AWS DataSync
User Guide
# 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  
  Data transfer between self-managed storage and AWS .................. 3  
  Data transfer between AWS storage services ............................... 4  
  Data transfer using a DataSync EC2 agent deployed in a Region .... 5  
Components and terminology .......................................................... 5  
  Agent ............................................................................................... 5  
  Location .......................................................................................... 5  
  Task ................................................................................................. 5  
  Task execution ................................................................................ 6  
How DataSync transfers files ........................................................... 6  
  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 .................................................. 10  
    Amazon EC2 instance requirements .......................................... 10  
  Network requirements for DataSync ............................................. 10  
    Network requirements to connect to your self-managed storage ... 11  
    Network requirements when using VPC endpoints ..................... 11  
    Network requirements when using public service endpoints or FIPS endpoints ........................................... 14  
Getting started ............................................................................ 18  
  Create an agent ............................................................................ 18  
    Deploy an agent ................................................................. 19  
    Choose a service endpoint ...................................................... 24  
    Activate your agent ............................................................... 26  
  Configure a source location ......................................................... 26  
  Configure a destination location .................................................. 28  
  Configure Task Settings .............................................................. 28  
    Data verification options ......................................................... 29  
    Ownership and permissions related options ............................ 29  
    File metadata options and file management ............................... 30  
    Bandwidth options ................................................................... 30  
    Filtering options ....................................................................... 30  
    Scheduling and queueing options ............................................. 31  
    Tags and logging options ........................................................ 31  
  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
AWS DataSync User Guide

Deleting a location ............................................................................................................. 88
Working with agents .......................................................................................................... 89
Creating and activating an agent ......................................................................................... 89
Using DataSync in a VPC ..................................................................................................... 90
  How DataSync works with VPC endpoints ........................................................................ 90
  Configuring DataSync to use private IP addresses for data transfer ................................ 90
Using the DataSync EC2 agent deployed in AWS Regions .................................................. 93
  Data transfer from in-cloud file system to in-cloud file system or Amazon S3 ................ 93
  Data transfer from S3 to in-cloud file systems ................................................................ 94
Editing your agent's properties .......................................................................................... 95
Using multiple agents for a location .................................................................................... 96
Understanding agent statuses ............................................................................................ 96
Deleting an agent ................................................................................................................ 96
Configuring your agent for multiple NICs .......................................................................... 97
Using the VM local console ............................................................................................... 98
  Logging in to the AWS DataSync local console ............................................................... 98
  Obtaining an activation key using the local console ......................................................... 98
  Configuring your agent network settings .......................................................................... 99
  Testing your agent connectivity to the internet ................................................................. 101
  Testing connectivity to self-managed storage .................................................................. 101
  Viewing your agent system resource status ..................................................................... 102
  Synchronizing your VMware agent time .......................................................................... 103
Running AWS DataSync commands on the local console ................................................. 103
Enabling AWS Support to help troubleshoot DataSync ..................................................... 104
Security ............................................................................................................................... 106
  Data protection ................................................................................................................ 106
  Data encryption ............................................................................................................... 106
  Identity and access management ....................................................................................... 107
  Using identity-based policies (IAM policies) .................................................................... 107
  DataSync API permissions reference ............................................................................... 111
  Overview of managing access ......................................................................................... 115
Logging ............................................................................................................................... 119
  Working with AWS DataSync information in CloudTrail ............................................... 119
  Understanding AWS DataSync log file entries ............................................................... 120
Compliance validation ........................................................................................................ 121
Resilience ............................................................................................................................ 122
Infrastructure security ....................................................................................................... 122
Troubleshooting ................................................................................................................ 123
  You need DataSync to use a specific NFS or SMB version to mount your share ............... 123
  You get a "Failed to retrieve agent activation key" error .................................................... 124
  You can't activate an agent created using a VPC endpoint ............................................... 124
  Your task status is unavailable and status indicates a mount error .................................... 124
  Your task execution fails with an Input/Output error message ......................................... 125
  Your task execution is stuck in launching status .............................................................. 125
  Your task execution fails with a permissions denied error message ................................. 125
  Preparing status for a task execution takes longer than expected to complete ................. 126
  Verifying status for a task execution takes longer than expected to complete ................. 126
  Your storage cost is higher than expected ....................................................................... 126
  How to enable AWS Support to help troubleshoot DataSync ........................................... 127
Troubleshooting an EC2 agent .......................................................................................... 128
  Enabling AWS Support to troubleshoot your DataSync running on Amazon EC2 .......... 128
  Connecting to the EC2 agent's local console ................................................................. 128
DataSync quotas and limits ................................................................................................. 129
  Quotas for tasks ............................................................................................................... 129
  Quotas for task executions .............................................................................................. 131
What is AWS DataSync?

AWS DataSync is an online data transfer service that simplifies, automates, and accelerates moving data between on-premises storage systems and AWS storage services, and also between AWS storage services. DataSync can copy data between Network File System (NFS), Server Message Block (SMB) file servers, self-managed object storage, AWS Snowcone, Amazon Simple Storage Service (Amazon S3) buckets, Amazon EFS file systems, and Amazon FSx for Windows File Server file systems.

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 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.
- **Archiving cold data** – Move cold data stored in on-premises storage directly to durable and secure long-term storage such as Amazon S3 Glacier or S3 Glacier Deep Archive. This can free up on-premises storage capacity and shut down legacy systems.
- **Data protection** – Move data into any Amazon S3 storage class, choosing the most cost-effective storage class for your needs. You can also send data to Amazon EFS or FSx for Windows File Server for a standby file system.
- **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 machine learning in the life sciences industry, video production in media and entertainment, big data analytics in financial services, and seismic research in the oil and gas industry.

Benefits

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

- **Simplify and automate data movement** – AWS DataSync makes it easier to move data over the network between on-premises storage and AWS storage services, and also between AWS storage services. DataSync automates both the management of data transfer processes and the infrastructure required for high-performance and secure data transfer.
- **Transfer data securely** – DataSync provides end-to-end security, including encryption and integrity validation, to help ensure that your data arrives securely, intact, and ready to use. DataSync accesses your AWS storage using built-in AWS security mechanisms such as AWS Identity and Access Management (IAM) roles. It also supports VPC endpoints, giving you the option to transfer data without traversing the public internet, and further increasing the security of data copied online.
• **Move data faster** – With DataSync, you can transfer data rapidly over the network into AWS. It uses a purpose-built network protocol and a parallel, multi-threaded architecture to accelerate your transfers. This speeds up migrations, recurring data processing workflows for analytics and machine learning, and data protection processes.

• **Reduce operational costs** – You can move data cost-effectively with the flat, per-gigabyte pricing of DataSync. You can save on script development, and deployment and maintenance costs, and avoid the need for costly commercial transfer tools.

### Additional AWS DataSync resources

We recommend that you read the following:

- **DataSync resources** – The resources page includes blogs, videos, and other training materials.
- **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. 6)

AWS DataSync architecture

Topics

• Data transfer between self-managed storage and AWS (p. 3)
• Data transfer between AWS storage services (p. 4)
• Data transfer using a DataSync EC2 agent deployed in a Region (p. 5)

The architectural diagrams show how DataSync transfers data between self-managed 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. 70).

Data transfer between self-managed storage and AWS

The following diagram shows a high-level view of the DataSync architecture for transferring files between self-managed storage and AWS services.
Data transfer between AWS storage services

The following diagram provides a high-level view of the DataSync architecture for transferring files between AWS services within the same AWS account. This architecture applies to both in-Region and cross-Region transfers.
Important
When you use DataSync to copy files or objects between AWS Regions, you pay for data transfer between Regions. This is billed as data transfer OUT from your source Region to your destination Region. For more information, see Data transfer pricing.

Data transfer using a DataSync EC2 agent deployed in a Region

You can use DataSync to transfer data between AWS services in different AWS accounts, or between self-managed file systems in AWS and Amazon S3, by deploying the DataSync Amazon EC2 agent in an AWS Region. For more information, see Using the DataSync EC2 agent deployed in AWS Regions (p. 93).

Components and terminology

The components of DataSync include the following:

- **Agent** – A virtual machine (VM) that is used to read data from or write data to a self-managed location. An agent is not required when transferring between AWS storage services in the same AWS account.
- **Location** – Any source or destination location that is used in the data transfer (for example, Amazon S3, Amazon EFS, FSx for Windows File Server, NFS, SMB, or self-managed object storage).
- **Task** – Consists of a source location and a destination location, and configuration that defines how data is transferred. A task always transfers data from the source to the destination. Configuration can include options such as task schedule, bandwidth limit, etc. A task is the complete definition of a data transfer.
- **Task execution** – An individual run of a task, which includes information such as start time, end time, bytes written, and status.

Agent

An *agent* is a VM that you own that is used to read or write data from self-managed storage systems. The agent can be deployed on VMware ESXi, KVM, Microsoft Hyper-V hypervisors, or it can be launched as an Amazon EC2 instance. 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. 89).

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 a destination location. AWS DataSync supports Network File System (NFS), Server Message Block (SMB), self-managed object storage, Amazon EFS, FSx for Windows File Server, and Amazon S3 as location types. For more information, see Working with locations (p. 70).

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.

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

For detailed information about these phases and statuses, see Understanding task execution statuses (p. 54).

**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 and metadata of files on 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. 61).

After the scanning is done and the differences are calculated, DataSync transitions to the **TRANSFERRING** status. At this point, DataSync starts transferring files and metadata from the source file system to the destination. DataSync copies changes to files with contents or metadata that are different between the source and the destination. You can narrow down the copied files by filtering the data or by configuring DataSync to **not overwrite files that are already present on the destination**.

**Note**

By default, any changes to metadata on the source storage result in this metadata being copied to the destination storage.

After the **TRANSFERRING** phase is done, DataSync verifies consistency between the source and destination file systems. This is the **VERIFYING** phase.
When 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. This additional check can verify the entire dataset or only the files that were transferred as part of the task execution. For most use cases, we recommend verifying only the files transferred.

**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. 54) and Enable verification in the Configuring task settings (p. 61) 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. 135).

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. 10)
- 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 6.5, 6.7, or 7.0) – 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.

- Microsoft Hyper-V Hypervisor (version 2012 R2 or 2016) – A free, standalone version of Hyper-V is available at the Microsoft Download Center. For this setup, you need a Microsoft Hyper-V Manager on a Microsoft Windows client computer to connect to the host.
  
  **Note**
  
  The DataSync VM is a generation 1 virtual machine. For more information about the differences between generation 1 and generation 2 VMs, see Should I create a generation 1 or 2 virtual machine in Hyper-V?

- Linux Kernel-based Virtual Machine (KVM) – A free, open-source virtualization technology. KVM is included in Linux versions 2.6.20 and newer. AWS DataSync is tested and supported for the CentOS/RHEL 7.8, Ubuntu 16.04 LTS, and Ubuntu 18.04 LTS distributions. Any other modern Linux distribution might work, but function or performance is not guaranteed. We recommend this option if you already have a KVM environment up and running and you're already familiar with how KVM works.
Note
Running KVM on Amazon EC2 is not supported, and cannot be used for DataSync agents. To run the agent on Amazon EC2, deploy an agent AMI. For more information about deploying an agent AMI on Amazon EC2, see Deploy your agent as an Amazon EC2 instance to access in-cloud file systems (p. 21).

- Amazon EC2 instance – DataSync provides an Amazon Machine Image (AMI) that contains the DataSync VM image. For the recommended instance types, 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 that transfer up to 20 million files.
  - 64 GB of RAM assigned to the VM, for tasks that 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.

If you are running DataSync on your AWS Snowcone device, choose the default instance, snc1.medium, which provides 2 CPU cores and 4 GiB of memory.

Warning
We don't recommend using a DataSync agent running on an Amazon EC2 instance to access your on-premises storage, due to increased network latency. To transfer data between your on-premises storage and AWS using DataSync, deploy the agent as a virtual machine in your data center as close to your on-premises storage as possible.

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 self-managed storage (p. 11)
- 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 self-managed storage

Deploy the DataSync agent close to your self-managed storage to minimize network latency when using native protocols such as Network File System (NFS) and Server Message Block (SMB). Doing this ensures that files travel over the network between the DataSync agent and the DataSync service using our purpose-built, accelerated protocol, which significantly speeds up transfers.

The following ports are required for communication between the DataSync agent and your NFS, Server Message Block (SMB), or Amazon S3 API compatible 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.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Supports NFS v3.x, NFS v4.0, and NFS v4.1.</td>
</tr>
<tr>
<td>Agent</td>
<td>SMB server</td>
<td>TCP/UDP</td>
<td>139 (SMB)</td>
<td>By the DataSync agent to mount a source SMB file share.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>or 445 (SMB)</td>
<td>Supports SMB 2.1 and SMB 3 versions.</td>
</tr>
<tr>
<td>Agent</td>
<td>Self-managed object</td>
<td>TCP</td>
<td>443 (HTTPS)</td>
<td>By the DataSync agent to access your self-managed object storage.</td>
</tr>
<tr>
<td></td>
<td>storage</td>
<td></td>
<td>or 80 (HTTP)</td>
<td></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 self-managed 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>
</tbody>
</table>
## Network requirements when using VPC 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>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>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Note</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Alternatively, you can obtain the activation key from the agent's local console. This method does not require connectivity between the browser and your agent. For more information about using the local console to get the activation key, see Obtaining an activation key using the local console (p. 98)</td>
</tr>
</tbody>
</table>

To find the correct IP address, open the Amazon VPC console, 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. 90).
Network requirements when using VPC 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>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></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>To find the related IP addresses, open the Amazon EC2 console and choose Network Interfaces 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. 90).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agent</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>

Following is an illustration of the ports required by DataSync when using private endpoints.
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</td>
<td>(HTTP) Used 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. <strong>Note</strong> Alternatively, you can obtain the activation key from the agent's local console. This method does not require connectivity between the browser and your agent. For more information about using the local console to get the activation key, refer to your DataSync guide.</td>
<td></td>
</tr>
</tbody>
</table>

**Note**
Alternatively, you can obtain the activation key from the agent's local console. This method does not require connectivity between the browser and your agent. For more information about using the local console to get the activation key, refer to your DataSync guide.
### 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</td>
<td>Used by the DataSync agent to activate with your AWS account. This is for agent activation only. You can block the endpoints after activation.</td>
<td>For public endpoint activation:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(HTTPS)</td>
<td></td>
<td>activation.datasync.$region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>For FIPS endpoint activation:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>activation.datasync-fips.$region.amazonaws.com</td>
</tr>
<tr>
<td>Agent</td>
<td>AWS</td>
<td>TCP</td>
<td>443</td>
<td>For communication between the DataSync agent and the AWS service endpoint. For information about Regions and service endpoints, see Choose a service endpoint (p. 24).</td>
<td>API endpoints:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(HTTPS)</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>$region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Data transfer endpoints for FIPS:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>cp.datasync-fips.$region.amazonaws.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>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>
</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>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>
<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>Support channel 54.201.223.107</td>
</tr>
<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>NTP 0.amazon.pool.ntp.org 1.amazon.pool.ntp.org 2.amazon.pool.ntp.org 3.amazon.pool.ntp.org</td>
</tr>
</tbody>
</table>

**Note**
If you want to change the default NTP configuration of your VM agent to use a different NTP server using the local console, see Configuring a Network Time Protocol (NTP) server for VMware agents (p. 103).

Following is an illustration of the ports required by DataSync when using public service endpoints or FIPS endpoints.
Network requirements 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. 107) 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 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. 18)
- Configure a source location (p. 26)
- Configure a destination location (p. 28)
- Configure Task Settings (p. 28)
- 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 self-managed storage, you first deploy and activate an AWS DataSync agent. The activation process associates your agent with your AWS account. An agent isn't required when transferring between AWS storage services in the same AWS account. To set up a data transfer between two AWS services, see Configure a source location (p. 26).

Topics
- Deploy an AWS DataSync agent (p. 19)
- Choose a service endpoint (p. 24)
Deploy an AWS DataSync agent

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. 19)
- Deploy your DataSync agent on KVM (p. 19)
- Deploy your DataSync agent on Hyper-V (p. 20)
- Deploy your agent as an Amazon EC2 instance to access in-cloud file systems (p. 21)
- Deploy your DataSync agent on an AWS Snowcone device (p. 23)
- Deploy your DataSync agent on AWS Outposts (p. 23)

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 self-managed 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 Amazon EC2 instance to access in-cloud file systems (p. 21).

AWS DataSync currently supports the VMware ESXi hypervisor. For information about hardware requirements for the VM, see Virtual machine requirements (p. 10). 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. 26) 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. 24).

Deploy your DataSync agent on KVM

You can download and deploy an AWS DataSync agent in your KVM 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 self-managed storage and if it's activated in the same AWS Region.
To deploy an agent on KVM

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 in a .zip file that contains a .qcow2 image file that can you can deploy in your KVM 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 Amazon EC2 instance to access in-cloud file systems (p. 21).

AWS DataSync currently supports the KVM hypervisor. For information about hardware requirements for the VM, see Virtual machine requirements (p. 10).

To get started installing your .qcow2 image for use in KVM, use the following command.

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

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

If you previously activated an agent in this AWS Region and want to use that agent, choose that agent, and then choose Create agent. The Configure a source location (p. 26) 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 deploy an agent, you choose a service endpoint (p. 24).

Deploy your DataSync agent on Hyper-V

You can download and deploy an AWS DataSync agent in your Hyper-V 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 self-managed storage and if it’s activated in the same AWS Region.

To deploy an agent on Hyper-V

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 in a .zip file that contains a .vhd image file that can you can deploy in your Hyper-V 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 Amazon EC2 instance to access in-cloud file systems (p. 21).

AWS DataSync currently supports the Hyper-V hypervisor. For information about hardware requirements for the VM, see Virtual machine requirements (p. 10). For information about how to deploy a .vhdx file in a Hyper-V host, see the documentation for your hypervisor.

If you previously activated an agent in this AWS Region and want to use that agent, choose that agent, and then choose Create agent. The Configure a source location (p. 26) 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 deploy an agent, you choose a service endpoint (p. 24).

Deploy your agent as an Amazon EC2 instance to access in-cloud file systems

You can deploy the DataSync Amazon EC2 instance in an AWS Region to transfer data from one AWS account to another, or from a self-managed, in-cloud file system. Doing this enables you to perform the following tasks:

• Transfer data between Amazon EFS or FSx for Windows File Server file system to AWS storage in a different AWS account.
• Transfer data from self-managed file systems to AWS storage services.

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 Amazon EFS or FSx for Windows File Server file systems in different AWS accounts, or from a self-managed file system to a managed file system, create the Amazon EC2 agent in the source AWS Region.
• To copy from Amazon S3, create the agent in the destination AWS Region.

For more information about these use cases, including high-level architecture diagrams, see Using the DataSync EC2 agent deployed in AWS Regions (p. 93).

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

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

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

*We don't recommend using a DataSync agent running on Amazon EC2 to access your on-premises storage, due to increased network latency. To transfer data between your on-premises storage and AWS using DataSync, we recommend deploying the agent as a virtual machine in your data center as close to your on-premises storage as possible.*

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

- Use the following CLI command to programmatically get the latest DataSync AMI ID for the specified Region.

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

**Example Example command and output**

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

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

For the recommended instance types, see *Amazon EC2 instance requirements (p. 10).*

If you activate an agent in this Region that has access to your file system using a mount target in the same Availability Zone and you want to use that agent, choose the agent, and then choose** Create agent**. The Configure a source location (p. 26) page appears.

**To deploy your DataSync agent as an Amazon EC2 instance**

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

```bash
```

In the URL, replace the ```source-file-system-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. Use the ```.AMI-command``` CLI command described in the preceding section to find the DataSync AMI ID for a specified AWS Region.

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:
   - 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 file system, 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 Amazon 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 file system allows inbound traffic from the agent. In addition, make sure that the agent allows outbound traffic to the source file system.

If you deploy your agent using a VPC endpoint, you need to allow additional ports. For more information, see How DataSync works with VPC endpoints (p. 90).

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 Amazon 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’s assigned a public Domain Name System (DNS) name and IP address, you can find these 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.

### Deploy your DataSync agent on an AWS Snowcone device

The DataSync agent AMI is pre-installed on your Snowcone device. You can use AWS OpsHub for Snow Family or the AWS Snowball Edge CLI command line tool to launch the agent and attach a virtual interface to the agent. Then, use the virtual interface's IP address to activate the agent.

For instructions on launching the agent using AWS OpsHub, see Using DataSync to transfer files to AWS.

For instructions on launching the agent using the Snowball CLI, see Launching AWS DataSync AMI.

For information about using the AWS Snowcone client, see Using the Snowcone client.

### Deploy your DataSync agent on AWS Outposts

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

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

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. 24).
- 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 self-managed 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. 90).

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. 24)
- Choose a FIPS service endpoint (p. 24)
- Choose a VPC endpoint (p. 25)

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. 26)

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. 26)**

### 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 self-managed 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**.

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)

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.

   For additional information about using DataSync in a VPC, see **Using AWS DataSync in a virtual private cloud** (p. 90).

**Next step: the section called “Activate your agent” (p. 26)**
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. 89).

   **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. 90).

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 verify your firewall allows the required ports.

   Alternatively, you can obtain the activation key from the agent's local console. This method doesn't require connectivity between the browser and your agent. For more information about using the local console to get the activation key, see Obtaining an activation key using the local console (p. 98).

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. 70).

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. 71)
- Creating a location for SMB (p. 73)
- Creating a location for object storage (p. 75)
- Creating a location for Amazon EFS (p. 76)
- Creating a location for FSx for Windows File Server (p. 77)
- Creating a location for Amazon S3 (p. 79)

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's installed on-premises uses this hostname 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. 70)](#).

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

- Creating a location for **NFS** (p. 71)
- Creating a location for **SMB** (p. 73)
- Creating a location for **object storage** (p. 75)
- Creating a location for **Amazon EFS** (p. 76)
- Creating a location for **FSx for Windows File Server** (p. 77)
- Creating a location for **Amazon S3** (p. 79)

### Configure Task Settings

After you have created an AWS DataSync agent and configured the source and destination locations, you can configure the settings for a new 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 the task.

You configure task settings when creating a new task in the AWS DataSync console. You can also edit task settings by opening the AWS DataSync console at [https://console.aws.amazon.com/datasync/](https://console.aws.amazon.com/datasync/), selecting the task you want to edit, and choosing **Edit**.

On the **Configure settings** page, for **Task name - optional**, enter a name for your task. **Task name** is an optional setting.

The **Options** section contains configuration options for running your task. The following sections provide more details about these options.

**Topics**

- Data verification options (p. 29)
- Ownership and permissions related options (p. 29)
- File metadata options and file management (p. 30)
- Bandwidth options (p. 30)
- Filtering options (p. 30)
- Scheduling and queueing options (p. 31)
- Tags and logging options (p. 31)
Data verification options

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. This additional check can verify the entire dataset or only the files that were transferred as part of the task execution. For most use cases, we recommend verifying only the files transferred.

Task data verification options specify how to verify data that's transferred by the task.

Data verification options are as follows:

- **Verify only the data transferred (recommended)** – This option 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 Amazon S3 storage classes in DataSync (p. 81).

- **Verify all data in the destination** – This option 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 Amazon S3 storage classes in DataSync (p. 81).

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

Ownership and permissions related options

DataSync preserves metadata between storage systems that have similar metadata structures. Different options are used to configure such metadata preservation depending on the storage system type.

**When copying data between NFS, Amazon EFS, and Amazon S3** choose one of the following if applicable:

- Choose **Copy ownership** to have DataSync copy POSIX file and folder ownership, such as the group ID of the file's owners and the user ID of the file's owner.
- Choose **Copy permissions** to have DataSync copy POSIX permissions for files and folders from the source to the destination.

**When copying between SMB and FSx for Windows File Server, or between two FSx for Windows File Server locations**, choose one of the following if applicable:

- Choose **Copy ownership, DACLS, and SACLs** to have DataSync copy the following:
  - The object owner.
  - NTFS discretionary access lists (DACLs), which determine whether to grant access to an object.
  - NTFS system access control lists (SACLs), which are used by administrators to log attempts to access a secured object.

  Copying SACLs requires granting additional permissions to the Windows user that DataSync uses to access your SMB location. See .SMBuser to learn more about choosing a user name that ensures sufficient permissions to files, folders, and metadata.

- Choose **Copy ownership and DACLS** to have DataSync copy the following:
  - The object owner.
  - NTFS discretionary access lists (DACLs), which determine whether to grant access to an object.
DataSync won’t copy NTFS system access control lists (SACLs) when you choose this option.

- Choose **Do not copy ownership or ACLs** if you want DataSync to not copy any ownership or permissions data. The objects that DataSync writes to your destination location are owned by the user whose credentials are provided for DataSync to access the destination location. Destination object permissions are determined based on the permissions configured on the destination server.

For more information about metadata preservation using DataSync see [How DataSync handles metadata and special files](p. 85).

### File metadata options and file management

You can configure DataSync tasks to copy file metadata, to keep deleted files, and to overwrite files in the destination. These option settings are as follows:

- Choose **Copy timestamps** to have DataSync copy the timestamp metadata from the source to the destination.
- Choose **Keep deleted files** to have DataSync keep 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 Amazon S3 storage classes in DataSync](p. 81).

- Choose **Overwrite files** if you want files at the destination to be overwritten by files from the source when the source data or metadata is different.

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 additional charges for certain storage classes (for example, for retrieval or early deletion). For detailed information, see [Considerations when working with Amazon S3 storage classes in DataSync](p. 81).

### Bandwidth options

You can configure a bandwidth limit for DataSync tasks. Bandwidth limit options are as follows:

- Choose **Use available** to have DataSync use all the network bandwidth that is available for the transfer.
- Choose **Set bandwidth limit (MiB/s)** to limit the maximum bandwidth that you want DataSync to use for this task.

You can change bandwidth limits for an in-progress task execution. For more information, see [Adjusting bandwidth throttling for a task execution](p. 68).

### Filtering options

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. The configuration options for filtering are as follows:

- In the **Data transfer configuration** section, use the **Exclude patterns** section to specify files, folders, and objects to exclude from your transfer. To include specific files, folders, and objects in your transfer, select **Specific files and folders** and then use the **Include patterns** section.
• To add additional patterns to your filters, choose Add pattern. For detailed information about filtering and syntax for creating patterns, see Filtering the data transferred by AWS DataSync (p. 62).
• You can modify filter patterns when you edit a task. You can also specify different patterns each time that you execute a task.

Scheduling and queueing options

You can schedule a DataSync task to be run at a specific time. If you are using a single agent to run multiple tasks, you can queue those tasks. Configuring options for scheduling are as follows:

• In the Schedule (optional) section, configure your task to run on a schedule that you specify, with a minimum interval of 1 hour.
• For Frequency, configure how frequently you want the task to run. For frequency configuration options, see Configuring a task schedule (p. 66).

If you are using a single agent to run multiple tasks, choose Queueing to make the tasks run in series (first in, first out). For more information, see Queueing task executions (p. 62).

Tags and logging options

You can add one or more tags to a DataSync task. A tag is a key-value pair that is associated with the task. You can also choose logging options to have DataSync publish logs for individual files or objects to the CloudWatch log group that you specify. Tags and logging options are as follows:

• 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.
• Choose Task logging - optional to have DataSync publish 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. If you don't have a policy in the current Region, a check box appears so that you can create the required policy automatically. For an example of such a policy, see Allowing DataSync to upload logs to Amazon CloudWatch log groups (p. 52).

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:

• Choose Log basic information such as transfer errors to publish only basic information (such as transfer errors) to CloudWatch.
• Choose Log all transferred objects, files, and folders to publish log records to CloudWatch Logs for all files or objects that the task copies and integrity checks.
• Choose Do not send logs to CloudWatch if you don't want DataSync logs to be published to CloudWatch.

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. 55).
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 with the default configuration

1. When the Status of the task changes from Creating to Available, choose Start, and then Start with defaults.
2. The task starts and you are redirected to the Task execution page.

To start your task with a modified configuration

1. When the Status of the task changes from Creating to Available, choose Start, and then Start with overriding options.
2. The Start task page displays the default configuration. You can choose to modify these settings before starting the task. Any changes that you make are applied only to this task execution.
3. After you review and make any changes to your task configuration, choose Start. The task starts and you are redirected to the Task execution page.

When you create a task, it first enters the Creating state. While the task is in the Creating state, AWS DataSync performs validation checks on the source and destination locations. After DataSync validates the locations, the task transitions to the Available state. If an agent on the source location goes offline, the task transitions to the Unavailable state.

For information about how DataSync transfers files, see How DataSync transfers files (p. 6).

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. 67).
2. Delete locations that you don't need. For instructions on how to delete a location, see Deleting a location (p. 88).
3. Delete agents that you don't need. For instructions about how to delete an agent, see Deleting an agent (p. 96).
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 the service works. We also recommend reading Using identity-based policies (IAM policies) for DataSync (p. 107) 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 AWS 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.

```
aws datasync help
```

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

```
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. 44)
- Step 4: Start a task execution (p. 45)
- Step 5: Use the CLI to monitor your task execution (p. 46)
- API filters for ListTasks and ListLocations (p. 47)

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**

To access your self-managed storage, you first deploy and activate an AWS DataSync agent. The activation process associates your agent with your AWS account. An agent isn’t required when transferring between AWS storage services within the same AWS account. To set up a data transfer between two AWS services, see Step 2: Create locations (p. 36).

A DataSync agent can transfer data through public service endpoints, Federal Information Processing Standard (FIPS) endpoints, and Amazon VPC endpoints. For more information, see Creating and activating an agent (p. 89).
Note
When you configure your agent to use Amazon 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. 90).

To create an agent to read from an NFS, SMB, or self-managed object storage 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 console. For information about how to get the .ova image or EC2 AMI, see Create an agent (p. 18). For information about hardware requirements and recommended EC2 instance types, see Virtual machine requirements (p. 10).

   Important
   If you are deploying your agent on Amazon EC2, deploy the agent such that it doesn't require network traffic between Availability Zones (to avoid charges for such traffic).

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

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

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 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.

   ```
   $ 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 required mainly for troubleshooting, network-specific settings, and so on.
   You log in to the agent VM local console using your VM's hypervisor client. For information about how to use the VM local console, see Working with your agent on the local console (p. 98).

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.
     ```
     $ curl "http://agent-ip-address/?gatewayType=SYNC&activationRegion=aws-region&no_redirect"
     ```
   - To activate an agent using a Amazon Virtual Private Cloud (Amazon VPC) endpoint, use the IP address of the Amazon VPC endpoint. Use the following command.
     ```
     $ curl "http://agent-ip-address/?gatewayType=SYNC&activationRegion=aws-region&privateLinkEndpoint=IP address of VPC endpoint&endpointType=PRIVATE_LINK&no_redirect"
     ```
AWS DataSync User Guide
Step 1: Create an agent

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 Amazon VPC endpoint setup.

For more information about Amazon VPC endpoint configuration, see step 5 in Configuring DataSync to use private IP addresses for data transfer (p. 90).

- 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.

```bash
$ 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.

FOEFT-7FPFR-GG7MC-39923-27DOH

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.

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

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

```bash
$ 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 Amazon VPC endpoint configuration, see step 5 in Configuring DataSync to use private IP addresses for data transfer (p. 90).

- **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. 90).

- **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. 90).

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

```json
{"AgentArn": "arn:aws:datasync:us-east-1:11122233444:agent/agent-0b0addbeef44baca3"}
```

**Note**
After you choose a service endpoint, you can’t change it later.
After you activate the agent, it closes port 80 and the port is no longer accessible. If you can't connect to the agent after you have activated it, verify the activation was successful by using this command:

```bash
$ 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. 70).

### Topics
- Create an NFS location (p. 36)
- Create an SMB location (p. 37)
- Create an object storage location (p. 38)
- Create an Amazon EFS location (p. 38)
- Create an FSx for Windows File Server location (p. 40)
- 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. 71).

**Note**
If you are using an NFS location on an AWS Snowcone device, see NFS server on AWS Snowcone (p. 72) for more information about transferring data to or from that device.

### 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
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. 73).

To create an SMB location using the CLI

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

```
$ 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. 265)` API operation.

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

```
{ "LocationArn": "arn:aws:datasync:us-east-1:11122233444:location/loc-0f01451b140b2af49" }
```
Create an object storage location

Use the following procedure to create a self-managed object storage location using the AWS CLI. An object storage location is the endpoint for an Amazon S3 API compatible object storage server. An object storage location defines an object storage server that can be read from or written to. You can also create an object storage location using the AWS Management Console.

For more information about object storage locations, including compatibility requirements, see Creating a location for object storage (p. 75).

To create a self-managed object storage location using the CLI

- Use the following command to create a self-managed object storage location.

  ```bash
  $ aws datasync create-location-object-storage --server-hostname object-storage-server.example.com --bucket-name myBucket --agent-arns arn:aws:datasync:us-east-1:123456789012:agent/agent-01234567890deadfb
  ```

  The following parameters are required in the `create-location-object-storage` command.

  - `server-hostname`: The DNS name or IP address of the self-managed object storage server.
  - `bucket-name`: The name that identifies the bucket on the self-managed object storage server at the location.
  - `agent-arns`: The ARNs of the agents to use for the self-managed object storage location.

  If your object storage requires a user name and password to authenticate, use the `--access-key` and `--secret-key` parameters to provide the user name and password, respectively.

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

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

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. 76).

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-22334a10",
            "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.

```bash
```
Create an FSx for Windows File Server location

Use the following procedure to create an FSx for Windows File Server location using the AWS CLI. An Amazon FSx location is the endpoint for an 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 FSx for Windows File Server (p. 77).

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

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

```bash
$ aws datasync create-location-fsx-windows 
  --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 Amazon 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 S3 bucket, see Creating a bucket in the Amazon S3 User Guide.

For DataSync to access an 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, the required IAM policies, and the S3 location using the AWS CLI.
AWS Security Token Service (AWS STS) must be activated in your account and Region for DataSync to assume the IAM role. For more information about temporary security credentials, see Temporary security credentials in IAM in the IAM User Guide.

You can also create an S3 location using the console. For more information, see Creating a location for Amazon S3 (p. 79).

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}
   {
     "Role": {
       "Path": "/",
       "RoleName": "YourBucketAccessRole",
       "RoleId": "role-id",
       "Arn": "arn:aws:iam::account-id:role/YourBucketAccessRole",
       "CreateDate": "2018-07-27T02:49:23.117Z",
       "AssumeRolePolicyDocument": {
         "Version": "2012-10-17",
         "Statement": [ ...
   ```
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. The following example shows the minimum permissions needed for DataSync to read and write to an S3 bucket in an AWS Region.

```json
{
   "Version": "2012-10-17",
   "Statement": [
       {
         "Action": [
             "s3:GetBucketLocation",
             "s3:ListBucket",
             "s3:ListBucketMultipartUploads"
         ],
         "Effect": "Allow",
         "Resource": "YourS3BucketArn"
       },
       {
         "Action": [
             "s3:AbortMultipartUpload",
             "s3:DeleteObject",
             "s3:GetObject",
             "s3:ListMultipartUploadParts",
             "s3:PutObjectTagging",
             "s3:GetObjectTagging",
             "s3:PutObject"
         ],
         "Effect": "Allow",
         "Resource": "YourS3BucketArn/"
       }
   ]
}
```

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/YourPolicyName'
```

For Amazon S3 buckets on AWS Outposts, use the following policy.

```json
{
   "Version": "2012-10-17",
   "Statement": [
       {
         "Action": [
             "s3-outposts:ListBucket",
             "s3-outposts:ListBucketMultipartUploads"
         ]
       }
   ]
}
```
"Effect": "Allow",
"Resource": [
  "s3OutpostsBucketArn",
  "s3OutpostsAccessPointArn"
],
"Condition": {
  "StringLike": {
    "s3-outposts:DataAccessPointArn": "s3OutpostsAccessPointArn"
  }
}
},
{
  "Action": [
    "s3-outposts:AbortMultipartUpload",
    "s3-outposts:DeleteObject",
    "s3-outposts:GetObject",
    "s3-outposts:ListMultipartUploadParts",
    "s3-outposts:PutObjectTagging",
    "s3-outposts:GetObjectTagging",
    "s3-outposts:PutObject"
  ],
  "Effect": "Allow",
  "Resource": [
    "s3OutpostsBucketArn/*",
    "s3OutpostsAccessPointArn"
],
  "Condition": {
    "StringLike": {
      "s3-outposts:DataAccessPointArn": "s3OutpostsAccessPointArn"
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "s3-outposts:GetAccessPoint"
  ],
  "Resource": "s3OutpostsAccessPointArn"
}
]}

5. Create the S3 location.

Use the following command to create your Amazon S3 location.

```
$ aws datasync create-location-s3 --s3-bucket-arn 'arn:aws:s3:::bucket' --s3-storage-class 'your-S3-storage-class' --s3-config 'BucketAccessRoleArn=arn:aws:iam::account-id:role/role-allowing-DS-operations' --subdirectory /your-folder
```

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

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

The location type information is encoded in the LocationUri. In this example, the `s3://` prefix in LocationUri shows the location's type.
If your Amazon S3 bucket is located on an AWS Outposts, you must deploy an Amazon EC2 agent on your Outpost. The agent must be in a Amazon VPC that is allowed to access the access point specified in the command. The agent also must be activated in the Outpost’s parent Region, and be able to route to the Amazon S3 on AWS Outposts endpoints for the bucket. For more information about launching a DataSync agent on AWS Outposts, see Deploy your DataSync agent on AWS Outposts (p. 23).

Use the following command to create an Amazon S3 location on your Outpost.

```bash
$ aws datasync create-location-s3 --s3-bucket-arn s3-bucket-arn access-point-arn --s3-config BucketAccessRoleArn=arn:aws:iam::account-id:role/role-allowing-DS-operations --agent-arns datasync-agent-arn-in-the-vpc-which-can-access-your-s3-access-point
```

**Note**

- Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges in the following scenarios:
  - *When using object versioning*: Changes to object data or metadata create a new version of the object.
  - *When using storage classes that can incur additional charges for overwriting, deleting, or retrieving, objects*: Changes to object data or metadata result in such charges. For more information, see Considerations when working with Amazon S3 storage classes in DataSync (p. 81).
- When you use object versioning, 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.

---

**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. 52).

3. Create a task by using the following command.

   ```bash
   ```
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.

```
```

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. 257).

```
```

When you create a task, you can optionally configure the task to include or exclude specific files, folders, and objects. For more information, see Filtering the data transferred by AWS DataSync (p. 62). You can also configure your task to run a schedule you specify, with a minimum interval of 1 hour. For more information, see Scheduling your task (p. 65).

**Note**

If a task remains in the **CREATING** status for more than a few minutes, your agent might be having trouble reaching your self-managed storage. Check the task's Error Code and Error Detail values. For example, NFS and SMB mount issues are often caused by a mistyped server hostname, or when the agent's access to your storage is blocked by firewall rules.
Step 5: Monitor your task execution

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

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

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

```json
{
   "TaskExecutionArn": "arn:aws:datasync:us-east-1:112233445566:task/task-08de6e6697796f026/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,
```

Note

Each agent can run a single task at a time.
Monitor your task execution in real time

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. 268) 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.

```bash
# 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"
```

## API filters for ListTasks and ListLocations

AWS DataSync supports filters as input arguments to the **ListTasks** and **ListLocations** API calls. This enables you to retrieve configurations of data transfer tasks, using filters such as the source or destination for the data transfer.

### Topics

- Parameters for API filtering (p. 47)
- API filtering for ListLocations (p. 48)
- API filtering for ListTasks (p. 49)

### Parameters for API filtering

You can use API filters to narrow down the list of resources returned by **ListTasks** and **ListLocations**. For example, to retrieve all your Amazon S3 locations, you can use **ListLocations** with filter name **LocationType** `S3` and **Operator** `Equals`.

To filter API results, you must specify a filter name, operator, and value.

- **Name** – The name of the filter being used. Each API call supports a list of filters that are available for it (for example, **LocationType** for **ListLocations**).
- **Values** – The values that you want to filter for. For example, you might want to display only Amazon S3 locations.
- **Operator** – The operator that is used to compare filter values (for example, **Equals** or **Contains**).

The following table lists the available operators.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Key types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>String, Number</td>
</tr>
</tbody>
</table>
API filtering for ListLocations

ListLocations supports the following filter names:

- **LocationType** – Filters on the location type: SMB, NFS, S3, FSXW_WINDOWS, and OBJECT_STORAGE.
- **LocationUri** – Filters on the uniform resource identifier (URI) assigned to the location, as returned by the DescribeLocation* API call (for example, `s3://bucket-name/your-prefix` for Amazon S3 locations).
- **CreationTime** – Filters on the time that the location was created. The input format is `yyyy-MM-dd:mm:ss` in Coordinated Universal Time (UTC).

The following AWS CLI example lists all locations of type Amazon S3 that have a location URI starting with the string `s3://DOC-EXAMPLE-BUCKET`, and that were created at or after 2019-12-15 17:15:20 UTC.

```
aws datasync list-locations --filters Name=LocationType,Values=s3,Operator=Equals
Name=LocationUri,Values=s3://DOC-EXAMPLE-BUCKET,Operator=BeginsWith
Name=CreationTime,Values=2019-12-15 17:15:20,Operator=GreaterThanOrEqual
```

This command returns output similar to the following.

```json
{
   "Locations": [ 
      {
         "LocationArn": "arn:aws:datasync:us-east-1:111122223333:location/loc-333333333abcdef0",
         "LocationUri": "s3://DOC-EXAMPLE-BUCKET-examples/"
      },
      {
         "LocationArn": "arn:aws:datasync:us-east-1:123456789012:location/loc-987654321abcdef0",
         "LocationUri": "s3://DOC-EXAMPLE-BUCKET-examples-2/"
      }
   ]
}
```
API filtering for ListTasks

ListTasks supports the following filter names.

- **LocationId** – Filters on both source and destination locations on Amazon Resource Name (ARN) values.
- **CreationTime** – Filters on the time that the task was created. The input format is `yyyy-MM-dd:mm:ss` in UTC.

The following AWS CLI example shows the syntax when filtering on **LocationId**.

```
aws datasync list-tasks --filters Name=LocationId,Values=arn:aws:datasync:us-east-1:your-account-id:location/your-location-id,Operator=Contains
```

The output of this command looks similar to the following.

```
{
   "Tasks": [
      {
         "TaskArn": "arn:aws:datasync:us-east-1:your-account-id:task/your-task-id",
         "Status": "AVAILABLE",
         "Name": "DOC-EXAMPLE-BUCKET"
      }
   ]
}
```
Monitoring your task

Topics

• Accessing Amazon CloudWatch metrics for DataSync (p. 50)
• DataSync CloudWatch metrics (p. 50)
• CloudWatch events for DataSync (p. 51)
• DataSync dimensions (p. 52)
• Allowing DataSync to upload logs to Amazon CloudWatch log groups (p. 52)

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. 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.

These statistics are retained for a period of 15 months.

<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>BytesVerifiedDestination</td>
<td>The total number of bytes of data that are verified at the destination location.</td>
</tr>
</tbody>
</table>
### CloudWatch events for DataSync

Amazon 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. Events are emitted on a best effort basis.

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. 96).</td>
</tr>
<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 and has been out of contact with the service for 5 minutes or longer. When the issue</td>
</tr>
</tbody>
</table>
### State Changes for a Location

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDING</td>
<td>DataSync is adding a location.</td>
</tr>
<tr>
<td>AVAILABLE</td>
<td>The location is created and is available to use.</td>
</tr>
</tbody>
</table>

### State Changes for a Task

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATING</td>
<td>DataSync attempts to mount the Network File System (NFS) location and create the task.</td>
</tr>
<tr>
<td>RUNNING</td>
<td>DataSync has mounted the source and it is functioning properly.</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>The task is not configured properly and is not available for use. If an agent that is associated with a source (NFS) location goes offline, the task transitions to the UNAVAILABLE status.</td>
</tr>
</tbody>
</table>

### State Changes for a Task Execution

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAUNCHING</td>
<td>DataSync is initializing the task execution.</td>
</tr>
<tr>
<td>PREPARING</td>
<td>DataSync is computing which files need to be transferred.</td>
</tr>
<tr>
<td>TRANSFERRING</td>
<td>DataSync is performing the actual transfer of your data to AWS</td>
</tr>
<tr>
<td>VERIFYING</td>
<td>DataSync performs a full data and metadata integrity verification to ensure that the data in your destination is an exact copy of your source.</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>The transfer is successful.</td>
</tr>
<tr>
<td>ERROR</td>
<td>The sync has failed.</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. To upload logs to your log group, DataSync requires a resource policy that grants sufficient permissions. When you create a task using the AWS Management Console, DataSync can automatically create the required resource policy. For more information, see Configure Task Settings (p. 28).
The following is an example resource policy that grants such permissions.

```json
{
    "Statement": [
        {
            "Sid": "DataSyncLogsToCloudWatchLogs",
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents",
                "logs:CreateLogStream"
            ],
            "Principal": {
                "Service": "datasync.amazonaws.com"
            },
            "Condition": {
                "ArnLike": {"aws:SourceArn": ["arn:aws:datasync:region:account-id:task/*"]},
                "StringEquals": {"aws:SourceAccount": "account-id"}
            },
        }
    ],
    "Version": "2012-10-17"
}
```

The policy uses condition statements to ensure that only DataSync tasks from the specified account have access to the specified CloudWatch log group. We recommend using condition statements to protect against the confused deputy problem. For more information, see The confused deputy problem in the IAM User Guide.

To specify the DataSync task or tasks, replace `region` with the Region code for the AWS Region where the tasks are located, and replace `account-id` with the AWS account ID of the account that contains the tasks. To specify the CloudWatch log group, replace the same values. You can also modify the Resource statement to target specific log groups. For more information about using SourceArn and SourceAccount, see Global condition keys in the IAM User Guide.

To apply the policy, save this policy statement to a file on your local computer. Then run the following AWS Command Line Interface (AWS CLI) command to apply the resource policy:

```
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.

For information about CloudWatch log groups, see Working with log groups and log streams in the Amazon CloudWatch Logs User Guide.
Working with tasks

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

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. 54)
- Understanding task execution statuses (p. 54)
- Creating a task (p. 55)
- Starting a task (p. 61)
- Filtering the data transferred by AWS DataSync (p. 62)
- Scheduling your task (p. 65)
- Deleting a task (p. 67)

Understanding task creation statuses

Following, you can find information about the possible statuses (phases) a task might go through when it's 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 hostname.</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's 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's using the same agent. DataSync runs the task in the queue in series. For more information, see Queueing task executions (p. 62).</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><strong>Meaning</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>QUEUEING</strong> This is the first phase of a task execution if there is another task running and it's using the same agent. For more information, see Queueing task executions (p. 62).</td>
</tr>
<tr>
<td><strong>LAUNCHING</strong> 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><strong>PREPARING</strong> 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. 61).</td>
</tr>
<tr>
<td><strong>TRANSFERRING</strong> 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's transferred is updated in real time.</td>
</tr>
<tr>
<td><strong>VERIFYING</strong> 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 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><strong>SUCCESS</strong> 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><strong>ERROR</strong> 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. 18) walk you through the process of creating a task.

**Topics**

- Creating a task for DataSync (p. 56)
- Creating a task to transfer data between self-managed storage and AWS (p. 56)
- Creating a task to transfer between in-cloud locations (p. 56)
- Configuring task settings (p. 61)
Creating a task for DataSync

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. 70).

- Creating a location for NFS (p. 71)
- Creating a location for SMB (p. 73)
- Creating a location for object storage (p. 75)
- Creating a location for Amazon EFS (p. 76)
- Creating a location for FSx for Windows File Server (p. 77)
- Creating a location for Amazon S3 (p. 79)

When you initially create a task, it enters the **CREATING** status. During the **CREATING** status, DataSync attempts to mount the NFS, SMB, or self-managed object storage location. The task transitions to the **AVAILABLE** status without waiting for the AWS location to become available. If an agent that's associated with a self-managed 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, SMB, or self-managed object storage 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, SMB, or self-managed object storage server hostname. For troubleshooting information, see Troubleshooting AWS DataSync issues (p. 123).

Creating a task to transfer data between self-managed storage and AWS

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

**To create a task**

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. 70).
   - 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. 26).

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:

- **Data transfer from in-cloud file system to in-cloud file system or Amazon S3** (p. 93) – Transfer data from Amazon EFS to Amazon EFS, from self-managed NFS to EFS, or to Amazon S3.
Creating a task to transfer from in-cloud NFS to in-cloud NFS or Amazon 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 Amazon S3 works, see Data transfer from in-cloud file system to in-cloud file system or Amazon S3 (p. 93).

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 Amazon EC2 instance to read files from in-cloud

To deploy the DataSync agent as an Amazon 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.


   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 Amazon EC2 instance to access in-cloud file systems (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 Amazon 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.
Creating a task to transfer between in-cloud locations

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 Amazon 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's 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 storage

Next, you create a task to transfer data.

   Note
Create the task in the AWS Region and AWS account where the destination EFS or Amazon 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 Amazon 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 Amazon 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 Amazon EC2 instance (p. 57).

   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 / (forward slash) 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's 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:
     • For **Location type**, choose **EFS**.
     • Choose your destination EFS.
     • For **Mount path**, enter `/` (forward slash).
     • For **Subnet** and **Security groups**, use the default settings and choose **Next**.
   
   • If you are copying to Amazon S3, do the following:
     • For **Location type**, choose Amazon S3 bucket.
     • For **Amazon S3 bucket**, choose your source Amazon S3 bucket.
     • For **Folder**, choose a folder prefix to use for the transfer, or you can keep it blank.
     • Choose your destination Amazon 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. 52).

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 Amazon S3 to in-cloud NFS

Use the following instructions to transfer data from Amazon S3 to an in-cloud NFS file system that's located in the same AWS account and AWS Region where the agent is deployed. This approach includes transfers from Amazon S3 to EFS, or from Amazon S3 to self-managed NFS. The following diagram illustrates this type of transfer. For information about how Amazon S3 to in-cloud NFS works, see Data transfer from S3 to in-cloud file systems (p. 94).

### Deploying the DataSync Amazon EC2 agent to write to your destination location

First, deploy the DataSync Amazon 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 Amazon 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 Amazon EC2 console. For a list of AMI IDs by AWS Region, see Deploy your agent as an Amazon EC2 instance to access in-cloud file systems (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 Amazon S3 bucket resides.

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

1. Open the DataSync console in the AWS Region where your source Amazon 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 Amazon 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 **Amazon S3 bucket**, choose your source Amazon 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 `/` (forward slash) 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. 52).
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.

For more information about configuring a DataSync task, see [Configure Task Settings](p. 28).

Available options are as follows:

- **Data verification options**: Task data verification options specify how to verify data that's transferred by the task. For more information about configuring these options, see [Data verification options](p. 29).
- **Ownership and permissions related options**: DataSync preserves metadata between storage systems that have similar metadata structures. Depending on the storage system type, different options are used to configure such metadata preservation. For more information about configuring these options, see [Ownership and permissions related options](p. 29).
- **File metadata options and file management**: You can configure DataSync tasks to copy file metadata, to keep deleted files, and to overwrite files in the destination. For more information about configuring file metadata and file management options, see [File metadata options and file management](p. 30).
- **Bandwidth options**: You can configure a bandwidth limit for DataSync tasks. For more information about configuring bandwidth options, see [Bandwidth options](p. 30).
- **Filtering options**: 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 more information about configuring filtering options, see [Filtering options](p. 30).
- **Scheduling and queueing options**: You can schedule a DataSync task to be run at a specific time. If you are using a single agent to run multiple tasks, you can queue those tasks. For more information about configuring scheduling and queueing options, see [Scheduling and queueing options](p. 31).
- **Tags and logging options**: You can add one or more tags to a DataSync task. You can also choose logging options to have DataSync publish logs for individual files or objects to the CloudWatch log group that you specify. For more information about configuring tags and logging options, see [Tags and logging options](p. 31).

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.

You can queue multiple executions of the same task by using different filter settings for each task execution. You can configure filter settings for a task execution using the **Start with overrides** option when you start a task. For more information about filters, see Filtering the data transferred by AWS DataSync (p. 62).

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 or an execution using different filters, the current task's execution is automatically queued. After a task execution finishes, DataSync runs the next queued execution. If you want to remove a task execution from the queue yourself, 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 a subset of the files in your source location. You can use filters to specify files, folders, and objects to exclude or include in your transfer. For example, if your source location includes temporary files that end with `.tmp`, you can create an exclude filter that ensures that these files aren't transferred to the destination.

Filtering is an optional configuration to specify if you want to transfer a subset of your source files. You can keep the filter configuration empty if you want to transfer all files from the source location to the destination location. You can also use exclude and include filters together in the same task.

Topics
- Filtering terms, definitions, and syntax (p. 62)
- Excluding data from a transfer (p. 63)
- Including data in a transfer (p. 64)
- Sample filters for common uses (p. 64)

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` or `/folderA|/folderB`

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

**Pattern**

A pattern within a filter. For example, `* . tmp` is a pattern that's 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 your source location and task and specify
excluding data from a transfer

the include filter /transfer_this/. In this case, DataSync transfers only the directory /my_source/transfer_this/ and its contents.

- To specify a folder directly under the source location, include a forward slash (/) in front of the folder name. In the example preceding, the pattern uses /transfer_this, not transfer_this.
- DataSync interprets the following patterns the same way and matches both the folder and its content.

/dir
/dir/

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

Special characters

Following are special characters for use with filtering.

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

Include filters define files, folders, and objects that DataSync transfers when you run a task. You can configure include filters when you create, edit, or start a task.

To create a task with an include filter, choose the Specific files and folders option, and then specify a list of patterns to include under Include patterns.

DataSync scans and transfers only files and folders that match the include filters. For example, to include a subset of your source folders, you might specify /important_folder_1|/important_folder_2.

After you have created a task, you can edit the task configuration to add or remove patterns from the include filter. Any changes that you make are applied to future executions of the task.

When you run a task, you can modify the include filter patterns by using the Start with overrides option. Any changes that you make are applied only to that execution of the task.

You can also use the AWS CLI to create or edit an include filter. The following example shows such a CLI command.

```
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**
Include filters support the wildcard (*) character only as the rightmost character in a pattern. For example, /documents*/code* is supported, but *.txt isn't supported.

Sample filters for common uses

In this section, you can find common uses for filtering and sample filters for them.

Exclude some folders from your source location
In some cases, you might exclude folders in your source location to not copy them to your destination location. For example, you might have temporary work-in-progress folders. Or you might use a NetApp system 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 a subset of the folders on your source location

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

/folder-to-transfer/*

Exclude specific file types

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

Transfer individual files you specify

To transfer a list of individual files, start a task execution with an include filter like the following:
"/folder/subfolder/file1.txt|/folder/subfolder/file2.txt|/folder/subfolder/file2.txt"

Note
The filter string length is limited to 409,600 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, with a minimum interval of 1 hour. 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 *

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

**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, with a minimum interval of 1 hour. Then enter your expression in the **Cron expression** box.

For detailed information about schedule expressions, see [Schedule expressions for rules](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/ScheduleExpressionsReference.html) in the *Amazon CloudWatch User Guide*.

**Editing a task schedule**

You can configure scheduling when you initially create a task (p. 28), 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**

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, with a minimum interval of 1 hour. For frequency configurations options, see [Configuring a task schedule](https://docs.aws.amazon.com/AmazonS3/latest/dg/objects-manage.html) (p. 66).

**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 task executions

In this section, you can find information about how to work with task executions in AWS DataSync.

A task execution is an individual run of a task, which includes information such as start time, end time, bytes written, and status.

After a task execution starts, you can monitor its progress, add or adjust bandwidth throttling for it, or cancel it before it completes.

Topics
- Adjusting bandwidth throttling for a task execution (p. 68)
- Canceling a task execution (p. 68)
- Monitoring a task execution using CloudWatch (p. 69)

Adjusting bandwidth throttling for a task execution

You can modify bandwidth throttling for a task execution using the AWS Management Console or the DataSync API. For information about using the API, see UpdateTaskExecution.

To modify bandwidth throttling
1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose Tasks.
3. Select the Task ID for the running task that you want to monitor. Using the console, you can modify any task execution that is currently running or that is queued. For more information about task execution status, see Understanding task execution statuses (p. 54).
4. Choose History to view task execution instances.
5. Select the active task execution to be modified.
6. For Action, choose Edit.
7. In the Edit Task Execution dialog box that appears, choose Use available to remove bandwidth throttling and use all available bandwidth for the task execution.

Choose Set bandwidth limit (MIB/s) to change the bandwidth limit.

To save changes to your task execution's bandwidth limit, choose Save changes. The new bandwidth limit setting goes into effect on the running or queued task execution within 60 seconds.

Canceling a task execution

Using the console, you can cancel any task execution that is currently running or that is queued. You can also cancel a task execution using the API. For more information, see CancelTaskExecution.

For information about task execution statuses, see Understanding task execution statuses (p. 54).
To cancel a task execution

1. Open the AWS DataSync console at https://console.aws.amazon.com/datasync/.
2. In the navigation pane, choose Tasks.
3. Select the Task ID for the running task that you want to monitor. The Status should be Running.
4. Choose History to view task execution instances. For information about task execution statuses, see Understanding task execution statuses (p. 54).
5. Select the active task execution to be stopped.
6. For Action, choose Stop.
7. In the Stop Task Execution dialog box that appears, choose Confirm to stop the task execution.

Monitoring a task execution using CloudWatch

Your task execution generates detailed Amazon CloudWatch metrics, logs, and events. To learn more about the monitoring information available through CloudWatch, see Monitoring your task (p. 50).
## 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 the following location types:

- Network File System (NFS)
- Server Message Block (SMB)
- Self-managed object storage
- Amazon EFS
- FSx for Windows File Server
- Amazon S3

When you create a task that transfers data between AWS services in different AWS Regions, one of the two locations you specify must reside in the Region where DataSync is being used. The other location must be specified in a different Region.

You can transfer data between AWS Regions except for China, or between AWS GovCloud (US-East and US-West) Regions.

**Note**

DataSync only copies your storage contents, and does not copy the configuration of the storage itself. For example, when copying objects between Amazon S3 buckets, DataSync does not copy any bucket-level settings or permissions. Likewise, when copying between SMB or NFS shares, DataSync does not copy any file system level settings or permissions.

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>Self-managed storage (including NFS shares, SMB shares, object storage, or NFS on your AWS Snowcone device)</td>
<td>Amazon S3 (in AWS Regions), Amazon EFS, or FSx for Windows File Server</td>
</tr>
<tr>
<td>Amazon S3 (in AWS Regions), Amazon EFS, or FSx for Windows File Server</td>
<td>Self-managed storage (including NFS shares, SMB shares, object storage, or NFS on your AWS Snowcone device)</td>
</tr>
<tr>
<td>Amazon S3 (in AWS Regions), Amazon EFS, or FSx for Windows File Server</td>
<td>Amazon S3 (in AWS Regions), Amazon EFS, or FSx for Windows File Server</td>
</tr>
<tr>
<td>Amazon S3 on AWS Outposts</td>
<td>Amazon S3 on AWS Outposts</td>
</tr>
<tr>
<td>Amazon S3 on AWS Outposts</td>
<td>Amazon S3 (in AWS Regions)</td>
</tr>
</tbody>
</table>
Creating a location for NFS

In addition, you can use the following combinations to transfer data between managed file systems and Amazon S3 buckets in different AWS accounts.

<table>
<thead>
<tr>
<th>Source (from)</th>
<th>Destination (to)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon EFS (configured as an NFS location) or FSx for Windows File Server (configured as an SMB location)</td>
<td>Amazon S3 (in AWS Regions), Amazon EFS, or FSx for Windows File Server</td>
</tr>
<tr>
<td>Amazon S3 (in AWS Regions)</td>
<td>Amazon EFS (configured as an NFS location) or FSx for Windows File Server (configured as an SMB location)</td>
</tr>
</tbody>
</table>

**Important**

When you use DataSync to copy files or objects between AWS Regions, you pay for data transfer between Regions. This is billed as data transfer OUT from your source Region to your destination Region. For more information, see Data transfer pricing.

**Topics**

- Creating a location for NFS (p. 71)
- Creating a location for SMB (p. 73)
- Creating a location for object storage (p. 75)
- Creating a location for Amazon EFS (p. 76)
- Creating a location for FSx for Windows File Server (p. 77)
- Creating a location for Amazon S3 (p. 79)
- How DataSync handles metadata and special files (p. 85)
- Deleting a location (p. 88)

**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 self-managed NFS server and makes it easier to securely transfer data between the self-managed 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. 255) 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. 72)**.

### 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 self-managed environment to connect to your self-managed location. An agent makes it easier to securely transfer data between the self-managed 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. 124)**.

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. 123)**.

### NFS server on AWS Snowcone

If you are copying data to or from your AWS Snowcone device, note the following configuration.

- **Agents** — Select the Amazon EC2 agent that you launched on your AWS Snowcone device. For more information about using DataSync with Snowcone, see **Using DataSync to transfer files to AWS in the AWS Snowcone User Guide**.

- **NFS server** — Specify the virtual IP address that you attached to the NFS server on your Snowcone device using AWS OpsHub for Snow Family or the AWS Snowball Edge CLI. For more information about
using AWS OpsHub, see Using AWS OpsHub for Snow Family for Snow Family to manage devices. For more information about using the AWS Snowcone client, see Using the Snowcone client.

- **Mount path** — Specify the NFS export path for the bucket you want to transfer data to or from. The format of the export path of an Amazon S3 bucket is /buckets/<bucket-name>. For more information about using the AWS Snowcone NFS server, see Using NFS file shares to manage file storage in the AWS Snowcone User Guide.

### 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 2.1 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.

When you copy data between SMB shares and Amazon FSx, both the source and the destination must belong to the same Active Directory domain, or have an Active Directory trust relationship between their domains. See .SMBuser to learn more about choosing a user that ensures sufficient permissions to files, folders, and 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 self-managed SMB server and makes it easier to securely transfer data between the self-managed 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 the file share. See .SMBuser to learn more about choosing a user that ensures sufficient permissions to files, folders, and metadata.
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 self-managed location. An agent makes it easier to securely transfer data between the self-managed 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 that can mount the location and has the permissions to access the SMB file share. This user can be a local user on your Windows file server, or it can be a domain user defined in your Active Directory.

To set object ownership, DataSync requires the SE_RESTORE_NAME privilege, which is usually granted to members of the built-in Active Directory groups Backup Operators and Domain Admins. Providing a user to DataSync with this privilege also helps ensure sufficient permissions to files, folders, and file metadata except for NTFS system access control lists (SACLs).

Additional privileges are required to copy SACLs. Specifically, this requires the Windows SE_SECURITY_NAME privilege, which is granted to members of the Domain Admins group. If your task will be configured to copy SACLs make sure that the user has the required privileges. To learn more about configuring a task to copy SACLs, see Ownership and permissions related options (p. 29).

When you copy data between SMB shares and FSx for Windows File Server locations, or between two FSx for Windows File Server locations, both the source and the destination must belong to the same Active Directory domain, or have an Active Directory trust relationship between their domains.

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 object storage

When you use a location in a task, you configure it as the source or destination location.

Your object storage system must be compatible with Amazon S3 for the following API calls.

- AbortMultipartUpload
- DeleteObject
- GetBucketLocation
- GetObject
- GetObjectTagging
- HeadBucket
- ListBucket
- ListBucketMultipartUploads
- ListMultipartUploadParts
- PutObject
- PutObjectTagging

Your object storage system must also support AWS Signature Version 4 for authenticating requests. AWS Signature Version 2 is deprecated, and DataSync does not support it.

To create a self-managed object storage 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 Object storage. You configure this location as a source or destination later.
5. For Agents, choose one or more agents that you want to use from the available agents. The agent connects to your self-managed object storage server and makes it easier to securely transfer data between the self-managed object storage location and AWS.
6. For Server, provide the domain name or IP address of the self-managed object storage server.
7. For Bucket name, enter the name of the self-managed object storage bucket that you will use for the data transfer.
8. For Folder, enter an object prefix that will be used for the data transfer. When your location is used as a source for a task, DataSync will only copy objects with the provided prefix. When your location is used as a destination for a task, all objects will be written under the provided prefix.
9. To select the object storage server protocol and server port, choose Additional settings. Select either HTTP or HTTPS.

By default, the Server port is set to 80 for HTTP, and 443 for HTTPS, but you can specify a custom port if your self-managed object storage server requires one.

10. (Optional) If credentials are required to access the self-managed object storage location, select Requires credentials, and enter the Access key and the Secret key that are needed to access the bucket. The access key and secret key parameters can also be used to provide user name and password, respectively.

11. (Optional) Tags are key-value pairs that help 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 EFS file system, see Getting started with Amazon Elastic File System in the Amazon Elastic File System User Guide.

The DataSync service mounts your file system from your virtual private cloud (VPC) using 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.

**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. 77)

**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 with more than 1 TiB of Standard class storage can drive 100 MiB/s per TB when burst credits are available.

  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 35,000 file system operations per second. This limit can impact the maximum throughput DataSync can achieve when copying files.

  Operations that read data or metadata consume one file operation, and operations that write data or update metadata consume five file operations. This means that a file system can support 35,000 read operations per second, or 7,000 write operations, or some combination of the two. File operations are counted from all connecting clients.

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

Creating a location for FSx for Windows File Server

A location for FSx for Windows File Server is an endpoint for an FSx for Windows File Server. DataSync accesses your 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.

DataSync authenticates against your FSx for Windows File Server file system using a user name and a password that you provide. When you copy data between SMB shares and Amazon FSx, or between two Amazon FSx locations, both the source and the destination must belong to the same Active Directory domain, or have an Active Directory trust relationship between their domains.

See .FSxWuser to learn more about choosing a user that ensures sufficient permissions to files, folders, and metadata.

The DataSync service mounts your file system from your virtual private cloud (VPC) using 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 FSx for Windows File Server in the current AWS Region, create one. For information about how to create an FSx for Windows File Server, see Getting started with Amazon FSx in the FSx for Windows File Server User Guide.

  **Note**
  DataSync currently doesn’t support transferring files to 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 Amazon 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 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 Amazon FSx file server. The path can include a subdirectory. If so, this is a subdirectory in the FSx for Windows File Server that is used to read data from the FSx location or write data to the Amazon 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 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 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 FSx for Windows File Server:

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

   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 FSx for Windows File Server, the user must be a member of the AWS Delegated FSx Administrators group. If you are using a self-managed Microsoft Active Directory with your 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.

   To set object ownership, DataSync requires the SE_RESTORE_NAME privilege, which is usually granted to members of the built-in Active Directory groups Backup Operators and Domain Admins. Providing a user to DataSync with this privilege also helps ensure sufficient permissions to files, folders, and file metadata except for NTFS system access control lists (SACLs).

   Additional privileges are required to copy SACLs. Specifically, this requires the Windows SE_SECURITY_NAME privilege, which is granted to members of the Domain Admins group. If your task will be configured to copy SACLs make sure that the user has the required privileges. To learn more about configuring a task to copy SACLs, see Ownership and permissions related options (p. 29).

   When you copy data between SMB shares and Amazon FSx, or between two Amazon FSx locations, both the source and the destination must belong to the same Active Directory domain, or have an Active Directory trust relationship between their domains.

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

   **Domain**
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. 80)
- Considerations when working with Amazon S3 storage classes in DataSync (p. 81)
- Manually configuring an IAM role to access your Amazon S3 bucket (p. 83)

To create an Amazon S3 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 Create location.
4. For Location type, choose Amazon S3 bucket.
5. For Amazon S3 Bucket, choose the Amazon S3 bucket that you want to use as an endpoint. You configure this location as a source or destination later.

If your Amazon S3 bucket is located on an AWS Outposts, specify an Amazon S3 access point. Choose the access point you want to use to transfer your objects. DataSync only supports access points for buckets located on AWS Outposts. For more information about Amazon S3 Access Points, see Managing data access with Amazon S3 access points in the Amazon S3 User Guide.

6. For Amazon 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 S3 User Guide. Some storage classes have behaviors that can affect your Amazon S3 cost. For detailed information, see Considerations when working with Amazon S3 storage classes in DataSync (p. 81).

For Amazon S3 on Outposts, DataSync automatically defaults to the Amazon S3 Outposts storage class.

7. For Agents: If your Amazon S3 bucket is located on an AWS Outposts, you must deploy an Amazon EC2 agent on your AWS Outposts. The agent must be in a VPC that is allowed to access the access point specified in step 5, and activated in your Outpost's parent Region. The agent must also be able to route to the Amazon S3 on Outposts endpoint for the bucket. For more information about launching a DataSync agent on AWS Outposts, see Deploy your DataSync agent on AWS Outposts (p. 23).

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

9. 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 Amazon 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 Amazon S3 bucket (p. 83).

10. (Optional) For **Key** and **Value**, provide values to tag your Amazon S3 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**. The location that you just created appears in the list of locations.

**Note**

- Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges in the following scenarios:
  - **When using object versioning**: Changes to object data or metadata create a new version of the object.
  - **When using storage classes that can incur additional charges for overwriting, deleting, or retrieving, objects**: Changes to object data or metadata will result in such charges. For more information, see Considerations when working with Amazon S3 storage classes in DataSync (p. 81).

- When you use versioning, 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 Amazon S3 buckets. Doing this can help you control your storage costs. For more information, see the blog post Amazon 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, configure the following settings.

### Amazon 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 S3 User Guide. Some storage classes have specific behaviors that can affect your Amazon S3 storage cost. For detailed information, see Considerations when working with Amazon S3 storage classes in DataSync (p. 81). For Amazon S3 on Outposts, DataSync automatically defaults to the Amazon S3 Outposts storage class.

### Folder

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

### IAM role

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

For DataSync to access a destination Amazon 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...
Considerations when working with Amazon 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>Amazon 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>
<tr>
<td>Amazon S3 Intelligent-Tiering</td>
<td>Choose Intelligent-Tiering to optimize storage costs by automatically moving data to the most cost-effective storage access tier.</td>
</tr>
<tr>
<td></td>
<td>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. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the Intelligent-Tiering storage class.</td>
</tr>
<tr>
<td></td>
<td>Objects less than 128 KB are not eligible for auto tiering in the Intelligent-Tiering storage class. These objects are stored in Standard.</td>
</tr>
<tr>
<td>Amazon S3 Standard-IA</td>
<td>Choose Standard-IA to store your infrequently accessed files redundantly in multiple Availability Zones that are geographically separated.</td>
</tr>
<tr>
<td>Amazon S3 storage class</td>
<td>Considerations</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td></td>
<td>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. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the Standard-IA storage class.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>Amazon S3 One Zone-IA</td>
<td>Choose Standard-IA to store your infrequently accessed files in a single Availability Zone.</td>
</tr>
<tr>
<td></td>
<td>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. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the One Zone-IA storage class.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>S3 Glacier</td>
<td>Choose S3 Glacier to archive the files for more active archives.</td>
</tr>
<tr>
<td></td>
<td>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. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 Glacier storage class.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>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>
</tbody>
</table>
Manually configuring an IAM role to access your Amazon 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 Amazon 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": [
      {
         "Effect": "Allow",
         "Principal": {
            "Service": "datasync.amazonaws.com"
         },
         "Action": "sts:AssumeRole"
      }
   ]
}
```

5. Choose Next: Permissions.

### Amazon S3 storage class | Considerations
---|---
S3 Glacier Deep Archive | 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. Changes to object data or metadata are equivalent to deleting an object and creating a new one to replace it. This results in additional charges for objects stored in the S3 Glacier Deep Archive storage class.

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 Verify only the data transferred to compare data and metadata checksums at the end of the transfer. Verify all data in the destination isn't an available option for this storage class, because it requires retrieving all existing objects from the destination.

Amazon S3 Outposts | The storage class for Amazon S3 on Outposts.
6. For Amazon S3 buckets in AWS Regions, choose **AmazonS3FullAccess**. You can also manually configure a more restrictive policy. For an example of such a policy, see the following.

```json
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "s3:GetBucketLocation",
            "s3:ListBucket",
            "s3:ListBucketMultipartUploads"
         ],
         "Effect": "Allow",
         "Resource": "YourS3BucketArn"
      },
      {
         "Action": [
            "s3:AbortMultipartUpload",
            "s3:DeleteObject",
            "s3:GetObject",
            "s3:ListMultipartUploadParts",
            "s3:GetObjectTagging",
            "s3:PutObjectTagging",
            "s3:PutObject"
         ],
         "Effect": "Allow",
         "Resource": "YourS3BucketArn/*"
      }
   ]
}
```

For Amazon S3 buckets on Outposts, use the following policy:

```json
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "s3-outposts:ListBucket",
            "s3-outposts:ListBucketMultipartUploads"
         ],
         "Effect": "Allow",
         "Resource": [
            "s3OutpostsBucketArn",
            "s3OutpostsAccessPointArn"
         ],
         "Condition": {
            "StringLike": {
               "s3-outposts:DataAccessPointArn": "s3OutpostsAccessPointArn"
            }
         }
      },
      {
         "Action": [
            "s3-outposts:AbortMultipartUpload",
            "s3-outposts:DeleteObject",
            "s3-outposts:GetObject",
            "s3-outposts:ListMultipartUploadParts",
            "s3:GetObjectTagging",
            "s3:PutObjectTagging"
         ],
         "Effect": "Allow",
         "Resource": "YourS3BucketArn/*"
      }
   ]
}
```
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.

**Note**
DataSync only copies your storage contents, and does not copy the configuration of the storage itself. For example, when copying objects between Amazon S3 buckets, DataSync does not copy any bucket-level settings or permissions. Likewise, when copying between SMB or NFS shares, DataSync does not copy any file-system level settings or permissions.

**Topics**
- Metadata copied by DataSync (p. 85)
- Default POSIX metadata applied by DataSync (p. 87)
- Links and directories copied by DataSync (p. 87)

**Metadata copied by DataSync**

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

**When copying between self-managed NFS or Amazon EFS, and Amazon EFS** – In this case, DataSync can copy the following metadata:

- File and folder modification timestamps
- User ID and group ID
- POSIX permissions
When copying between self-managed SMB or FSx for Windows File Server, and FSx for Windows File Server – In this case, DataSync can copy the following metadata:

- File timestamps: access time, modification time, and creation time
- File owner security identifier (SID)
- Standard file attributes:
  - Read-only (R)
  - Archive (A)
  - System (S)
  - Hidden (H)
  - Compressed (C)
  - Not content indexed (N)
  - Encrypted (E)
  - Temporary (T)
  - Offline (O)

  **Note**

  DataSync tries to copy the Archive and Compressed attributes. However, these attributes might not be applied on the destination, so they are ignored during the verification phase.

- NTFS discretionary access lists (DACLs) which determine whether to grant access to an object.
- NTFS system access control lists (SACLs) which are used by administrators to log attempts to access a secured object.

When copying between self-managed NFS or Amazon EFS, 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 NFS shares on file gateways in Storage Gateway. A file gateway enables low latency access from on-premises to data that was copied to Amazon S3 by DataSync. The metadata is also interoperable with FSx for Lustre.

When DataSync copies objects that contain this metadata back to an NFS server, the file metadata is restored. Restoring metadata requires granting elevated permissions to the NFS server when you are Creating a location for NFS (p. 71).

When copying between self-managed object storage and Amazon S3, or between two Amazon S3 buckets – In this case, DataSync copies object metadata and tags.

  **Note**

  DataSync does not copy other object information, such as object access control lists (ACLs) or prior object versions.

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</td>
</tr>
<tr>
<td>From FSx for Windows File Server to an NFS share</td>
<td></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 or self-managed object storage 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 location 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>0644</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 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. 52).

Symbolic links

Symbolic links are preserved when copying between an NFS server and Amazon EFS.
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 it is used to read or write data from a self-managed storage system. To create an agent to read from a self-managed source location, you download the AWS DataSync image and deploy it in your hypervisor and activate it. DataSync supports VMware ESXi, KVM, and Microsoft Hyper-V hypervisors.

Note
The default credentials for the DataSync image 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 Working with your agent on the local console (p. 98).

Topics
- Creating and activating an agent (p. 89)
- Using AWS DataSync in a virtual private cloud (p. 90)
- Using the DataSync EC2 agent deployed in AWS Regions (p. 93)
- Editing your agent's properties (p. 95)
- Using multiple agents for a location (p. 96)
- Understanding agent statuses (p. 96)
- Deleting an agent (p. 96)
- Configuring your agent for multiple NICs (p. 97)

Creating and activating an agent

After you deploy an agent into your hypervisor or Amazon EC2 environment, activate the agent. Activate your agent in the AWS Region where the Amazon S3 bucket, Amazon EFS, or 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 network traffic 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. 90).

- **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. 98). 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 DataSync endpoints (p. 101).

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 DataSync endpoints (p. 101).

For instructions on how to create an agent on a VMware ESXi host, see Deploy your DataSync agent on VMware (p. 19).

For instructions on how to create an agent on a KVM host, see Deploy your DataSync agent on KVM (p. 19).

For instructions on how to create an agent on a Microsoft Hyper-V host, see Deploy your DataSync agent on Hyper-V (p. 20).

For instructions on how to create an agent on an Amazon EC2 instance, see Deploy your agent as an Amazon EC2 instance to access in-cloud file systems (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 self-managed 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), or when accessing your self-managed object storage using the Amazon S3 API.

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.
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, NFS, or self-managed object 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, SMB, or the Amazon S3 API. 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 Standards (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. 19).

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 must 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.

   Important
   - If you are using a DataSync Amazon EC2 agent, choose the Availability Zone where your agent resides to avoid charges for network traffic between Availability Zones.
   - To learn more about data transfer prices for all AWS Regions, see Amazon EC2 On-Demand pricing.

   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 self-managed 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. 24).

   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.

**Using the DataSync EC2 agent deployed in AWS Regions**

This section describes data transfer between in-cloud file systems and Amazon S3 buckets using the DataSync EC2 agent.

**Note**
To transfer files or objects between datasets in Amazon S3, Amazon EFS, or Amazon FSx in the same AWS account, you don’t need to deploy a DataSync agent. To learn more, see Data transfer between AWS storage services (p. 4).

You can use the DataSync EC2 agent deployed in an AWS Region to transfer data between in-cloud file systems and Amazon S3 in different AWS accounts, or between self-managed in-cloud file systems and Amazon S3.

For more information about deploying a DataSync EC2 agent in a Region, see Deploy your agent as an Amazon EC2 instance to access in-cloud file systems (p. 21).

**Topics**
- Data transfer from in-cloud file system to in-cloud file system or Amazon S3 (p. 93)
- Data transfer from S3 to in-cloud file systems (p. 94)

**Data transfer from in-cloud file system to in-cloud file system or Amazon S3**

To transfer data from one AWS account to another, or from a self-managed in-cloud file system, the DataSync agent must be located in the same AWS Region and same AWS account where the source file system is deployed. This type of transfer includes the following:

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

**Important**
Deploy your agent such that it does not require network traffic between Availability Zones (to avoid charges for such traffic).

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

To learn more about data transfer prices for all AWS Regions, see Amazon EC2 On-Demand pricing.
For example, the following diagram shows a high-level view of the DataSync architecture for transferring data from in-cloud NFS to in-cloud NFS or Amazon S3.

**Note**

Deploy the agent in the AWS Region and AWS account where the source file system resides.

- When you’re copying between two Amazon EFS file systems in different AWS accounts, we recommend that you use the NFS (source) to EFS (destination) transfer.
- When you’re copying between two Amazon FSx file systems in different AWS accounts, we recommend that you use the SMB (source) to Amazon FSx (destination) transfer.

### Data transfer from S3 to in-cloud file systems

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

For most workloads, we recommend that you activate one agent for each self-managed 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 self-managed 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 self-managed 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.

<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 and has been out of contact with the service for 5 minutes or longer. 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. In the left navigation pane, choose Agents.
2. Choose the agent you want to delete.
3. Choose Delete, enter delete in the text box that appears, and then choose Delete.

Deleting an agent cannot be undone, but you can re-use the VM or Amazon EC2 instance. This can simplify the process of deploying the VM or instance. You can configure and initialize the VM or EC2 instance once, and then re-use it when needed with a new agent activation on that VM or EC2 instance.

**To create and activate an agent on a VM or EC2 instance after deleting an agent**

1. Delete the old agent (see the preceding steps for instructions). Do not delete the VM or EC2 instance.
2. Wait until the old agent is deleted and the VM is ready to be activated, usually about 3 minutes. Alternatively, you can verify that the agent has been deleted by checking the status of port 80. When the VM is ready to be activated, port 80 will be open.
3. Create and activate a new DataSync agent on the existing VM or Amazon EC2 instance. For information about creating a DataSync agent, see Creating and activating an agent (p. 89). The new agent can be activated in a different AWS Region, depending on network connectivity.

**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, SMB, or object storage server might reside on a virtual LAN (VLAN) that lacks internet connectivity for security reasons.

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

In some cases, you might configure one adapter to connect to the 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.
Working with your agent on the local console

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

Topics
• Logging in to the AWS DataSync local console (p. 98)
• Obtaining an activation key using the local console (p. 98)
• Configuring your agent network settings (p. 99)
• Testing your agent connection to DataSync endpoints (p. 101)
• Testing connectivity to self-managed storage (p. 101)
• Viewing your agent system resource status (p. 102)
• Configuring a Network Time Protocol (NTP) server for VMware agents (p. 103)
• Running AWS DataSync commands on the local console (p. 103)
• Enabling AWS Support to help troubleshoot your running agent (p. 104)

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 connect to the agent using SSH (Secure Shell) 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 5 on the main menu). For information about how to run the command, see Running AWS DataSync commands on the local console (p. 103).

Obtaining an activation key using the local console

If your agent has not already been activated, you can obtain an activation key for your agent from the local console. This option is displayed only until the agent has been activated.

To get an activation key for your agent from the local console

1. Log in to your agent's local console.
2. On the AWS DataSync Activation - Configuration main menu, enter 0 to get an activation key.
3. Enter the AWS Region that your agent will be activated in.
4. Enter the service endpoint type that your agent will be using. Options include public, FIPS, and VPC with AWS PrivateLink.
5. The activation key is automatically generated and displayed on screen. Select and copy this value.
6. Using the activation key copied from the last step, use the following CLI command to create and activate the agent:

   ```
   # aws datasync create-agent --agent-name your-new-agent-name --activation-key generated-activation-key
   ```

   On successful activation, this command returns something similar to the following.

   ```
   {
   }
   ```

   You can also insert the activation key in the DataSync console using the agent creation wizard.

   After the agent is activated, the console menu displays the Agent ID and AWS Region. The option for getting an activation key is no longer visible in the console menu.

---

### 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 DataSync Activation - Configuration** main menu, enter 1 to begin configuring your network.
3. On the **Network Configuration** menu, choose one of the following options.

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get information about your network adapter</td>
<td>Enter 1. A list of adapter names appears, and you are prompted to enter an adapter name—for example, <code>eth0</code>. If the adapter you specify is in use, the following information about the adapter is displayed:</td>
</tr>
<tr>
<td></td>
<td>Media access control (MAC) address</td>
</tr>
<tr>
<td></td>
<td>IP address</td>
</tr>
<tr>
<td></td>
<td>Netmask</td>
</tr>
<tr>
<td></td>
<td>Agent IP address</td>
</tr>
<tr>
<td></td>
<td>DHCP enabled status</td>
</tr>
<tr>
<td></td>
<td>You use the same adapter name when you configure a static IP address (option 3) as when</td>
</tr>
</tbody>
</table>

---

99
<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>you set your agent's default route adapter (option 5).</td>
<td></td>
</tr>
</tbody>
</table>
| 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. |
| Reset all your agent's network configuration to DHCP | Enter 4.  
All network interfaces are set to use DHCP.  
**Important**  
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. |
| Set your agent's default route adapter | Enter 5.  
The available adapters for your agent are shown, and you are prompted to choose one of the adapters—for example, eth0. |
| Edit your agent's DNS configuration | Enter 6.  
The available adapters of the primary and secondary DNS servers are displayed. You are prompted to provide the new IP address. |
| View your agent's DNS configuration | Enter 7.  
The available adapters of the primary and secondary DNS servers are displayed.  
**Note**  
For some versions of the VMware hypervisor, you can edit the adapter configuration in this menu. |
| View routing tables | Enter 8.  
The default route of your agent is displayed. |
Testing your agent connection to DataSync endpoints

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 DataSync endpoints**

1. Log in to your agent's local console.
2. On the **AWS DataSync Activation - Configuration** main menu, enter 2 to begin testing network connectivity.
3. Enter the service endpoint type that your agent is connecting to. Valid endpoint types include public, FIPS, and VPC endpoints using AWS PrivateLink.

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

a. To test public endpoint connectivity, enter 1, followed by the AWS Region in which your agent is activated. Connectivity test results against the correct endpoints for your agent's Region are displayed. 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.

b. To test FIPS endpoint connectivity, enter 2, followed by the AWS Region in which your agent is activated. Connectivity test results against the correct endpoints for your agent's Region are displayed. 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.

c. To test VPC connectivity, enter 3. Network connectivity test results for your agent's VPC endpoints are displayed.

Each VPC endpoint displays either a **PASSED** or **FAILED** message.

For information about network and firewall requirements, see [Network requirements for DataSync](p. 10).

Testing connectivity to self-managed storage

You can use the console to test connectivity to your self-managed storage, including NFS, SMB, or object storage servers.

**To test connectivity to self-managed storage servers**

1. Log in to your agent's local console.
2. On the **AWS DataSync Activation - Configuration** main menu, enter 3 to begin testing network connectivity of self-managed storage.
3. Choose the location type for connectivity testing. Options include the following.
   a. Enter 1 to test connectivity to an NFS server.
b. Enter 2 to test connectivity to an SMB server.
c. Enter 3 to test connectivity to an object storage server.

Enter the IP address or server domain name of the NFS server.

Connectivity test results, either PASSED or FAILED, are displayed for the specified server, along with the IP address and port of the tested server.

### Viewing your agent system resource status

When you log in to your agent console, virtual CPU cores, root volume size, and RAM are automatically checked. If there are any errors or warnings, they are flagged on the console menu display with a banner that provides details about those errors or warnings.

If there are no errors or warnings when the console starts, the menu displays white text. The View System Resource Check option will display (0 Errors).

If there are errors or warnings, the console menu displays the number of errors and warnings, in red and yellow respectively, in a banner across the top of the menu. For example, (1 ERROR, 1 WARNING).

**To view the status of a system resource check**

1. Log in to your agent’s local console.
2. On the AWS DataSync Activation - Configuration main menu, enter 4 to view the results of the system resource check.

The console displays an [OK], [WARNING], or [FAIL] message for each resource as described in the table following.

For Amazon EC2 instances, the system resource check verifies that the instance type is one of the instances recommended for use with DataSync. If the instance type matches that list, a single result is displayed in green text, as follows.

[ OK ] Instance Type Check

If the Amazon EC2 instance is not on the recommended list, the system resource check verifies the following resources:

- CPU cores check: At least four cores are required.
- Disk size check: A minimum of 80 GB available disk space is required.
- RAM check: A minimum of 32 GiB of RAM is required for up to 20 million file transfers per task. A minimum of 64 GiB of RAM is required for more than 20 million file transfers per task.
- CPU flags check: The agent VM CPU must have either SSSE3 or SSE4 instruction set flags.

If the Amazon EC2 instance is not on the list of recommended instances for DataSync, but it has sufficient resources, the result of the system resource check displays four results, all in green text.

The same resources are verified for agents deployed in Hyper-V, KVM, and VMware VMs.

VMware agents are also checked for supported version; unsupported versions trigger a red banner error. Supported versions include VMware version 6.5 and 6.7.
Configuring a Network Time Protocol (NTP) server for VMware agents

If you are using a VMware VM, you can view Network Time Protocol (NTP) server configurations and synchronize the VM time on your agent with your VMware hypervisor host.

**To manage system time**

1. Log in to your agent's local console.
2. On the AWS DataSync Activation - Configuration main menu, enter 5 to manage your system's time.
3. On the System Time Management menu, enter 1 to view and synchronize the VM system time.

<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. 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. 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 times when operations such as snapshots occur and the actual times that the operations occur.</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>

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 DataSync Activation - Configuration main menu, enter 5 for Command Prompt.

   Note
   If you are using a VMware VM, enter 6 for the Command Prompt.

3. The commands available to be used through the console include the following.

<table>
<thead>
<tr>
<th>Use this command</th>
<th>To do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip</td>
<td>Display or configure routing, devices, and tunnels</td>
</tr>
<tr>
<td>save-routing-table</td>
<td>Save newly added routing table entry</td>
</tr>
<tr>
<td>ifconfig</td>
<td>Display or configure network interfaces</td>
</tr>
<tr>
<td>iptables</td>
<td>Administer IPv4 packet filtering and network address translation (NAT)</td>
</tr>
<tr>
<td>save-iptables</td>
<td>Persist IP tables</td>
</tr>
<tr>
<td>dig</td>
<td>Perform DNS lookup for DNS hostname</td>
</tr>
<tr>
<td>open-support-channel</td>
<td>Connect to AWS Support</td>
</tr>
<tr>
<td>h</td>
<td>Display available command list</td>
</tr>
<tr>
<td>exit</td>
<td>Return to console configuration menu</td>
</tr>
</tbody>
</table>

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

Enabling AWS Support to help troubleshoot your running agent

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

To enable AWS Support access to AWS DataSync

1. Log in to your host'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. (Item 5 on the main menu opens the command prompt. For VMware VMs, choose item 6.) For information about how to run the command, see Running AWS DataSync commands on the local console (p. 103).

2. At the prompt, enter 5 to open the command prompt (for VMware VMs, use 6).
3. Enter h to open the AVAILABLE COMMANDS window.
4. In the AVAILABLE COMMANDS window, enter the following to connect to AWS Support:

   open-support-channel

   If you are using the agent with VPC endpoints, you must provide a VPC endpoint IP address for your support channel, as follows:
enabling aws support to help troubleshoot datasync

open-support-channel VPCe IP address

Your firewall must allow the outbound TCP port 22 to initiate a support channel to AWS. When you connect to AWS Support, DataSync assigns you a support number. Make a note of your support number.

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. When the support channel is established, provide your support service number to AWS Support so that they can provide troubleshooting assistance.
6. When the support session is completed, press Enter to end it.
7. Enter exit to log out of the DataSync local console.
8. Follow the prompts to exit the local console.
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. 106)
- Identity and access management in AWS DataSync (p. 107)
- Logging AWS DataSync API calls with AWS CloudTrail (p. 119)
- Compliance validation for AWS DataSync (p. 121)
- Resilience in AWS DataSync (p. 122)
- Infrastructure security in AWS DataSync (p. 122)

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) 1.2 to encrypt all network traffic 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 S3 User 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. 115).

Topics

- Using identity-based policies (IAM policies) for DataSync (p. 107)
- DataSync API permissions: Actions, resources (p. 111)
- Overview of managing access permissions for DataSync (p. 115)

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. 118)
- Permissions (p. 119)

We recommend that you read 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. 115).

The sections in this topic cover the following:

- AWS managed policies for DataSync (p. 108)
- Permissions required to use the DataSync console (p. 108)
- Customer managed policy examples (p. 110)

The following shows an example of a permissions policy.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowsSpecifiedActionsOnAllTasks",
```
"Effect": "Allow",
"Action": [
    "datasync:DescribeTask",
    "datasync:ListTasks"
],
}

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 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:

The following AWS managed policies, which you can attach to users in your account, are specific 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. 108).

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. 110)
- Example 2: Allow DataSync to read and write to your Amazon S3 bucket (p. 110)
- Example 3: Allow DataSync to upload logs to CloudWatch log groups (p. 111)

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 the following.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads"
      ],
      "Effect": "Allow",
      "Resource": "YourS3BucketArn"
    },
    {
      "Action": [
        "s3:AbortMultipartUpload",
```

110
"s3:DeleteObject",
"s3:GetObject",
"s3:ListMultipartUploadParts",
"s3:GetObjectTagging",
"s3:PutObjectTagging",
"s3:PutObject"
],
"Effect": "Allow",
"Resource": "YourS3BucketArn/*"
}
]
}

Example 3: Allow DataSync to upload logs to CloudWatch log groups

DataSync requires permissions to be able to upload logs to your Amazon CloudWatch log groups. You can use CloudWatch log groups to monitor and debug your tasks.

For an example of an IAM policy that grants such permissions, see Allowing DataSync to upload logs to Amazon CloudWatch log groups (p. 52).

DataSync API permissions: Actions, resources

When you are setting up Permissions (p. 119) 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. 115).

DataSync API operations and required permissions for actions

CancelTaskExecution

Actions: datasync:CancelTaskExecution


CreateAgent

Actions: datasync:CreateAgent

Resource: None

CreateLocationEfs

Actions: datasync:CreateLocationEfs


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

CreateLocationFSxWindows

Actions: datasync:CreateLocationFSxWindows


and

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

CreateLocationNfs

Actions: datasync:CreateLocationNfs


CreateLocationS3

Actions: datasync:CreateLocationS3

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

CreateLocationSmb

Actions: datasync:CreateLocationSmb


CreateTask

Actions: datasync:CreateTask


DeleteAgent

Actions: datasync:DeleteAgent


DeleteLocation

Actions: datasync:DeleteLocation


DeleteTask

Actions: datasync:DeleteTask


DescribeAgent

Actions: datasync:DescribeAgent


DescribeLocationEfs

Actions: datasync:DescribeLocationEfs

DescribeLocationNfs

**Actions:** datasync:DescribeLocationNfs

**Resource:** arn:aws:datasync:*:account-id:location/location-id

DescribeLocationS3

**Actions:** datasync:DescribeLocationS3

**Resource:** arn:aws:datasync:*:account-id:location/location-id

DescribeLocationSmb

**Actions:** datasync:DescribeLocationSmb

**Resource:** arn:aws:datasync:*:account-id:location/location-id

DescribeTask

**Actions:** datasync:DescribeTask

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

DescribeTaskExecution

**Actions:** datasync:DescribeTaskExecution

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

ListAgents

**Actions:** datasync:ListAgents

**Resource:** None

ListLocations

**Actions:** datasync:ListLocations

**Resource:** None

ListTagsForResource

**Actions:** datasync:ListTagsForResource

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

or

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

or

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

or

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

ListTaskExecutions

**Actions:** datasync:ListTaskExecutions

**Resource:** None
ListTasks

Actions: datasync:ListTasks

Resource: None

StartTaskExecution

Actions: datasync:StartTaskExecution


TagResource

Actions: datasync:TagResource


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

Actions: datasync:UntagResource


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

Actions: datasync:UpdateAgent


UpdateTask

Actions: datasync:UpdateTask


Related topics

- Permissions (p. 119)
- Customer managed policy examples (p. 110)
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. 115)
- Understanding resource ownership (p. 115)
- Managing access to resources (p. 116)
- Specifying policy elements: Actions, effects, resources, and principals (p. 117)
- Specifying conditions in a policy (p. 118)
- Controlling access (p. 118)

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. 111).

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 Identity and Access Management policy reference* in the *IAM User Guide*.

Policies attached to an IAM identity are referred to as *identity-based* policies (IAM policies) and policies attached to a resource are referred to as *resource-based* policies. DataSync supports only identity-based policies (IAM policies).

**Topics**

• [Identity-based policies (IAM policies)](p. 116)
• [Resource-based policies](p. 117)

**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.
Overview of managing access

For more information about using identity-based policies with DataSync, see Using identity-based policies (IAM policies) for DataSync (p. 107). 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. 111)), 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 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. 115).

- **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 Identity and Access Management policy reference in the IAM User Guide.

For a table showing all of the DataSync API actions, see DataSync API permissions: Actions, resources (p. 111).
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*.
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...
Understanding AWS DataSync log file entries

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.

For more information, see CloudTrail userIdentity element in the AWS CloudTrail User Guide.

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": "AIDAJOERGY7LS5PKXTM KO",
        "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",
```
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. 123)
- You get a "Failed to retrieve agent activation key" error (p. 124)
- You can't activate an agent created using a VPC endpoint (p. 124)
- Your task status is unavailable and status indicates a mount error (p. 124)
- Your task execution fails with an Input/Output error message (p. 125)
- Your task execution is stuck in launching status (p. 125)
- Your task execution fails with a permissions denied error message (p. 125)
- Preparing status for a task execution takes longer than expected to complete (p. 126)
- Verifying status for a task execution takes longer than expected to complete (p. 126)
- Your storage cost is higher than expected (p. 126)
- How to enable AWS Support to troubleshoot your running on-premises agent (p. 127)

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. 150) API operation.
- For SMB, use the optional `Version` parameter for the CreateLocationSmb (p. 163) 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.

```
$ 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.

```
$ 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 elastic network interface. For information about Support Channel, see Enabling AWS Support to help troubleshoot your running agent (p. 104).

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 DataSync endpoints (p. 101).

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 agent (p. 104).

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. 61).

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 an 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.
- 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.

How to enable AWS Support to 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 agent (p. 104).
Troubleshooting an EC2 agent

Following, you can find information on how to troubleshoot EC2 agent issues.

Enabling AWS Support to 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-key](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html#retrieving-the-public-key) in 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 Working with your agent on the local console (p. 98).
AWS DataSync quotas and limits

Following, you can find information on AWS DataSync resources and their quotas and limits.

Topics
- Quotas for tasks (p. 129)
- Quotas for task executions (p. 131)
- Limits for DataSync file systems (p. 131)
- Limits for DataSync filters (p. 131)

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>
<th>Can quota be increased?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of tasks you can create in account per AWS Region</td>
<td>100</td>
<td>Yes</td>
</tr>
<tr>
<td>Maximum number of files per task, when transferring data between self-managed storage and AWS services</td>
<td>50 million</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Important**
For tasks that transfer more than 20 million files, make sure that you allocate a minimum of 64 GB of RAM to the virtual machine (VM). For minimum resource requirements for DataSync, see Virtual machine requirements (p. 10).

**Note**
As an alternative to requesting an increase, you can create tasks on different subdirectories using include/exclude filters. For more information about using filters, see Filtering the data transferred.
### Quotas for tasks

<table>
<thead>
<tr>
<th>Resource</th>
<th>Quota</th>
<th>Can quota be increased?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of files per task, when transferring data between AWS storage services</td>
<td>25 million</td>
<td>Yes</td>
</tr>
<tr>
<td>Maximum number of files per task, when running DataSync on an AWS Snowcone device</td>
<td>200,000</td>
<td>No</td>
</tr>
<tr>
<td>Maximum throughput per task</td>
<td>10 Gbps</td>
<td>No</td>
</tr>
</tbody>
</table>

You can take the following steps to request an increase for the permitted 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 quota increase**

1. Open the [AWS Support Center](https://aws.amazon.com/support/) page, sign in if necessary, and then choose **Create case**.
2. For **Create case**, choose **Service limit increase**.
3. For **Limit type**, choose **DataSync**.
4. For **Region**, select your AWS Region, and for **Limit**, select the quota that you want to increase.
5. Fill in the case description, and then choose your preferred method of contact.
If you need to increase a different quota, fill out a separate request.

### Quotas for task executions

Following are the quotas on task executions for each customer account in an AWS Region.

<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>

### Limits for DataSync file systems

The following table lists file system limits for DataSync.

If the storage systems at the source and destination locations have higher limits on the lengths of the total path and path components (file names, directories, and subdirectories), DataSync might not be able to access some objects on those systems.

<table>
<thead>
<tr>
<th>Description</th>
<th>Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum total file path length</td>
<td>4096 bytes</td>
</tr>
<tr>
<td>Maximum file path component (file name, directory, or subdirectory) length</td>
<td>255 bytes</td>
</tr>
<tr>
<td>Maximum length of Windows domain</td>
<td>253 characters</td>
</tr>
<tr>
<td>Maximum length of server hostname</td>
<td>255 characters</td>
</tr>
<tr>
<td>Maximum Amazon S3 object name length</td>
<td>1024 UTF-8 characters</td>
</tr>
</tbody>
</table>

### Limits for DataSync filters

Following are the limits on DataSync filters per task or task execution.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of characters in a filter string</td>
<td>409,600</td>
</tr>
</tbody>
</table>
Additional resources for AWS DataSync

In this section, you can find additional information about and resources for AWS DataSync.

Topics
- Transferring data from a self-managed storage array (p. 132)
- Additional AWS DataSync use cases (p. 132)

Transferring data from a self-managed storage array

You might want to transfer data from a self-managed 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
- NetApp: Snapshot management
- HPE 3PAR: Snapshots and copy data management
- HDS: Hitachi Copy-on-Write Snapshot User Guide

Additional AWS DataSync use cases

In this section, you can find information about use cases in AWS DataSync that are not common to most users.

Topics
- Transferring files in opposite directions (p. 132)
- Using multiple tasks to write to the same Amazon S3 bucket (p. 133)
- Allowing Amazon S3 access from a private VPC endpoint (p. 133)

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 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 Controlling access from VPC endpoints with bucket policies in the Amazon S3 User Guide.
API Reference

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. 135)
- Data Types (p. 247)
- Common Errors (p. 272)
- Common Parameters (p. 274)

Actions

The following actions are supported:
- CancelTaskExecution (p. 137)
- CreateAgent (p. 139)
- CreateLocationEfs (p. 143)
- CreateLocationFsxWindows (p. 147)
- CreateLocationNfs (p. 150)
- CreateLocationObjectStorage (p. 154)
- CreateLocationS3 (p. 158)
- CreateLocationSmb (p. 163)
- CreateTask (p. 167)
- DeleteAgent (p. 172)
- DeleteLocation (p. 174)
- DeleteTask (p. 176)
- DescribeAgent (p. 178)
- DescribeLocationEfs (p. 181)
- DescribeLocationFsxWindows (p. 184)
- DescribeLocationNfs (p. 187)
- DescribeLocationObjectStorage (p. 190)
- DescribeLocationS3 (p. 193)
- DescribeLocationSmb (p. 196)
- DescribeTask (p. 199)
- DescribeTaskExecution (p. 205)
- ListAgents (p. 210)
- ListLocations (p. 212)
- ListTagsForResource (p. 215)
- ListTaskExecutions (p. 218)
- ListTasks (p. 221)
- StartTaskExecution (p. 224)
- TagResource (p. 228)
• UntagResource (p. 230)
• UpdateAgent (p. 232)
• UpdateLocationNfs (p. 234)
• UpdateLocationObjectStorage (p. 237)
• UpdateLocationSmb (p. 240)
• UpdateTask (p. 243)
• UpdateTaskExecution (p. 246)
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. 274).

The request accepts the following data in JSON format.

TaskExecutionArn (p. 137)

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. 272).

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 V2
- 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 Creating and 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

```
{
  "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. 274).

The request accepts the following data in JSON format.

**ActivationKey** (p. 139)

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 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 Creating and 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. 139)

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. 139)

The ARNs of the security groups used to protect your data transfer task subnets. See SecurityGroupArns.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Maximum length of 128.


Required: No

**SubnetArns**  (p. 139)

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. 139)

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. 266) objects
CreateAgent

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

VpcEndpointId (p. 139)

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

{  "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. 141)

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. 272).

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-2:1112233444: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 V2
- 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

```
{
    "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 (p. 274).

The request accepts the following data in JSON format.

Ec2Config (p. 143)

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. 250) object

Required: Yes

EfsFilesystemArn (p. 143)

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-.*$
Required: Yes

**Subdirectory (p. 143)**

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\-\._\+/\(\)]*$`

Required: No

**Tags (p. 143)**

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. 266)` objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

**Response Syntax**

```
{
"LocationArn": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LocationArn (p. 144)**

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. 272).
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:datasync:us-east-2:11122233344: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 V2
- 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 Server 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. 274).

The request accepts the following data in JSON format.

**Domain (p. 147)**

The name of the Windows domain that the FSx for Windows File 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

**FsxFilesystemArn (p. 147)**

The Amazon Resource Name (ARN) for the FSx for Windows File Server file system.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):fsx:[a-z]{12}:file-system/fs-.*$*

Required: Yes

**Password (p. 147)**

The password of the user who has the permissions to access files and folders in the FSx for Windows File Server file system.

Type: String
Length Constraints: Maximum length of 104.

Pattern: ^.{0,104}$

Required: Yes

**SecurityGroupArns (p. 147)**

The Amazon Resource Names (ARNs) of the security groups that are to use to configure the FSx for Windows File Server 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. 147)**

A subdirectory in the location's path. This subdirectory in the Amazon FSx for Windows File Server file system is used to read data from the Amazon FSx for Windows File Server source location or write data to the FSx for Windows File Server destination.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[a-zA-Z0-9\-_+/\(\)]\p{Zs}]+$

Required: No

**Tags (p. 147)**

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. 266) objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

**User (p. 147)**

The user who has the permissions to access files and folders in the FSx for Windows File Server file system.

For information about choosing a user name that ensures sufficient permissions to files, folders, and metadata, see user.

Type: String

Length Constraints: Maximum length of 104.

Pattern: ^[^\x5B\x5D\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. 149)**

The Amazon Resource Name (ARN) of the FSx for Windows File Server 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. 272).

**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 V2
- 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. 274).

The request accepts the following data in JSON format.

MountOptions  (p. 150)

The NFS mount options that DataSync can use to mount your NFS share.

Type:  NfsMountOptions  (p. 255) object

Required: No

OnPremConfig  (p. 150)

Contains a list of Amazon Resource Names (ARNs) of agents that are used to connect to an NFS server.

If you are copying data to or from your AWS Snowcone device, see NFS Server on AWS Snowcone for more information.

Type:  OnPremConfig  (p. 256) object

Required: Yes

ServerHostname  (p. 150)

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.

If you are copying data to or from your AWS Snowcone device, see NFS Server on AWS Snowcone for more information.

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])$

Required: Yes

Subdirectory (p. 150)

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.

If you are copying data to or from your AWS Snowcone device, see NFS Server on AWS Snowcone for more information.

For information about NFS export configuration, see 18.7. The /etc/exports Configuration File in the Red Hat Enterprise Linux documentation.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[^a-zA-Z0-9_\-+./\(\)\p{Zs}]+$

Required: Yes

Tags (p. 150)

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. 266) objects

Array Members: Minimum number of 0 items. Maximum number of 50 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. 151)**

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. 272).

**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**

```
{
  "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

```
{
  "LocationArn": "arn:aws:datsync:us-east-2:11122233344: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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateLocationObjectStorage

Creates an endpoint for a self-managed object storage bucket. For more information about self-managed object storage locations, see Creating a location for object storage.

Request Syntax

```json
{
    "AccessKey": "string",
    "AgentArns": [ "string" ],
    "BucketName": "string",
    "SecretKey": "string",
    "ServerHostname": "string",
    "ServerPort": number,
    "ServerProtocol": "string",
    "Subdirectory": "string",
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }
    ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

AccessKey (p. 154)

Optional. The access key is used if credentials are required to access the self-managed object storage server. If your object storage requires a user name and password to authenticate, use AccessKey and SecretKey to provide the user name and password, respectively.

Type: String


Pattern: ^.+$

Required: No

AgentArns (p. 154)

The Amazon Resource Name (ARN) of the agents associated with the self-managed object storage server location.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: Yes
BucketName (p. 154)

The bucket on the self-managed object storage server that is used to read data from.

Type: String


Pattern: ^[a-zA-Z0-9-\_\-\+\./\(\)]\p{Zs}]+$

Required: Yes

SecretKey (p. 154)

Optional. The secret key is used if credentials are required to access the self-managed object storage server. If your object storage requires a user name and password to authenticate, use AccessKey and SecretKey to provide the user name and password, respectively.

Type: String


Pattern: ^.+$

Required: No

ServerHostname (p. 154)

The name of the self-managed object storage server. This value is the IP address or Domain Name Service (DNS) name of the object storage server. An agent uses this host name to mount the object storage server in a network.

Type: String

Length Constraints: Maximum length of 255.

Pattern: ^(([a-zA-Z0-9\-]*[a-zA-Z0-9])\.)*([A-Za-z0-9-\-]*[A-Za-z0-9])$

Required: Yes

ServerPort (p. 154)

The port that your self-managed object storage server accepts inbound network traffic on. The server port is set by default to TCP 80 (HTTP) or TCP 443 (HTTPS). You can specify a custom port if your self-managed object storage server requires one.

Type: Integer


Required: No

ServerProtocol (p. 154)

The protocol that the object storage server uses to communicate. Valid values are HTTP or HTTPS.

Type: String

Valid Values: HTTPS | HTTP

Required: No

Subdirectory (p. 154)

The subdirectory in the self-managed object storage server that is used to read data from.
Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[a-zA-Z0-9_-\-\_\\./\(+\\)]{0,2}]*$

Required: No

**Tags (p. 154)**

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. 266)** objects

Array Members: Minimum number of 0 items. Maximum number of 50 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. 156)**

The Amazon Resource Name (ARN) of the agents associated with the self-managed object storage server location.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:location/loc-[0-9a-z]{17}$

### Errors

For information about the errors that are common to all actions, see **Common Errors (p. 272)**.

**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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateLocationS3

Creates an endpoint for an Amazon S3 bucket.

For more information, see Create an Amazon S3 location in the AWS DataSync User Guide.

Request Syntax

```
{
    "AgentArns": [ "string" ],
    "S3BucketArn": "string",
    "S3Config": {
        "BucketAccessRoleArn": "string"
    },
    "S3StorageClass": "string",
    "Subdirectory": "string",
    "Tags": [ {
        "Key": "string",
        "Value": "string"
    } ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

AgentArns (p. 158)

If you are using DataSync on an AWS Outpost, specify the Amazon Resource Names (ARNs) of the DataSync agents deployed on your Outpost. For more information about launching a DataSync agent on an AWS Outpost, see Deploy your DataSync agent on AWS Outposts.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: No

S3BucketArn (p. 158)

The ARN of the Amazon S3 bucket. If the bucket is on an AWS Outpost, this must be an access point ARN.

Type: String

Length Constraints: Maximum length of 156.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):(s3|s3-outposts):[a-z\-0-9]*:[0-9]*:.*$
CreateLocationS3

Required: Yes

**S3Config (p. 158)**

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. 264) object

Required: Yes

**S3StorageClass (p. 158)**

The Amazon S3 storage class that you want to store your files in when this location is used as a task destination. For buckets in AWS Regions, the storage class defaults to Standard. For buckets on AWS Outposts, the storage class defaults to AWS S3 Outposts.

For more information about S3 storage classes, see Amazon S3 Storage Classes. 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.

Type: String

Valid Values: STANDARD | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

Required: No

**Subdirectory (p. 158)**

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. 158)**

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. 266) objects

Array Members: Minimum number of 0 items. Maximum number of 50 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. 159)

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. 272).

InternalException

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

InvalidRequestException

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

Examples

Step 1. Allow to assume the IAM role required to write to the bucket

The following example shows the simplest policy that grants the required permissions for AWS DataSync to access a destination Amazon S3 bucket, followed by an IAM role to which the create-location-s3-iam-role policy has been attached.

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

```
"Role": {
  "Path": "/",
  "RoleName": "MyBucketAccessRole",
```

160
Step 2. Allow the created IAM role to write to the bucket

Attach a policy that has sufficient permissions to access the bucket to the role. An example of such policy is the AWSDataSyncFullAccess managed policy.

For more information, see AWSDataSyncFullAccess in the IAM console.

You don't need to create this policy. It's managed by AWS, so all that you need to do is specify its ARN in the attach-role-policy command.

IAM_POLICY_ARN='arn:aws:iam::aws:policy/AWSDataSyncFullAccess'

Step 3. Create an endpoint for an Amazon S3 bucket

The following example creates an endpoint for an Amazon S3 bucket.

When the S3 endpoint is created, a response similar to the second example following returns the Amazon Resource Name (ARN) for the new Amazon S3 location.

Sample Request

```json
{
    "S3BucketArn": "arn:aws:s3:::MyBucket",
    "S3Config": {
        "BucketAccessRoleArn": "arn:aws:iam::111222333444:role/MyBucketAccessRole",
    },
    "S3StorageClass": "STANDARD",
    "Subdirectory": "/MyFolder",
    "Tags": [
        { "Key": "Name", "Value": "s3Bucket-1"
    ]
}
}
```

Sample Response

```json
{
}
```
"LocationArn": "arn:aws:datasync:us-east-2:11122333444: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 V2
- 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

```
{
  "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. 274).

The request accepts the following data in JSON format.

**AgentArns** (p. 163)

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 4 items.
- Length Constraints: Maximum length of 128.
- Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$`
- Required: Yes

**Domain** (p. 163)

The name of the Windows domain that the SMB server belongs to.

- Type: String
- Pattern: `^[A-Za-z0-9]+[A-Za-z0-9-\.]*[A-Za-z0-9-]*[A-Za-z0-9]$`
- Required: No

**MountOptions** (p. 163)

The mount options used by DataSync to access the SMB server.
CreateLocationSmb

Type: SmbMountOptions (p. 265) object

Required: No

Password (p. 163)

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. 163)

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. 163)

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. 163)

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.
CreateLocationSmb

Type: Array of **TagListEntry** (p. 266) objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

**User** (p. 163)

The user who can mount the share, has the permissions to access files and folders in the SMB share.

For information about choosing a user name that ensures sufficient permissions to files, folders, and metadata, see **user**.

Type: String

Length Constraints: Maximum length of 104.

Pattern: `^[^\x5B\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. 165)

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. 272).

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400
Examples

Example

This example illustrates one usage of CreateLocationSmb.

Sample Request

```
{
   "AgentArns": [
      "arn:aws:datasync:us-east-2:11122233444:agent/agent-0b0addbeef44b3nfs",
   ],
   "Domain": "AMAZON",
   "MountOptions": {
      "Version": "SMB3"
   },
   "Password": "string",
   "ServerHostname": "MyServer.amazon.com",
   "Subdirectory": "share",
   "Tags": [
      {
         "Key": "department",
         "Value": "finance"
      }
   ],
   "User": "user-1"
}
```

Example

This example illustrates one usage of CreateLocationSmb.

Sample Response

```
{"arn:aws:datasync:us-east-1:11122233444:location/loc-0f01451b4b2af49"}
```

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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateTask

Creates a task.

A task includes a source location and a destination location, and a configuration that specifies how data is transferred. A task always transfers data from the source location to the destination location. The configuration specifies options such as task scheduling, bandwidth limits, etc. A task is the complete definition of a data transfer.

When you create a task that transfers data between AWS services in different AWS Regions, one of the two locations that you specify must reside in the Region where DataSync is being used. The other location must be specified in a different Region.

You can transfer data between commercial AWS Regions except for China, or between AWS GovCloud (US) Regions.

**Important**

When you use DataSync to copy files or objects between AWS Regions, you pay for data transfer between Regions. This is billed as data transfer OUT from your source Region to your destination Region. For more information, see Data Transfer pricing.

Request Syntax

```json
{
    "CloudWatchLogGroupArn": "string",
    "DestinationLocationArn": "string",
    "Excludes": [
        {
            "FilterType": "string",
            "Value": "string"
        }
    ],
    "Includes": [
        {
            "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",
        "SecurityDescriptorCopyFlags": "string",
        "TaskQueueing": "string",
        "TransferMode": "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. 274).

The request accepts the following data in JSON format.

**CloudWatchLogGroupArn** *(p. 167)*

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. 167)*

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. 167)*

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. 251) objects* 

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

**Includes** *(p. 167)*

A list of filter rules that determines which files to include when running a task. The pattern contains 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. 251) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

Name (p. 167)

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. 167)

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 operation.

Type: Options (p. 257) object

Required: No

Schedule (p. 167)

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.

Type: TaskSchedule (p. 272) object

Required: No

SourceLocationArn (p. 167)

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. 167)

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. 266) objects

Array Members: Minimum number of 0 items. Maximum number of 50 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. 170)**

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. 272)](CommonErrors.html).

**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**

```
{
  "Options": {
    "Atime": "BEST_EFFORT",
    "Gid": "NONE",
    "Mtime": "PRESERVE",
    "PosixPermissions": "PRESERVE",
    "PreserveDevices": "NONE",
    "PreserveDeletedFiles": "PRESERVE",
    "Uid": "NONE",
    "VerifyMode": "POINT_IN_TIME_CONSISTENT",
  }
}
```

170
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-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 V2
- 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

```
{
  "AgentArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**AgentArn** (p. 172)

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.

<table>
<thead>
<tr>
<th>Type</th>
<th>String</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length Constraints: Maximum length of 128.</td>
<td></td>
</tr>
<tr>
<td>Pattern: `arn:(aws</td>
<td>aws-cn</td>
</tr>
<tr>
<td>Required: Yes</td>
<td></td>
</tr>
</tbody>
</table>

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. 272).

**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 V2
- 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

```
{
    "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**LocationArn (p. 174)**

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. 272).

**Internal Exception**

This exception is thrown when an error occurs in the AWS DataSync service.

HTTP Status Code: 500

**Invalid Request Exception**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DeleteTask

Deletes a task.

Request Syntax

{  "TaskArn": "string"}

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

TaskArn  (p. 176)

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. 272).

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 V2
• 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

```
{
   "AgentArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

AgentArn (p. 178)

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

```
{
   "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. 178)

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}$

**CreationTime (p. 178)**

The time that the agent was activated (that is, created in your account).

Type: Timestamp

**EndpointType (p. 178)**

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. 178)**

The time that the agent last connected to DataSync.

Type: Timestamp

**Name (p. 178)**

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. 178)**

The subnet and the security group that DataSync used to access a VPC endpoint.

Type: PrivateLinkConfig (p. 262) object

**Status (p. 178)**

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. 272).

**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

This example illustrates one usage of DescribeAgent.

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 V2
- 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

```
{  
  "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**LocationArn** (p. 181)

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

```
{  
  "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. 181)

The time that the EFS location was created.

Type: Timestamp

**Ec2Config** (p. 181)

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. 250) object

**LocationArn** (p. 181)

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. 181)

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. 272).

**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**

```json
{
    "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50fb"
}
```

**Example**

This example illustrates one usage of DescribeLocationEfs.

**Sample Response**

```json
{
}
```
"CreationTime": "",
"Ec2Config": {
    sg-011719588293d62f"],
    "SubnetArn": "arn:aws:ec2:us-east-2:11122233344:subnet/subnet-f45a0e678",
},
"LocationArn": "arn:aws:datasync:us-east-2:11122233344:location/loc-07db7abfc326c50fb",
"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 V2
- 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 File Server location.

Request Syntax

```
{
   "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**LocationArn (p. 184)**

The Amazon Resource Name (ARN) of the FSx for Windows File Server 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. 184)**

The time that the FSx for Windows File Server location was created.

Type: Timestamp

**Domain (p. 184)**

The name of the Windows domain that the FSx for Windows File 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-]*$

LocationArn (p. 184)
The Amazon Resource Name (ARN) of the FSx for Windows File Server 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. 184)
The URL of the FSx for Windows File Server location that was described.

Type: String
Length Constraints: Maximum length of 4356.
Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z0-9.\-\]+$

SecurityGroupArns (p. 184)
The Amazon Resource Names (ARNs) of the security groups that are configured for the FSx for Windows File Server 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. 184)
The user who has the permissions to access files and folders in the FSx for Windows File Server file system.

Type: String
Length Constraints: Maximum length of 104.
Pattern: ^[^[\x5B\x5D\/;|=,\+\?]{1,104}$

Errors
For information about the errors that are common to all actions, see Common Errors (p. 272).

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 V2
- 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. 274).

The request accepts the following data in JSON format.

**LocationArn (p. 187)**

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-zA-Z0-9]+:[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. 187)**

The time that the NFS location was created.

Type: Timestamp
DescribeLocationNfs

**LocationArn**  (p. 187)

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. 187)

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. 187)

The NFS mount options that DataSync used to mount your NFS share.

Type:  NfsMountOptions (p. 255) object

**OnPremConfig**  (p. 187)

A list of Amazon Resource Names (ARNs) of agents to use for a Network File System (NFS) location.

Type:  OnPremConfig (p. 256) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 272).

**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-2:11122333444:location/loc-07db7abfc326c50aa"
}
```
Example

This example illustrates one usage of DescribeLocationNfs.

Sample Response

```
{
    "CreationTime": 1532660733.39,
    "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50aa",
    "LocationUri": "hostname.amazon.com",
    "OnPremConfig": {
        "AgentArns": [ "arn:aws:datasync:us-east-2:111222333444:agent/agent-0b0addbeef44b3nfs" ]
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeLocationObjectStorage

Returns metadata about a self-managed object storage server location. For more information about self-managed object storage locations, see Creating a location for object storage.

Request Syntax

```

"LocationArn": "string"

```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

`LocationArn` (p. 190)

The Amazon Resource Name (ARN) of the self-managed object storage server location that was described.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `arn:(aws|aws-cn|aws-us-gov|aws-us|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}`

Required: Yes

Response Syntax

```

"AccessKey": "string",
"AgentArns": [ "string" ],
"CreationTime": number,
"LocationArn": "string",
"LocationUri": "string",
"ServerPort": number,
"ServerProtocol": "string"

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

`AccessKey` (p. 190)

Optional. The access key is used if credentials are required to access the self-managed object storage server. If your object storage requires a user name and password to authenticate, use `AccessKey` and `SecretKey` to provide the user name and password, respectively.

Type: String

Pattern: ^.+$

**AgentArns (p. 190)**

The Amazon Resource Name (ARN) of the agents associated with the self-managed object storage server location.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$

**CreationTime (p. 190)**

The time that the self-managed object storage server agent was created.

Type: Timestamp

**LocationArn (p. 190)**

The Amazon Resource Name (ARN) of the self-managed object storage server 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}$

**LocationUri (p. 190)**

The URL of the source self-managed object storage server location that was described.

Type: String

Length Constraints: Maximum length of 4356.

Pattern: ^(efs|nfs|s3|smb|fsxw)://[a-zA-Z\-0-9./\-]+$

**ServerPort (p. 190)**

The port that your self-managed object storage server accepts inbound network traffic on. The server port is set by default to TCP 80 (HTTP) or TCP 443 (HTTPS).

Type: Integer


**ServerProtocol (p. 190)**

The protocol that the object storage server uses to communicate. Valid values are HTTP or HTTPS.

Type: String

Valid Values: HTTPS | HTTP

**Errors**

For information about the errors that are common to all actions, see **Common Errors (p. 272)**.
DescribeLocationObjectStorage

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 V2
- 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. 274).

The request accepts the following data in JSON format.

**LocationArn (p. 193)**

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

```
{
  "AgentArns": [ "string" ],
  "CreationTime": number,
  "LocationArn": "string",
  "LocationUri": "string",
  "S3Config": {
    "BucketAccessRoleArn": "string"
  },
  "S3StorageClass": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**AgentArns (p. 193)**

If you are using DataSync on an AWS Outpost, the Amazon Resource Name (ARNs) of the EC2 agents deployed on your Outpost. For more information about launching a DataSync agent on an AWS Outpost, see Deploy your DataSync agent on AWS Outposts.

Type: Array of strings
Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$`

**CreationTime (p. 193)**

The time that the Amazon S3 bucket location was created.

Type: Timestamp

**LocationArn (p. 193)**

The Amazon Resource Name (ARN) of the Amazon S3 bucket or access point.

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. 193)**

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. 193)**

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. 264) object

**S3StorageClass (p. 193)**

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. 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.

Type: String

Valid Values: STANDARD | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 272).

**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**

```
{
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50s3"
}
```

**Example**

This example illustrates one usage of DescribeLocationS3.

**Sample Response**

```
{
  "CreationTime": 1532660733.39,
  "LocationArn": "arn:aws:datasync:us-east-2:111222333444:location/loc-07db7abfc326c50s3",
  "LocationUri": "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 V2
- 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

```
{
   "LocationArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**LocationArn (p. 196)**

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

```
{
   "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. 196)**

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 4 items.
Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$`

**CreationTime (p. 196)**

The time that the SMB location was created.

Type: Timestamp

**Domain (p. 196)**

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]$`

**LocationArn (p. 196)**

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. 196)**

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. 196)**

The mount options that are available for DataSync to use to access an SMB location.

Type: `SmbMountOptions (p. 265)` object

**User (p. 196)**

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}$`

**Errors**

For information about the errors that are common to all actions, see *Common Errors (p. 272).*

**InternalException**

This exception is thrown when an error occurs in the AWS DataSync service.
HTTP Status Code: 500

**InvalidRequestException**

This exception is thrown when the client submits a malformed request.

HTTP Status Code: 400

## Examples

### Example

This example illustrates one usage of DescribeLocationSmb.

**Sample Request**

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

### Example

This example illustrates one usage of DescribeLocationSmb.

**Sample Response**

```json
{
  "AgentArns": [
    "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 V2
- 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

```
{
  "TaskArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**TaskArn (p. 199)**

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

```
{
  "CloudWatchLogGroupArn": "string",
  "CreationTime": number,
  "CurrentTaskExecutionArn": "string",
  "DestinationLocationArn": "string",
  "DestinationNetworkInterfaceArns": [ "string" ],
  "ErrorCode": "string",
  "ErrorDetail": "string",
  "Excludes": [ 
    { 
      "FilterType": "string",
      "Value": "string"
    }
  ],
  "Includes": [ 
    { 
      "FilterType": "string",
      "Value": "string"
    }
  ],
  "Name": "string",
  "Options": { 
    "Atime": "string",
    "BytesPerSecond": number,
    "Gid": "string",
    "LogLevel": "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. 199)

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. 199)

The time that the task was created.

Type: Timestamp

CurrentTaskExecutionArn (p. 199)

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. 199)

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/[a-z\-0-9a-z]{17}$`

**DestinationNetworkInterfaceArns** *(p. 199)*

The Amazon Resource Names (ARNs) of the destination elastic network interfaces (ENIs) that were created for your subnet.

Type: Array of strings

Length Constraints: Maximum length of 128.

Pattern: `^arn:aws[\-a-z]{0,11}:ec2:[a-z\-0-9]*:[0-9]{12}:network-interface/eni-[0-9a-f]+$`

**ErrorCode** *(p. 199)*

Errors that AWS DataSync encountered during execution of the task. You can use this error code to help troubleshoot issues.

Type: String

**ErrorDetail** *(p. 199)*

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. 199)*

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. 251) objects*

Array Members: Minimum number of 0 items. Maximum number of 1 item.

**Includes** *(p. 199)*

A list of filter rules that determines which files to include when running a task. The pattern contains 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. 251) objects*

Array Members: Minimum number of 0 items. Maximum number of 1 item.

**Name** *(p. 199)*

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\-0-9\s+=._:@/-]+$`

**Options** *(p. 199)*

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 operation.

Type: Options (p. 257) object

Schedule (p. 199)

The schedule used to periodically transfer files from a source to a destination location.

Type: TaskSchedule (p. 272) object

SourceLocationArn (p. 199)

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}$

SourceNetworkInterfaceArns (p. 199)

The Amazon Resource Names (ARNs) of the source elastic network interfaces (ENIs) that were 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. 199)

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. 199)

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}$

Errors

For information about the errors that are common to all actions, see Common Errors (p. 272).

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

```json
{
  "TaskArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026"
}
```

Example

This example illustrates one usage of DescribeTask.

Sample Response

```json
{
  "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": "PRESEVERE",
    "PosixPermissions": "PRESEVERE",
    "PreserveDevices": "NONE",
    "PreserveDeletedFiles": "PRESEVERE",
    "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 V2
• 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

```json
{
   "TaskExecutionArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**TaskExecutionArn (p. 205)**

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

```json
{
   "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"
   }
}
```
"PreserveDeletedFiles": "string",
"PreserveDevices": "string",
"SecurityDescriptorCopyFlags": "string",
"TaskQueueing": "string",
"TransferMode": "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. 205)**

The physical number of bytes transferred over the network.

Type: Long

**BytesWritten (p. 205)**

The number of logical bytes written to the destination AWS storage resource.

Type: Long

**EstimatedBytesToTransfer (p. 205)**

The estimated physical number of bytes that is to be transferred over the network.

Type: Long

**EstimatedFilesToTransfer (p. 205)**

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. 205)**

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. 251)** objects
FilesTransferred \( \text{(p. 205)} \)

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 \( \text{(p. 205)} \)

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 \( \text{(p. 251)} \) objects

Options \( \text{(p. 205)} \)

Represents the options that are available to control the behavior of a StartTaskExecution 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, the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution.

Type: Options \( \text{(p. 257)} \) object

Result \( \text{(p. 205)} \)

The result of the task execution.

Type: TaskExecutionResultDetail \( \text{(p. 268)} \) object

StartTime \( \text{(p. 205)} \)

The time that the task execution was started.

Type: Timestamp

Status \( \text{(p. 205)} \)

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 \( \text{(p. 205)} \)

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-1:111222333444:task/task-0208075f79cedf4a2/execution/exec-08ef1e88ec491019b` executed the task with the ARN `arn:aws:datasync:us-east-1:111222333444:task/task-0208075f79cedf4a2`.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}/execution/exec-[0-9a-f]{17}$`

### Errors

For information about the errors that are common to all actions, see [Common Errors (p. 272)](#).

**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**

```json
{
  "TaskExecutionArn": "arn:aws:datasync:us-east-1:111222333444:task/task-08de6e6697796f026/execution/exec-04ce9d516d69bd52f"
}
```

**Example**

This example illustrates one usage of DescribeTaskExecution.

**Sample Response**

```json
{
  "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-08de6e697796f026/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 V2
- 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

```json
{
    "MaxResults": number,
    "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

MaxResults (p. 210)

The maximum number of agents to list.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

NextToken (p. 210)

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

```json
{
    "Agents": [
        {
            "AgentArn": "string",
            "Name": "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.

**Agents (p. 210)**

A list of agents in your account.

Type: Array of AgentListEntry (p. 249) objects

**NextToken (p. 210)**

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. 272).

**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 V2
- 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
{
  "Filters": [
    {
      "Name": "string",
      "Operator": "string",
      "Values": [ "string" ]
    }
  ],
  "MaxResults": number,
  "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

Filters  (p. 212)

You can use API filters to narrow down the list of resources returned by ListLocations. For example, to retrieve all tasks on a specific source location, you can use ListLocations with filter name LocationType S3 and Operator Equals.

Type: Array of LocationFilter (p. 252) objects

Required: No

MaxResults  (p. 212)

The maximum number of locations to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

NextToken  (p. 212)

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. 213)*

An array that contains a list of locations.

Type: Array of **LocationListEntry** *(p. 253)* objects

**NextToken** *(p. 213)*

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. 272)*.

**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 V2
• 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. 274).

The request accepts the following data in JSON format.

**MaxResults (p. 215)**

The maximum number of locations to return.

- Type: Integer
- Valid Range: Minimum value of 0. Maximum value of 100.
- Required: No

**NextToken (p. 215)**

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. 215)**

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-ios|aws-ios-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",
}
```
ListTagsForResource

"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. 215)

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: [a-zA-Z0-9=_-]+

Tags (p. 215)

Array of resource tags.

Type: Array of TagListEntry (p. 266) 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. 272).

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 V2
- 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](#).

The request accepts the following data in JSON format.

**MaxResults** (p. 218)

The maximum number of executed tasks to list.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

**NextToken** (p. 218)

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. 218)

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. 218)**

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. 218)**

A list of executed tasks.

Type: Array of  TaskExecutionListEntry (p. 267) objects

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 272).

**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 V2
- 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

```
{
  "Filters": [
    {
      "Name": "string",
      "Operator": "string",
      "Values": [ "string" ]
    }
  ],
  "MaxResults": number,
  "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**Filters (p. 221)**

You can use API filters to narrow down the list of resources returned by ListTasks. For example, to retrieve all tasks on a specific source location, you can use ListTasks with filter name LocationId and Operator Equals with the ARN for the location.

Type: Array of TaskFilter (p. 270) objects

Required: No

**MaxResults (p. 221)**

The maximum number of tasks to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

**NextToken (p. 221)**

An opaque string that indicates the position at which to begin the next list of tasks.

Type: String

Length Constraints: Maximum length of 65535.

Pattern: [a-zA-Z0-9_= -]+

Required: No

Response Syntax

```
{
  
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken** *(p. 221)*

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. 221)*

A list of all the tasks that are returned.

Type: Array of TaskListEntry *(p. 271)* objects

Errors

For information about the errors that are common to all actions, see Common Errors *(p. 272).*

**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 V2
- 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

```
{
    "Excludes": [
        {
            "FilterType": "string",
            "Value": "string"
        }
    ],
    "Includes": [
        {
            "FilterType": "string",
            "Value": "string"
        }
    ],
    "OverrideOptions": {
        "Atime": "string",
        "BytesPerSecond": number,
        "Gid": "string",
        "LogLevel": "string",
        "Mtime": "string",
        "OverwriteMode": "string",
        "PosixPermissions": "string",
        "PreserveDeletedFiles": "string",
        "PreserveDevices": "string",
        "SecurityDescriptorCopyFlags": "string",
        "TaskQueueing": "string",
        "TransferMode": "string",
        "Uid": "string",
        "VerifyMode": "string"
    },
    "TaskArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

Excludes (p. 224)

A list of filter rules that determines which files to exclude from a task. The list contains a single filter string that consists of the patterns to exclude. The patterns are delimited by "|" (that is, a pipe), for example, "folder1|folder2".

Type: Array of FilterRule (p. 251) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.
Required: No

Includes (p. 224)

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. 251) objects
Array Members: Minimum number of 0 items. Maximum number of 1 item.
Required: No

OverrideOptions (p. 224)

Represents the options that are available to control the behavior of a StartTaskExecution 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, the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution.

Type: Options (p. 257) object
Required: No

TaskArn (p. 224)

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. 225)

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. 272).

**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-2:111222333444:task/task-08de6e6697796f026"
}
```

Example

This example illustrates one usage of StartTaskExecution.

Sample Response

```json
{
  "TaskExecutionArn": "arn:aws:datasync:us-east-2:111222333444:task/task-08de6e6697796f026/
   execution/exec-04ce9d516d69bd52f"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- 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

```
{
  "ResourceArn": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

ResourceArn (p. 228)

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. 228)

The tags to apply.

Type: Array of TagListEntry (p. 266) objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 272).

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 V2
- 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. 274).

The request accepts the following data in JSON format.

**Keys (p. 230)**

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. 230)**

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. 272).

**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 V2
- 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. 274).

The request accepts the following data in JSON format.

**AgentArn (p. 232)**

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. 232)**

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-Z0-9-0-9\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. 272).

**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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateLocationNfs

Updates some of the parameters of a previously created location for Network File System (NFS) access. For information about creating an NFS location, see Creating a location for NFS.

Request Syntax

```
{
  "LocationArn": "string",
  "MountOptions": {
    "Version": "string"
  },
  "OnPremConfig": {
    "AgentArns": [ "string" ]
  },
  "Subdirectory": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

LocationArn  (p. 234)

The Amazon Resource Name (ARN) of the NFS location to update.

Type: String

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: Yes

MountOptions  (p. 234)

Represents the mount options that are available for DataSync to access an NFS location.

Type: NfsMountOptions  (p. 255) object

Required: No

OnPremConfig  (p. 234)

A list of Amazon Resource Names (ARNs) of agents to use for a Network File System (NFS) location.

Type: OnPremConfig  (p. 256) object

Required: No

Subdirectory  (p. 234)

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

If you are copying data to or from your AWS Snowcone device, see NFS Server on AWS Snowcone for more information.

For information about NFS export configuration, see 18.7. The /etc/exports Configuration File in the Red Hat Enterprise Linux documentation.

Type: String
Length Constraints: Maximum length of 4096.
Pattern: ^\[a-zA-Z0-9_\-+/]+$.
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. 272).

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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
UpdateLocationObjectStorage

Updates some of the parameters of a previously created location for self-managed object storage server access. For information about creating a self-managed object storage location, see Creating a location for object storage.

Request Syntax

```json
{
    "AccessKey": "string",
    "AgentArns": [ "string" ],
    "LocationArn": "string",
    "SecretKey": "string",
    "ServerPort": number,
    "ServerProtocol": "string",
    "Subdirectory": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

AccessKey (p. 237)

Optional. The access key is used if credentials are required to access the self-managed object storage server. If your object storage requires a user name and password to authenticate, use AccessKey and SecretKey to provide the user name and password, respectively.

Type: String


Pattern: ^.+$

Required: No

AgentArns (p. 237)

The Amazon Resource Name (ARN) of the agents associated with the self-managed object storage server location.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: No

LocationArn (p. 237)

The Amazon Resource Name (ARN) of the self-managed object storage server location to be updated.

Type: String
Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}$

Required: Yes

SecretKey (p. 237)

Optional. The secret key is used if credentials are required to access the self-managed object storage server. If your object storage requires a user name and password to authenticate, use AccessKey and SecretKey to provide the user name and password, respectively.

Type: String


Pattern: ^.+$

Required: No

ServerPort (p. 237)

The port that your self-managed object storage server accepts inbound network traffic on. The server port is set by default to TCP 80 (HTTP) or TCP 443 (HTTPS). You can specify a custom port if your self-managed object storage server requires one.

Type: Integer


Required: No

ServerProtocol (p. 237)

The protocol that the object storage server uses to communicate. Valid values are HTTP or HTTPS.

Type: String

Valid Values: HTTPS | HTTP

Required: No

Subdirectory (p. 237)

The subdirectory in the self-managed object storage server that is used to read data from.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^[a-zA-Z0-9 _\-\+\\./\()\p{Zs}]*$

Required: No

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 272).
**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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**UpdateLocationSmb**

Updates some of the parameters of a previously created location for Server Message Block (SMB) file system access. For information about creating an SMB location, see Creating a location for SMB.

**Request Syntax**

```json
{
    "AgentArns": [ "string" ],
    "Domain": "string",
    "LocationArn": "string",
    "MountOptions": {
        "Version": "string"
    },
    "Password": "string",
    "Subdirectory": "string",
    "User": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**AgentArns (p. 240)**

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 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+: [0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: No

**Domain (p. 240)**

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

**LocationArn (p. 240)**

The Amazon Resource Name (ARN) of the SMB location to update.

Type: String

Length Constraints: Maximum length of 128.
UpdateLocationSmb

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:location/loc-[0-9a-z]{17}#

Required: Yes

MountOptions (p. 240)

Represents the mount options that are available for DataSync to access an SMB location.

Type: SmbMountOptions (p. 265) object

Required: No

Password (p. 240)

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: No

Subdirectory (p. 240)

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 that you specified, DataSync must have permissions to mount the SMB share and to access all the data in that share. To ensure this, do either of the following:

• 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.

• Use credentials of a member of the Backup Operators group to mount the share.

Doing either of these options enables the agent to access the data. For the agent to access directories, you must also enable all execute access.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: ^([a-zA-Z0-9\-]+\./\.)?$

Required: No

User (p. 240)

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: 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. 272).

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 V2
- 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"
    }
  ],
  "Includes": [
    {
      "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",
    "SecurityDescriptorCopyFlags": "string",
    "TaskQueueing": "string",
    "TransferMode": "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. 274).

The request accepts the following data in JSON format.

CloudWatchLogGroupArn (p. 243)

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. 243)**

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. 251) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

**Includes (p. 243)**

A list of filter rules that determines which files to include when running a task. The pattern contains 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. 251) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

**Name (p. 243)**

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. 243)**

Represents the options that are available to control the behavior of a StartTaskExecution 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, the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution.

Type: Options (p. 257) object

Required: No

**Schedule (p. 243)**

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.

Type: TaskSchedule (p. 272) object

Required: No
TaskArn (p. 243)

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. 272).

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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateTaskExecution

Updates execution of a task.

You can modify bandwidth throttling for a task execution that is running or queued. For more information, see Adjusting Bandwidth Throttling for a Task Execution.

**Note**
The only Option that can be modified by UpdateTaskExecution is **BytesPerSecond**.

**Request Syntax**

```json
{
    "Options": {
        "Atime": "string",
        "BytesPerSecond": number,
        "Gid": "string",
        "LogLevel": "string",
        "Mtime": "string",
        "OverwriteMode": "string",
        "PosixPermissions": "string",
        "PreserveDeletedFiles": "string",
        "PreserveDevices": "string",
        "SecurityDescriptorCopyFlags": "string",
        "TaskQueueing": "string",
        "TransferMode": "string",
        "Uid": "string",
        "VerifyMode": "string"
    },
    "TaskExecutionArn": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 274).

The request accepts the following data in JSON format.

**Options (p. 246)**

Represents the options that are available to control the behavior of a StartTaskExecution 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, the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution.

Type: Options (p. 257) object

Required: Yes

**TaskExecutionArn (p. 246)**

The Amazon Resource Name (ARN) of the specific task execution that is being updated.

Type: String

Length Constraints: Maximum length of 128.
Data Types

The following data types are supported:

- `AgentListEntry` (p. 249)
- `Ec2Config` (p. 250)
- `FilterRule` (p. 251)
- `LocationFilter` (p. 252)
- `LocationListEntry` (p. 253)
- `NfsMountOptions` (p. 255)
- `OnPremConfig` (p. 256)
- `Options` (p. 257)
• PrivateLinkConfig (p. 262)
• S3Config (p. 264)
• SmbMountOptions (p. 265)
• TagListEntry (p. 266)
• TaskExecutionListEntry (p. 267)
• TaskExecutionResultDetail (p. 268)
• TaskFilter (p. 270)
• TaskListEntry (p. 271)
• TaskSchedule (p. 272)
AgentListEntry

Represents a single entry in a list of agents. AgentListEntry returns an array that contains a list of agents when the ListAgents 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\-\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 V2
- 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-iso|aws-iso-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-iso|aws-iso-b):ec2:[a-z\-0-9]*:[0-9]{12}:subnet/.*$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- 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 V2
- AWS SDK for Ruby V3
LocationFilter

You can use API filters to narrow down the list of resources returned by ListLocations. For example, to retrieve all your Amazon S3 locations, you can use ListLocations with filter name LocationType S3 and Operator Equals.

Contents

Name

The name of the filter being used. Each API call supports a list of filters that are available for it (for example, LocationType for ListLocations).

Type: String

Valid Values: LocationUri | LocationType | CreationTime

Required: Yes

Operator

The operator that is used to compare filter values (for example, Equals or Contains). For more about API filtering operators, see API filters for ListTasks and ListLocations.

Type: String

Valid Values: Equals | NotEquals | In | LessThanOrEqual | LessThan | GreaterThanOrEqual | GreaterThan | Contains | NotContains | BeginsWith

Required: Yes

Values

The values that you want to filter for. For example, you might want to display only Amazon S3 locations.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[0-9a-zA-Z_\ \-\:\*\.\%/\?-_]*$`

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 V2
- 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` 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` operation is called.


`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 V2
- 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 file systems.
- **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 V2
- 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 4 items.

Length Constraints: Maximum length of 128.

Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]+:[0-9]{12}:agent/agent-[0-9a-z]{17}$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
Options

Represents the options that are available to control the behavior of a StartTaskExecution 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, the default value is used. You can override the defaults options on each task execution by specifying an overriding Options value to StartTaskExecution.

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 POSIX group ID (GID) of the file's owners. This option should only be set for NFS, EFS, and S3 locations. For more information about what metadata is copied by DataSync, see Metadata Copied by DataSync.

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. This option is required for cases when you need to run the same task more than one time.

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 Amazon S3 storage classes in DataSync](#) 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. This option should only be set for NFS, EFS, and S3 locations. For more information about what metadata is copied by DataSync, see [Metadata Copied by DataSync](#).

Default value: PRESERVE.

PRESERVE: Preserve POSIX-style permissions (recommended).
NONE: Ignore permissions.

**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 Amazon S3 storage classes in DataSync** 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 re-create the files with that device name and metadata on the destination. DataSync does not copy the contents of such devices, only the name and metadata.

**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

**SecurityDescriptorCopyFlags**

A value that determines which components of the SMB security descriptor are copied from source to destination objects.

This value is only used for transfers between SMB and Amazon FSx for Windows File Server locations, or between two Amazon FSx for Windows File Server locations. For more information about how DataSync handles metadata, see **How DataSync Handles Metadata and Special Files**.
Default value: OWNER_DACL.

**OWNER_DACL:** For each copied object, DataSync copies the following metadata:
- Object owner.
- NTFS discretionary access control lists (DACLs), which determine whether to grant access to an object.

When choosing this option, DataSync does NOT copy the NTFS system access control lists (SACLs), which are used by administrators to log attempts to access a secured object.

**OWNER_DACL_SACL:** For each copied object, DataSync copies the following metadata:
- Object owner.
- NTFS discretionary access control lists (DACLs), which determine whether to grant access to an object.
- NTFS system access control lists (SACLs), which are used by administrators to log attempts to access a secured object.

Copying SACLs requires granting additional permissions to the Windows user that DataSync uses to access your SMB location. For information about choosing a user that ensures sufficient permissions to files, folders, and metadata, see user.

**NONE:** None of the SMB security descriptor components are copied. Destination objects are owned by the user that was provided for accessing the destination location. DACLs and SACLs are set based on the destination server’s configuration.

**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.

**TransferMode**

A value that determines whether DataSync transfers only the data and metadata that differ between the source and the destination location, or whether DataSync transfers all the content from the source, without comparing to the destination location.

**CHANGED:** DataSync copies only data or metadata that is new or different content from the source location to the destination location.

**ALL:** DataSync copies all source location content to the destination, without comparing to existing content on the destination.
Options

Required: No

Uid

The POSIX user ID (UID) of the file's owner. This option should only be set for NFS, EFS, and S3 locations. To learn more about what metadata is copied by DataSync, see Metadata Copied by DataSync.

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.

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 V2
- 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\d(\.(25\[0-5]\|2\[0-4]\d\|0-1)\?\d\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-iso|aws-iso-b):ec2:[a-z\-\d]*[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-iso|aws-iso-b):ec2:[a-z\-\d]*[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 V2
- 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 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 V2
- 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 V2
- 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 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 V2
- 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 the ListTaskExecutions 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 V2
- 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.
Required: No

**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 V2
- AWS SDK for Ruby V3
TaskFilter

You can use API filters to narrow down the list of resources returned by ListTasks. For example, to retrieve all tasks on a source location, you can use ListTasks with filter name LocationId and Operator Equals with the ARN for the location.

Contents

Name
The name of the filter being used. Each API call supports a list of filters that are available for it. For example, LocationId for ListTasks.

Type: String

Valid Values: LocationId | CreationTime

Required: Yes

Operator
The operator that is used to compare filter values (for example, Equals or Contains). For more about API filtering operators, see API filters for ListTasks and ListLocations.

Type: String

Valid Values: Equals | NotEquals | In | LessThanOrEqual | LessThan | GreaterThanOrEqual | GreaterThan | Contains | NotContains | BeginsWith

Required: Yes

Values
The values that you want to filter for. For example, you might want to display only tasks for a specific destination location.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^[0-9a-zA-Z_\-\:\*\./\-\?]*/$

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 V2
- 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 operation is called. A task includes the source and destination file systems to sync and the options to use for the tasks.

Contents

Name

The name of the task.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.
Pattern: ^[a-zA-Z0-9\s+=._:@/-]+$
Required: No

Status

The status of the task.
Type: String
Valid Values: AVAILABLE | CREATING | QUEUED | RUNNING | UNAVAILABLE
Required: No

TaskArn

The Amazon Resource Name (ARN) of the task.
Type: String
Length Constraints: Maximum length of 128.
Pattern: ^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):datasync:[a-z\-0-9]*:[0-9]{12}:task/task-[0-9a-f]{17}$
Required: No

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 V2
- 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 V2
- AWS SDK for Ruby V3

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400
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

NotAuthorized
You do not have permission to perform this action.
HTTP Status Code: 400

OptInRequired
The AWS access key ID needs a subscription for the service.
HTTP Status Code: 403

RequestExpired
The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.
HTTP Status Code: 400
ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

Common 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.
Common Parameters

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:** July 28, 2021

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>New AWS Region</td>
<td>AWS DataSync is now available in the Asia Pacific (Osaka) Region. For more information, see <a href="https://aws.amazon.com">AWS DataSync regions</a> in the <a href="https">AWS General Reference</a>.</td>
<td>July 28, 2021</td>
</tr>
<tr>
<td>Fully automated transfers between AWS storage services</td>
<td>AWS DataSync can now transfer files or objects between Amazon S3, Amazon EFS, or FSx for Windows File Server, with just a few clicks in the DataSync console. For more information, see [Data transfer between AWS storage services](p. 4).</td>
<td>November 9, 2020</td>
</tr>
<tr>
<td>Adjusting the network bandwidth used by a running task</td>
<td>AWS DataSync now enables customers to adjust the network bandwidth used by a running DataSync task. This helps to minimize impact on other users or applications when a task spans multiple days. For more information, see [Adjusting bandwidth throttling for a task execution](p. 68).</td>
<td>November 9, 2020</td>
</tr>
<tr>
<td>Enhanced support for on-premises DataSync virtual machine (VM) functions</td>
<td>The AWS DataSync agent VM host console now supports enhanced functions, including activating an agent from the local console. For more information, see [Working with your agent on the local console](p. 98).</td>
<td>October 19, 2020</td>
</tr>
<tr>
<td>AWS DataSync can now transfer data to and from AWS Outposts</td>
<td>DataSync now supports transferring objects to and from Amazon S3 on AWS Outposts. For more information about setting up such transfers, see <a href="https">Creating a location for Amazon S3</a>.</td>
<td>September 30, 2020</td>
</tr>
<tr>
<td>Support for API filtering</td>
<td>AWS DataSync now supports filtering for the ListTasks operation.</td>
<td>August 18, 2020</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>and ListLocations API calls, enabling you to easily retrieve configuration of data transfer tasks, using filters such as the source or destination for the data transfer. For more information, see API filters for ListTasks and ListLocations (p. 47).</td>
<td>July 27, 2020</td>
<td></td>
</tr>
<tr>
<td>Support for copying data from your self-managed object storage</td>
<td>AWS DataSync now supports data transfer between self-managed object storage and Amazon S3, Amazon Elastic File System, or FSx for Windows File Server. For more information, see Creating a location for object storage (p. 75).</td>
<td>July 27, 2020</td>
</tr>
<tr>
<td>AWS DataSync can now automatically configure your Amazon CloudWatch Logs configuration</td>
<td>When using DataSync, you now have the option of automatically generating the CloudWatch log group and resource policy required to publish logs for your data transfer, simplifying task creation and monitoring setup.</td>
<td>July 1, 2020</td>
</tr>
<tr>
<td>Support for Linux Kernel-based Virtual Machine (KVM) and Microsoft Hyper-V hypervisors</td>
<td>AWS DataSyn now provides the ability to deploy on-premises agents on the KVM and Microsoft Hyper-V virtualization platforms, in addition to the existing VMware and Amazon EC2 options. For more information, see Supported hypervisors (p. 9).</td>
<td>July 1, 2020</td>
</tr>
<tr>
<td>AWS DataSync can now transfer data to and from AWS Snowcone</td>
<td>DataSync now supports transferring files to and from AWS Snowcone, the smallest member of the AWS Snow Family of edge computing and data transfer devices. Snowcone is portable, ruggedized, and secure - small and light enough to fit in a backpack, and able to withstand harsh environments.</td>
<td>June 17, 2020</td>
</tr>
<tr>
<td>New AWS Region</td>
<td>AWS DataSync is now available in the Africa (Capetown) Region and the EU (Milan) Region. For more information, see AWS DataSync Regions in the AWS General Reference</td>
<td>June 16, 2020</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<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 FSx for Windows File Server file systems. For more information about configuring file level logging, see Configuring task settings (p. 61).</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 FSx for Windows File Server. For more information, see Creating a location for FSx for Windows File Server (p. 77).</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. 65).</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>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. 79).</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>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------------</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. 70).</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. 24).</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 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. 90).</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. 62).</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.