endpoint will not be accessible over the public internet.

Type: String

Pattern: ^vpce-[0-9a-f]{17}$
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3Config

The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that is used to access an Amazon S3 bucket.

For detailed information about using such a role, see Creating a Location for Amazon S3 in the AWS DataSync User Guide.

Contents

BucketAccessRoleArn

The Amazon S3 bucket to access. This bucket is used as a parameter in the CreateLocationS3 (p. 143) operation.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]{12}:role/.*$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
SmbMountOptions

Represents the mount options that are available for DataSync to access an SMB location.

Contents

Version

The specific SMB version that you want DataSync to use to mount your SMB share. If you don't specify a version, DataSync defaults to AUTOMATIC. That is, DataSync automatically selects a version based on negotiation with the SMB server.

Type: String

Valid Values: AUTOMATIC | SMB2 | SMB3

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
TagListEntry

Represents a single entry in a list of AWS resource tags. TagListEntry returns an array that contains a list of tasks when the ListTagsForResource (p. 195) operation is called.

Contents

Key

The key for an AWS resource tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=._:/-]+$

Required: Yes

Value

The value for an AWS resource tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: ^[a-zA-Z0-9\s+=._:@/-]+$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
TaskExecutionListEntry

Represents a single entry in a list of task executions. TaskExecutionListEntry returns an array that contains a list of specific invocations of a task when ListTaskExecutions (p. 198) operation is called.

Contents

Status

The status of a task execution.

Type: String

Valid Values: QUEUED | LAUNCHING | PREPARING | TRANSFERRING | VERIFYING | SUCCESS | ERROR

Required: No

TaskExecutionArn

The Amazon Resource Name (ARN) of the task that was executed.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*: [0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
TaskExecutionResultDetail

Describes the detailed result of a TaskExecution operation. This result includes the time in milliseconds spent in each phase, the status of the task execution, and the errors encountered.

Contents

ErrorCode

Errors that AWS DataSync encountered during execution of the task. You can use this error code to help troubleshoot issues.

Type: String

Required: No

ErrorDetail

Detailed description of an error that was encountered during the task execution. You can use this information to help troubleshoot issues.

Type: String

Required: No

PrepareDuration

The total time in milliseconds that AWS DataSync spent in the PREPARING phase.

Type: Long

Valid Range: Minimum value of 0.

Required: No

PrepareStatus

The status of the PREPARING phase.

Type: String

Valid Values: PENDING | SUCCESS | ERROR

Required: No

TotalDuration

The total time in milliseconds that AWS DataSync took to transfer the file from the source to the destination location.

Type: Long

Valid Range: Minimum value of 0.

Required: No

TransferDuration

The total time in milliseconds that AWS DataSync spent in the TRANSFERRING phase.

Type: Long

Valid Range: Minimum value of 0.
TransferStatus

The status of the TRANSFERRING Phase.

Type: String

Valid Values: PENDING | SUCCESS | ERROR

Required: No

VerifyDuration

The total time in milliseconds that AWS DataSync spent in the VERIFYING phase.

Type: Long

Valid Range: Minimum value of 0.

Required: No

VerifyStatus

The status of the VERIFYING Phase.

Type: String

Valid Values: PENDING | SUCCESS | ERROR

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
TaskListEntry

Represents a single entry in a list of tasks. TaskListEntry returns an array that contains a list of tasks when the ListTasks (p. 201) operation is called. A task includes the source and destination file systems to sync and the options to use for the tasks.

Contents

<table>
<thead>
<tr>
<th>Name</th>
<th>The name of the task.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td>Length Constraints: Minimum length of 1. Maximum length of 256.</td>
<td></td>
</tr>
<tr>
<td>Pattern: ^[a-zA-Z0-9\s+=._:@/-]+$</td>
<td></td>
</tr>
<tr>
<td>Required: No</td>
<td></td>
</tr>
</tbody>
</table>

Status

The status of the task.

Type: String

Valid Values: AVAILABLE | CREATING | QUEUED | RUNNING | UNAVAILABLE

Required: No

TaskArn

The Amazon Resource Name (ARN) of the task.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws iso|aws iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
TaskSchedule

Specifies the schedule you want your task to use for repeated executions. For more information, see Schedule Expressions for Rules.

Contents

ScheduleExpression

A cron expression that specifies when AWS DataSync initiates a scheduled transfer from a source to a destination location.

Type: String

Length Constraints: Maximum length of 256.

Pattern: ^[a-zA-Z0-9\ \_\*\?\,\|\^\-\/\#\s\(\)\+]*$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400
InvalidClientTokenId
The X.509 certificate or AWS access key ID provided does not exist in our records.
HTTP Status Code: 403

InvalidParameterCombination
Parameters that must not be used together were used together.
HTTP Status Code: 400

InvalidParameterValue
An invalid or out-of-range value was supplied for the input parameter.
HTTP Status Code: 400

InvalidQueryParameter
The AWS query string is malformed or does not adhere to AWS standards.
HTTP Status Code: 400

MalformedQueryString
The query string contains a syntax error.
HTTP Status Code: 404

MissingAction
The request is missing an action or a required parameter.
HTTP Status Code: 400

MissingAuthenticationToken
The request must contain either a valid (registered) AWS access key ID or X.509 certificate.
HTTP Status Code: 403

MissingParameter
A required parameter for the specified action is not supplied.
HTTP Status Code: 400

OptInRequired
The AWS access key ID needs a subscription for the service.
HTTP Status Code: 403

RequestExpired
The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.
HTTP Status Code: 400

ServiceUnavailable
The request has failed due to a temporary failure of the server.
HTTP Status Code: 503
ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signature Version 4 Signing Process in the Amazon Web Services General Reference.

**Action**

The action to be performed.

Type: string

Required: Yes

**Version**

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

**X-Amz-Algorithm**

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

**X-Amz-Credential**

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string (“aws4_request”). The value is expressed in the following format: access_key/YYYYMMDD/region/service/aws4_request.

For more information, see Task 2: Create a String to Sign for Signature Version 4 in the Amazon Web Services General Reference.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string
Required: Conditional

**X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 in the Amazon Web Services General Reference.

Type: string

Required: Conditional

**X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS Security Token Service, go to AWS Services That Work with IAM in the IAM User Guide.

Condition: If you're using temporary security credentials from the AWS Security Token Service, you must include the security token.

Type: string

Required: Conditional

**X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-SignedHeaders**

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Task 1: Create a Canonical Request For Signature Version 4 in the Amazon Web Services General Reference.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional
## Document History for AWS DataSync

The following table describes the documentation for this release of AWS DataSync.

- **API version:** datasync-2018-11-09
- **Latest documentation update:** April 24, 2020

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enhanced monitoring capabilities with file level logging.</td>
<td>You can now enable detailed logging for files and objects copied between your NFS servers, SMB servers, Amazon S3 buckets, Amazon EFS file systems, and Amazon FSx for Windows File Server file systems. For more information about configuring file level logging, see Configuring Task Settings (p. 55).</td>
<td>April 24, 2020</td>
</tr>
<tr>
<td>Support for copying data between your AWS DataSync Server Message Block (SMB) share and Amazon FSx for Windows File Server.</td>
<td>You can now copy data between your DataSync Server Message Block (SMB) share and Amazon FSx for Windows File Server. For more information, see Creating a Location for Amazon FSx for Windows File Server (p. 69).</td>
<td>January 24, 2020</td>
</tr>
<tr>
<td>Support for scheduling tasks</td>
<td>You can now run tasks manually or schedule them to run based on a specified schedule. For more information, see Scheduling Your Task (p. 61).</td>
<td>November 20, 2019</td>
</tr>
<tr>
<td>New AWS Region</td>
<td>AWS DataSync is now available in the Asia Pacific (Hong Kong) Region, Asia Pacific (Mumbai) Region, EU (Stockholm) Region, South America (Sao Paulo) Region and AWS GovCloud (US-East) Region. For more information, see AWS DataSync Regions in the AWS General Reference.</td>
<td>November 20, 2019</td>
</tr>
<tr>
<td>New AWS Region</td>
<td>AWS DataSync is now available in the Canada (Central) Region, EU (London) Region, and EU (Paris) Region. For more information, see AWS DataSync Regions in the AWS General Reference.</td>
<td>October 02, 2019</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>Support for Amazon S3 storage classes</td>
<td>You can now transfer objects directly into Amazon S3 storage classes. For more information, see Creating a Location for Amazon S3 (p. 70).</td>
<td>September 24, 2019</td>
</tr>
<tr>
<td>New AWS Region</td>
<td>AWS DataSync is now available in the Middle East (Bahrain) Region. For more information, see AWS DataSync Regions in the AWS General Reference.</td>
<td>August 28, 2019</td>
</tr>
<tr>
<td>Support for copying data between your Server Message Block (SMB) share and Amazon S3 or Amazon EFS.</td>
<td>You can now copy data between your SMB file share and Amazon Simple Storage Service or Amazon Elastic File System. For more information, see Working with Locations (p. 64).</td>
<td>August 22, 2019</td>
</tr>
<tr>
<td>Support for Federal Information Processing Standard (FIPS) endpoints</td>
<td>You can now use FIPS endpoints to create agents and run tasks. For more information, see Choose a Service Endpoint (p. 25).</td>
<td>August 05, 2019</td>
</tr>
<tr>
<td>Support for using virtual private cloud (VPC) endpoints</td>
<td>You can now create a private connection between your on-premises agent and AWS and run tasks in a private network. Doing this increases the security of your data as it's copied over the network. For more information, see Using AWS DataSync in a Virtual Private Cloud (p. 79).</td>
<td>August 05, 2019</td>
</tr>
<tr>
<td>New AWS Region</td>
<td>AWS DataSync is now available in the AWS GovCloud (US-West) Region. For more information, see AWS DataSync Regions in the AWS General Reference.</td>
<td>June 11, 2019</td>
</tr>
<tr>
<td>Support for filtering</td>
<td>You can now apply filters to transfer only a subset of the files in your source location when you transfer data from your source to your destination location. For more information, see Filtering the Data Transferred by AWS DataSync (p. 58).</td>
<td>May 22, 2019</td>
</tr>
<tr>
<td>First release of AWS DataSync.</td>
<td>General release of the AWS DataSync service.</td>
<td>November 26, 2018</td>
</tr>
</tbody>
</table>
AWS glossary

For the latest AWS terminology, see the AWS glossary in the AWS General Reference.