Amazon FSx for Lustre

Lustre User Guide

Amazon FSx for Lustre: Lustre User Guide

Table of Contents

What Is Amazon FSx for Lustre?

Amazon FSx for Lustre provides fully managed file systems that are optimized for compute-intensive workloads, such as high-performance computing, machine learning, and media processing workflows. Many of these applications require the high performance and low latencies of scale-out, parallel file systems. Operating these file systems typically requires specialized expertise and administrative overhead, requiring you to provision storage servers and tune complex performance parameters. With Amazon FSx for Lustre, in minutes you can launch and run a Lustre file system that can process massive datasets at up to hundreds of gigabytes per second of throughput, millions of IOPS, and submillisecond latencies.

Amazon FSx for Lustre provides high-performance storage at low cost, because it is nonreplicated, on-demand storage for short-term, compute-intensive processing of datasets. The service is seamlessly integrated with Amazon Simple Storage Service (Amazon S3). Thus, you can automatically copy Amazon S3 data into your file system to run analyses for hours to days, write results back to S3, and then delete your file system. Amazon FSx for Lustre also enables you to burst your compute-intensive workloads from on-premises to the AWS Cloud by importing data over AWS Direct Connect or VPN.

As a fully managed service, Amazon FSx for Lustre eliminates the administrative overhead of setting up and provisioning file servers and storage volumes. Additionally, Amazon FSx for Lustre keeps the underlying software powering your file systems up-to-date and provides rich integration with other AWS services like Amazon S3, Amazon CloudWatch, and AWS CloudTrail.

Assumptions

In this guide, we make the following assumptions:

- If you use Amazon Elastic Compute Cloud (Amazon EC2), we assume that you're familiar with that service. For more information on how to use Amazon EC2, see the Amazon EC2 documentation.
- We assume that you are familiar with using Amazon Virtual Private Cloud (Amazon VPC). For more information on how to use Amazon VPC, see the Amazon VPC User Guide.
- We assume that you haven't changed the rules on the default security group for your VPC based on the Amazon VPC service. If you have, make sure that you add the necessary rules to allow network traffic from your Amazon EC2 instance to your Amazon FSx for Lustre file system. For more details, see Security (p. 29).

Pricing for Amazon FSx for Lustre

With Amazon FSx for Lustre, there are no upfront hardware or software costs. You pay for only the resources used, with no minimum commitments, setup costs, or additional fees. For information about the pricing and fees associated with the service, see Amazon FSx for Lustre Pricing.

Amazon FSx for Lustre Forums

If you encounter issues while using Amazon FSx for Lustre use the forums.

Are You a First-Time User of Amazon FSx for Lustre?

If you are a first-time user of Amazon FSx for Lustre, we recommend that you read the following sections in order:

1. If you're ready to create your first Amazon FSx for Lustre file system, try the Getting Started with Amazon FSx for Lustre (p. 6).
2. For information on performance, see Amazon FSx for Lustre Performance (p. 15).
3. For information on linking your file system to an Amazon S3 bucket data repository, see Using Data Repositories (p. 10).
4. For Amazon FSx for Lustre security details, see Security (p. 29).
5. For information on the scalability limits of Amazon FSx for Lustre, including throughput and file system size, see Limits (p. 39).
6. For information on the Amazon FSx for Lustre API, see the Amazon FSx for Lustre API Reference.

Setting Up

Before you use Amazon FSx for Lustre for the first time, complete the following tasks:

1. Sign Up for AWS (p. 3)
2. Create an IAM User (p. 3)

Sign Up for AWS

When you sign up for Amazon Web Services (AWS), your AWS account is automatically signed up for all services in AWS, including Amazon FSx for Lustre.

If you have an AWS account already, skip to the next task. If you don't have an AWS account, use the following procedure to create one.

To create an AWS account

1. Open https://aws.amazon.com/, and then choose **Create an AWS Account**.

   Note
   If you previously signed in to the AWS Management Console using AWS account root user credentials, choose **Sign in to a different account**. If you previously signed in to the console using IAM credentials, choose **Sign-in using root account credentials**. Then choose **Create a new AWS account**.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code using the phone keypad.

Note your AWS account number, because you need it for the next task.

Create an IAM User

Services in AWS, such as Amazon FSx for Lustre, require that you provide credentials when you access them, so that the service can determine whether you have permissions to access its resources. AWS recommends that you don't use the root credentials of your AWS account to make requests. Instead, create an AWS Identity and Access Management (IAM) user and grant that user full access. We call these users administrator users.

You can use the administrator user credentials, instead of root credentials of your account, to interact with AWS and perform tasks, such as create users and grant them permissions. For more information, see Root Account Credentials vs. IAM User Credentials in the *AWS General Reference* and IAM Best Practices in the *IAM User Guide*.

If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM Management Console.

To create an IAM user for yourself and add the user to an Administrators group

1. Use your AWS account email address and password to sign in as the *AWS account root user* to the IAM console at https://console.aws.amazon.com/iam/.

Note
We strongly recommend that you adhere to the best practice of using the **`Administrator`** IAM user below and securely lock away the root user credentials. Sign in as the root user only to perform a few account and service management tasks.

2. In the navigation pane of the console, choose **Users**, and then choose **Add user**.
3. For **User name**, type **`Administrator`**.
4. Select the check box next to **AWS Management Console access**, select **Custom password**, and then type the new user's password in the text box. You can optionally select **Require password reset** to force the user to create a new password the next time the user signs in.
5. Choose **Next: Permissions**.
6. On the **Set permissions** page, choose **Add user to group**.
7. Choose **Create group**.
8. In the **Create group** dialog box, for **Group name** type **`Administrators`**.
9. For **Filter policies**, select the check box for **AWS managed - job function**.
10. In the policy list, select the check box for **AdministratorAccess**. Then choose **Create group**.
11. Back in the list of groups, select the check box for your new group. Choose **Refresh** if necessary to see the group in the list.
12. Choose **Next: Tags** to add metadata to the user by attaching tags as key-value pairs.
13. Choose **Next: Review** to see the list of group memberships to be added to the new user. When you are ready to proceed, choose **Create user**.

You can use this same process to create more groups and users, and to give your users access to your AWS account resources. To learn about using policies to restrict users' permissions to specific AWS resources, go to Access Management and Example Policies.

To sign in as this new IAM user, first sign out of the AWS Management Console. Then use the following URL, where *your_aws_account_id* is your AWS account number without the hyphens (for example, if your AWS account number is `1234-5678-9012`, your AWS account ID is `123456789012`).

```
https://your_aws_account_id.signin.aws.amazon.com/console/
```

Enter the IAM user name and password that you just created. When you're signed in, the navigation bar displays ***your_user_name@your_aws_account_id***.

If you don't want the URL for your sign-in page to contain your AWS account ID, you can create an account alias. To do so, from the IAM dashboard, choose **Create Account Alias** and enter an alias, such as your company name. To sign in after you create an account alias, use the following URL.

```
https://your_account_alias.signin.aws.amazon.com/console/
```

To verify the sign-in link for IAM users for your account, open the IAM console and check under **AWS Account Alias** on the dashboard.

Adding Permissions to Use Data Repositories in Amazon S3

Amazon FSx for Lustre is deeply integrated with Amazon S3. This integration means that you can seamlessly access the objects stored in your Amazon S3 buckets from applications mounting your Amazon FSx for Lustre file system. For more information, see Using Data Repositories (p. 10).

To use data repositories, you must first allow Amazon FSx for Lustre certain IAM permissions in a role associated with the account for your administrator user.

To embed an inline policy for a role using the console

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Roles**.
3. In the list, choose the name of the role to embed a policy in.
4. Choose the **Permissions** tab.
5. Scroll to the bottom of the page and choose **Add inline policy**.

   Note
   You can't embed an inline policy in a service-linked role in IAM. Because the linked service defines whether you can modify the permissions of the role, you might be able to add additional policies from the service console, API, or AWS CLI. To view the service-linked role documentation for a service, see **AWS Services That Work with IAM** and choose **Yes** in the **Service-Linked Role** column for your service.
6. Choose **Creating Policies with the Visual Editor**
7. Add the following permissions policy statement.

```
{
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "iam:CreateServiceLinkedRole",
            "iam:AttachRolePolicy",
            "iam:PutRolePolicy"
        ],
        "Resource": "arn:aws:iam::*:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/*"
    }
}
```

After you create an inline policy, it is automatically embedded in your role.

For more information about service-linked roles, see Using Service-Linked Roles for Amazon FSx for Lustre (p. 30).

Next Step

Getting Started with Amazon FSx for Lustre (p. 6)

Getting Started with Amazon FSx for Lustre

Following, you can learn how to get started using Amazon FSx for Lustre. These steps walk you through creating an Amazon FSx for Lustre file system, accessing it from your compute instances, and (optionally) using your Amazon FSx for Lustre file system to process the data in your Amazon S3 bucket with your file-based applications.

This getting started exercise includes these steps:

Topics

Prerequisites

To perform this getting started exercise, you need the following:

- An AWS account with the permissions necessary to create an Amazon FSx for Lustre file system and an Amazon EC2 instance. For more information, see Setting Up (p. 3).
- An Amazon EC2 instance running a supported Linux release in your virtual private cloud (VPC) based on the Amazon VPC service. The Lustre client supports Amazon Linux, Amazon Linux 2, CentOS 7.5, RedHat 7.5, SUSE Linux 12 SP3, and Ubuntu 16.04. For this getting started exercise, we recommend CentOS 7.5, which is available in the AWS Marketplace. When creating your Amazon EC2 instance for this getting started exercise, keep the following in mind:
  - We recommend that you create your instance in your default VPC.
  - Verify that an inbound rule exists for the security group you're using with the following values.
    - **Type:** TCP
    - **Protocol:** 6
    - **Port Range:** 988
    - **Source:** Anywhere 0.0.0.0/0

Step 1: Create Your Amazon FSx for Lustre File System

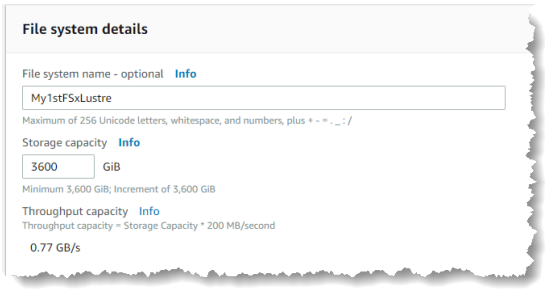
Next, you create your file system in the console.

To create your file system

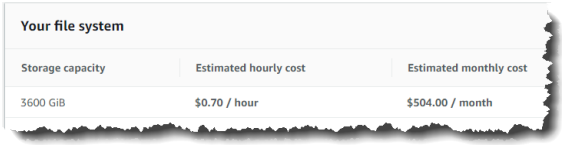
1. Open the Amazon FSx for Lustre console.
2. From the dashboard, choose **Create file system** to start the file system creation wizard.

3. Choose **FSx for Lustre** and then choose **Next** to display the Create File System page.
4. Provide information in the **File system detail** section, following.

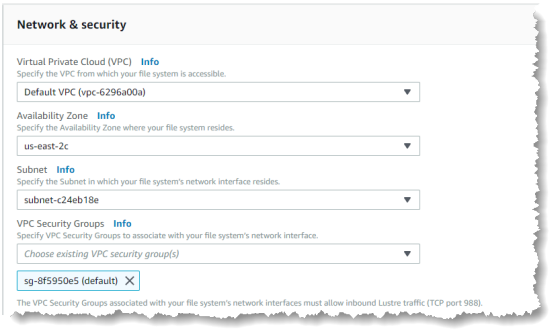
   - Provide a name for your file system. You can use up to 256 Unicode letters, white space, and numbers plus the special characters **+ - = . _ : /**.
   - Provide the **storage capacity** for your file system, in GiB. This value can be any whole number in increments of 3,600 GiB.



   The estimated costs for your file system are displayed in **Your file system**.



5. Provide networking and security group information in the **Network & security** section, following.



   - Choose the VPC that you want to associate with your file system. For the purposes of this getting started exercise, choose the same VPC that you chose for your Amazon EC2 instance.
   - Choose any value for the **Availability Zones** and **Subnet**.
   - For **VPC security groups**, the ID for the default security group for your VPC should be already added. If you're not using the default security group, make sure that the following inbound rule is added to the security group you're using for this getting started exercise.

Type	Protocol	Port Range	Source	Description
TCP	6	988	*The ID of this security group*	Access to your file system over the Lustre protocol

6. (Optional) For **Data repository integration**, choose **Amazon S3** and specify the Amazon S3 bucket (with optional prefix) as the data repository source.

Keep **Export prefix** at the default setting. For more information about the data repository integration, see Using Data Repositories (p. 10)

Important
If you link one or more Amazon FSx for Lustre file systems to an Amazon S3 bucket, don't delete the Amazon S3 bucket until all linked file systems have been deleted.

7. Choose **Review and create**.
8. Review the settings for your Amazon FSx for Lustre file system, and choose **Create file system**.

Now that you've created your file system, make a note of its fully qualified domain name for a later step. You can find the fully qualified domain name for a file system by choosing the name of the file system in the **File Systems** dashboard, and then choosing **Attach**.

Step 2: Install and Configure the Lustre Client on your Instance Before Mounting Your File System

To mount your Amazon FSx for Lustre from your Amazon EC2 instance, first install the Lustre client.

To download the Lustre client onto your Amazon EC2 instance

1. Connect to your Amazon EC2 instance.
2. Install the Lustre client on your CentOS 7.5 instance with the following procedure:
   a. Open a terminal on your client.
   b. Download and install the Lustre client with the following commands. The client comes in two packages that must be downloaded and installed.

```
sudo yum -y install https://downloads.whamcloud.com/public/lustre/lustre-2.10.5/
el7.5.1804/client/RPMS/x86_64/kmod-lustre-client-2.10.5-1.el7.x86_64.rpm
sudo yum -y install https://downloads.whamcloud.com/public/lustre/lustre-2.10.5/
el7.5.1804/client/RPMS/x86_64/lustre-client-2.10.5-1.el7.x86_64.rpm
```

Note
You might need to reboot your compute instance for the client to finish installing.

To mount your file system

1. Make a directory for the mount point with the following command.

```
$ sudo mkdir -p /mnt/fsx
```

2. Mount the Amazon FSx for Lustre file system to the directory that you created. Use the following command and replace *file_system_dns_name* with the actual file system's DNS name.

```
sudo mount -t lustre file_system_dns_name@tcp:/fsx /mnt/fsx
```

3. To see the contents of your data repository in your file system, use the following command.

```
ls /mnt/fsx
```

Step 3: Run Your Analysis

Now that your file system has been created and mounted to a compute instance, you can use it to run your high-performance compute workload.

If you linked your file system to an Amazon S3 data repository, you can export data that you've written to your file system back to your Amazon S3 bucket at any time. From a terminal on one of your compute instances, run the following command to export a file to your Amazon S3 bucket.

```
sudo lfs hsm_archive filename
```

For more information on how to run this command on a folder or large collection of files quickly, see Using Data Repositories (p. 10).

Step 4: Clean Up Resources

After you have finished this exercise, you should follow these steps to clean up your resources and protect your AWS account.

To clean up resources

1. If you want to do a final export, run the following command.

   ```
   nohup find /mnt/fsx -type f -print0 | xargs -0 -n 1 sudo lfs hsm_archive &
   ```

2. On the Amazon EC2 console, terminate your instance. For more information, see Terminate Your Instance in the *Amazon EC2 User Guide for Linux Instances.*
3. On the Amazon FSx for Lustre console, delete your file system with the following procedure:
   a. In the navigation pane, choose **File systems**.
   b. Choose the file system that you want to delete from list of file systems on the dashboard.
   c. For **Actions**, choose **Delete file system**.
   d. In the dialog box that appears, confirm that you want to delete your file system, and choose **Delete file system**.
4. If you created an Amazon S3 bucket for this exercise, and if you don't want to preserve the data you exported, you can now delete it. For more information, see How Do I Delete an S3 Bucket? in the *Amazon Simple Storage Service Console User Guide.*

Using Data Repositories

Amazon FSx for Lustre is designed for short-term, compute-intensive workloads where your long-term data is stored in a durable data repository, such as Amazon S3 or an on-premises data store. With Amazon FSx for Lustre, your file system is generally brought up for the duration of your compute job (typically several hours or days). Amazon FSx for Lustre is cost-optimized for these workloads by providing high-performance, nonreplicated file storage that's designed to be spun up and down as needed. An Amazon FSx for Lustre repository isn't meant to store durable, long-term data.

When you use Amazon FSx for Lustre with a durable storage repository, you can ingest and process large volumes of file data in a high-performance file system. At the same time, you can periodically write intermediate results to your data repository. By taking this approach, you can restart your workload at any time from the latest data you've stored in your data repository. When your workload is done, you can write final results from your file system to your data repository, and delete your file system.

Amazon FSx for Lustre is deeply integrated with Amazon S3. This integration means that you can seamlessly access the objects stored in your Amazon S3 buckets from applications mounting your Amazon FSx for Lustre file system. Amazon FSx for Lustre also supports cloud bursting workloads with on-premises data repositories by enabling you to copy data from on-premises clients using AWS Direct Connect or VPN. When you use Amazon FSx for Lustre with either type of data repository, you can have your data copied into your Amazon FSx for Lustre file system as needed. You can also run your compute-intensive workloads on Amazon EC2 instances in the AWS Cloud and copy your results to your data repository when your workload is done.

> **Important**
> In many cases, you link one or more Amazon FSx for Lustre file systems to an Amazon S3 bucket. If so, don't delete the Amazon S3 bucket until all linked file systems have been deleted.

Importing Data from Your Amazon S3 Bucket

During file system creation, you have the option of Amazon FSx for Lustre automatically preloading a listing of file metadata (file name, size, and modification time) into your file system. Doing this allows clients to view the listing of files in your data repository as soon as your file system is available.

> **Note**
> Importing data from your Amazon S3 data repository happens during file system creation. If you have a large number of files to import, this affects how long it takes for your file system to be created.

Amazon FSx for Lustre automatically copies file data for a given time the first time you open that file from the Amazon S3 data repository into your file system. This data movement is managed by Amazon FSx for Lustre and occurs transparently to your applications. Subsequent reads of these files are served directly out of the file system with consistent submillisecond latencies.

Amazon FSx for Lustre copies data from your Amazon S3 data repository when a file is first accessed. Because of this approach, the initial read or write to a file incurs a small latency penalty. If your application is sensitive to this latency and you know which file your application needs to access, you can optionally preload contents of an individual file. You do so using the `hsm_restore` command, as follows.

You can use the `hsm_action` command to verify that the file's contents have finished loading into the file system. A return value of `NOOP` indicates that the file has successfully been loaded. Run the following commands from a compute instance with the file system mounted.

```
sudo lfs hsm_restore path/to/file
sudo lfs hsm_action path/to/file
```

You can preload the entirety of your file system (or the entirety of a directory within your file system) using the following commands. If you request the preloading of multiple files simultaneously, Amazon FSx for Lustre loads your files from your Amazon S3 data repository in parallel.

```
nohup find local/directory -type f -print0 | xargs -0 -n 1 sudo lfs hsm_restore &
```

Exporting Data to Your Amazon S3 Bucket

In Amazon FSx for Lustre, you can export files that you have written or modified in your file system to your Amazon S3 data repository at any time. When you export a particular file or directory, your file system exports only files that have been created or modified since the last export, or since file system creation.

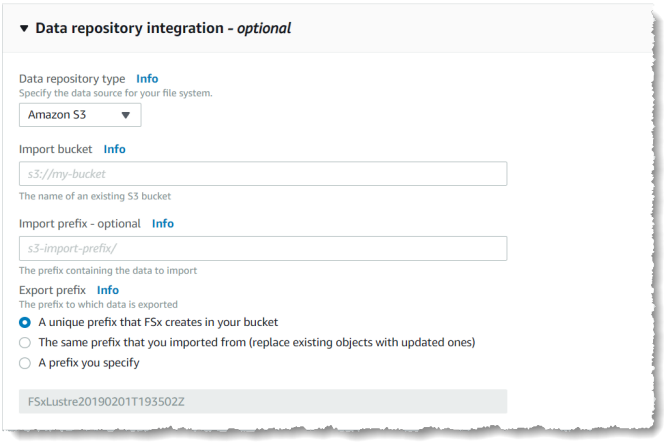
Setting the Export Path in the Console

You can specify an export path when you create your file system using the Amazon FSx console.

To specify an export path using the console

1. Open the Amazon FSx console at https://console.aws.amazon.com/fsx/.
2. From the dashboard, choose **Create file system** to start the file system creation wizard.
3. Choose **Amazon FSx for Lustre** for the file system type, and then choose **Next**.
4. Provide information required for the **File system details** and **Network and security** sections. For more information, see Step 1: Create Your Amazon FSx for Lustre File System (p. 6).
5. Expand the **Data repository integration** section and choose **Amazon S3** for **Data repository type**. Specify the Amazon S3 **Import bucket** and an optional **Import prefix** as the data repository source.
6. Choose one of the three **Export prefix** options:

   - **A unique prefix that Amazon FSx creates in your bucket** – Choose this option to export new and changed objects using a prefix generated by Amazon FSx for Lustre. The prefix looks like the following: `/FSxLustre`*`file-system-creation-timestamp`*. The timestamp is in UTC format, for example `FSxLustre20181105T222312Z`. This option is the default.
   - **The same prefix that you imported from (replace existing objects with updated ones)** – Choose this option to replace existing objects with updated ones.
   - **A prefix you specify** – Choose this option to preserve your imported data and to export new and changed objects using a prefix that you specify.



7. (Optional) Set **Maintenance preferences** or use the system defaults.
8. Choose **Next** and review the file system settings.
9. Choose **Create file system**.

Setting the Export Path Using the AWS CLI

You can specify an export path when you create your file system using the AWS CLI or the Amazon FSx API.

The following example use the `adminuser` as the `profile` parameter value. To provide your own credentials, use an appropriate user profile. For more information about the AWS CLI, see Getting Set Up with the AWS Command Line Interface in the *AWS Command Line Interface User Guide*.

To create an Amazon FSx for Lustre file system using the Amazon FSx CLI

- To create an Amazon FSx for Lustre file system, use the Amazon FSx CLI command `create-file-system`, as shown following. The corresponding API operation is CreateFileSystem.

```
$ aws fsx create-file-system \
      --client-request-token CRT1234 \
      --file-system-type LUSTRE \
      --lustre-configuration import-path=s3://lustre-export-test-bucket/ export-
path=s3://lustre-export-test-bucket/export \
      --storage-capacity 3600 \
      --subnet-ids subnet-123456 \
      --tags Key="Name",Value=Lustre-TEST-1 \
      --region us-east-2
```

After successfully creating the file system, Amazon FSx returns the file system description as JSON, as shown in the following example.

```
{
    "FileSystems": [
        {
            "OwnerId": "owner-id-string",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0d32332e8912345",
            "FileSystemType": "LUSTRE",
```

```
            "Lifecycle": "CREATING",
            "StorageCapacity": 3600,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-039fcf55123456789.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/fs-0d32332e8912345",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "WeeklyMaintenanceStartTime": "2:04:30",
                "DataRepositoryConfiguration": {
                    "ImportPath": "s3://lustre-export-test-bucket/",
                    "ExportPath": "s3://lustre-export-test-bucket/export",
                    "ImportedFileChunkSize": 1024
                }
            }
        }
    ]
}
```

Determining a File System's Export Path

To determine the export path for your file system, call the `aws fsx describe-file-systems` command.

To export an individual file to your data repository and verify that the file has successfully been exported to your data repository, run the commands shown following. A return value of `NOOP` indicates that the file has successfully been exported.

```
sudo lfs hsm_archive path/to/export/file
sudo lfs hsm_action path/to/export/file
```

To export your entire file system (or an entire directory in your file system), run the following commands. If you export multiple files simultaneously, Amazon FSx for Lustre exports your files to your Amazon S3 data repository in parallel.

```
nohup find local/directory -type f -print0 | xargs -0 -n 1 sudo lfs hsm_archive &
```

To determine whether the export has completed, run the following command.

```
find path/to/export/file -type f -print0 | xargs -0 -n 1 -P 8 sudo lfs hsm_action | grep
 "ARCHIVE" | wc -l
```

If the command returns with zero files remaining, then the export is complete.

Using Amazon FSx for Lustre with Your On-Premises Data Repository

You can use Amazon FSx for Lustre to process data stored in your on-premises data repository with in-cloud compute instances. Amazon FSx for Lustre supports access over AWS Direct Connect and VPN, enabling you to mount your file systems from on-premises clients.

To use Amazon FSx for Lustre with your on-premises data

1. Create a file system. For more information, see Step 1: Create Your Amazon FSx for Lustre File System (p. 6) in the getting started exercise.
2. Mount the file system from on-premises clients. For more information, see Mounting from On-Premises or a Peered Amazon VPC (p. 21).
3. Copy the data that you want to process into your Amazon FSx for Lustre file system.
4. Run your compute-intensive workload on in-cloud Amazon EC2 instances mounting your file system. If you want to, you can periodically copy intermediate results to your data repository.
5. When you're finished, copy the final results from your file system back to your on-premises data repository, and delete your Amazon FSx for Lustre file system.

Amazon FSx for Lustre Performance

Amazon FSx for Lustre is built on Lustre, a popular open-source parallel file system that provides scale-out performance that increases linearly with a file system's size. Lustre file systems scale horizontally across multiple file servers and disks, giving each client direct access to the data stored on each disk to remove many of the bottlenecks present in traditional file systems. Amazon FSx for Lustre builds on Lustre's scalable architecture to support high levels of performance across large numbers of clients.

Aggregate File System Performance

Amazon FSx for Lustre file systems scale to hundreds of GBps of throughput and millions of IOPS. Amazon FSx for Lustre also supports concurrent access to the same file or directory from thousands of compute instances. This access enables rapid data checkpointing from application memory to storage, which is a common technique in high-performance computing (HPC).

The throughput that a file system can support is based on its size. With Amazon FSx for Lustre, each terabyte (TiB) of storage capacity supports a baseline of 200 MBps of throughput for file system read operations in aggregate across all client connections to the file system. File systems can also burst for short durations to meet the performance needs of spiky workloads. The following table provides a set of example file systems of different sizes.

File System (GiB)	Baseline Throughput (MBps)
3,600	720
7,200	1,440
14,400	2,880
28,800	5,760
57,760	11,520
115,520	23,040

Regardless of file system size, Amazon FSx for Lustre provides consistent, submillisecond latencies for file operations.

File System Storage Layout

All file data in Lustre is stored on disks called object storage targets (OSTs), and all file metadata (including, file names, timestamps, permissions, and more) is stored on disks called metadata targets (MDTs). Amazon FSx for Lustre file systems are composed of a single MDT and multiple OSTs, each of which is built on SSD storage. A file system's MDT is provisioned to be 3 percent of the total file system's storage capacity, and each OST is approximately 1,200 GiB in size. Amazon FSx for Lustre automatically spreads your file data across the OSTs that make up your file system to balance storage capacity with throughput and IOPS load.

To view the listing and storage utilization of the MDT and OSTs comprising your file system, you can run the following command from a client that has the file system mounted. To view the storage capacity and consumption of each disk, run the following command.

```
lfs df -h mount/path
```

The output of this command looks like the following.

Example

```
UUID                       bytes        Used   Available Use% Mounted on
fsx-MDT0000_UUID         102.8G      216.1M      102.6G   0% /fsx[MDT:0]
fsx-OST0000_UUID           1.1T       44.6M        1.1T   0% /fsx[OST:0]
fsx-OST0001_UUID           1.1T       44.6M        1.1T   0% /fsx[OST:1]
fsx-OST0002_UUID           1.1T       45.5M        1.1T   0% /fsx[OST:2]

filesystem_summary:        3.3T      134.8M        3.3T   0% /fsx
```

Striping Data in your File System

Using Lustre, you can configure how files are striped across OSTs. When a file is striped across multiple OSTs, read or write requests to the file are spread across those OSTs, increasing the aggregate throughput or IOPS your applications can drive through it.

By default, each file created in Amazon FSx for Lustre using standard Linux tools is stored on a single disk. For files imported from Amazon S3, the file system's `ImportedFileChunkSize` parameter determines how many OSTs imported files will be striped across. Files larger than the `ImportedFileChunkSize` will be stored on multiple OSTs.

You can view the striping configuration of a file or directory using the following command:

```
lfs getstripe filename
```

This command reports a file's stripe count, stripe size, and stripe offset. The *stripe count* is how many OSTs the file is striped across. The *stripe size* is how much continuous data is stored on an OST. The *stripe offset* is the index of the first OST that the file is striped across. For more information, see Configuring Lustre File Striping on wiki.lustre.org.

A file's striping parameters are set when the file is first created. Use the following command to create a new, empty file with a determined striping configuration.

```
lfs setstripe filename --stripe-count # of OSTs --stripe-size # of bytes
```

To modify the striping of an existing file, you can create a new file with the desired striping configuration using the `lfs setstripe` command. You then copy the original file into this new file.

Monitoring Performance and Usage

Every minute, Amazon FSx for Lustre emits usage metrics for each disk (MDT and OST) to Amazon CloudWatch.

To view aggregate file system usage details, you can look at the Sum statistic of each metric. For example, the Sum of the `DataReadBytes` statistic reports the total read throughput seen by all the OSTs in a file system. Similarly, the Sum of the `FreeDataStorageCapacity` statistic reports the total available storage capacity for file data in the file system.

For more information on monitoring your file system's performance, see Monitoring Amazon FSx for Lustre (p. 25).

Performance Tips

When using Amazon FSx for Lustre, keep the following performance tips in mind. For service limits, see Limits (p. 39).

- **Average I/O size** – Because Amazon FSx for Lustre is a network file system, each file operation goes through a round trip between the client and Amazon FSx for Lustre, incurring a small latency overhead. Due to this per-operation latency, overall throughput generally increases as the average I/O size increases, because the overhead is amortized over a larger amount of data.
- **Request model** – By enabling asynchronous writes to your file system, pending write operations are buffered on the Amazon EC2 instance before they are written to Amazon FSx for Lustre asynchronously. Asynchronous writes typically have lower latencies. When performing asynchronous writes, the kernel uses additional memory for caching. A file system that has enabled synchronous writes issues synchronous requests to Amazon FSx for Lustre. Every operation goes through a round trip between the client and Amazon FSx for Lustre.

  Note
  Your chosen request model has tradeoffs in consistency (if you're using multiple Amazon EC2 instances) and speed.
- **Amazon EC2 instances** – Applications that perform a large number of read and write operations likely need more memory or computing capacity than applications that don't. When launching your Amazon EC2 instances for your compute-intensive workload, choose instance types that have the amount of these resources that your application needs. The performance characteristics of Amazon FSx for Lustre file systems don't depend on the use of Amazon EBS-optimized instances.
- **Workload balance across OSTs** – In some cases, your workload isn't driving the aggregate throughput that your file system can provide (200 MB/s per TiB of storage). If so, you can use CloudWatch metrics to troubleshoot if performance is affected by an imbalance in your workload's I/O patterns. To identify if this is the cause, look at the Maximum CloudWatch metric for Amazon FSx for Lustre.

  If this statistic shows a load at or above 240 MBps of throughput (the throughput capacity of a single, 1.2 TiB, Amazon FSx for Lustre disk), your workload is not evenly spread out across your disks. If this is the case, you can use the `lfs setstripe` command to modify the striping of files your workload is most frequently accessing. For optimal performance, stripe files with high throughput requirements across all the OSTs comprising your file system.

  If your files are imported from a data repository, you can also modify the `ImportedFileChunkSize` parameter when creating your next Amazon FSx for Lustre file system to stripe your high-throughput files evenly across your OSTs.

  For example, suppose that your workload uses a 7,200 GiB file system (which is comprised of 6x 1,200 GiB OSTs) and needs to drive high throughput across 2.4 GiB files. In this case, you can set the `ImportedFileChunkSize` to `(2.4 GiB / 6 OSTs) = 400 MiB` so that your files are spread evenly across your file system's OSTs.

Accessing File Systems

Using Amazon FSx for Lustre, you can burst your compute-intensive workloads from on-premises into the AWS Cloud by importing data over AWS Direct Connect or VPN. You can access your Amazon FSx for Lustre file system from on-premises, copy data into your file system as-needed, and run compute-intensive workloads on in-cloud instances.

In the following section, you can learn how to access your Amazon FSx for Lustre file system on a Linux instance. In addition, you can find how to use the file `fstab` to automatically remount your file system after any system restarts.

Before you can mount a file system, you must create, configure, and launch your related AWS resources. For detailed instructions, see Getting Started with Amazon FSx for Lustre (p. 6). Next, you can install and configure the Lustre client on your compute instance.

Topics

Installing the Lustre Client

To mount your Amazon FSx for Lustre file system from a Linux instance, first you need to install the open-source Lustre client. Amazon FSx for Lustre supports access from the Lustre client versions 2.10.5 and 2.10.6.

Then, depending on your operating system version, use one of the procedures following.

To install the Lustre client as an RPM package (Amazon Linux)

1. Open a terminal on your client.
2. Determine which kernel is currently running on the compute instance. The Lustre client requires Amazon Linux kernel 4.14, version 104 or higher. Run the following command to determine which kernel is running.

   ```
   uname -r
   ```

   If the command returns `4.14.104-78.84.amzn1.x86_64` or a higher version of 4.14, continue to step 4 to download and install the Lustre client. If the result is less than 4.14.104, continue to the next step to install the supported kernel.
3. Update the kernel and reboot your Amazon EC2 instance by running the following command.

   ```
   sudo yum -y update kernel && sudo reboot
   ```

4. Download and install the Lustre client with the following command.

```
sudo yum install -y lustre-client
```

To install the Lustre client as an RPM package (Amazon Linux 2)

1. Open a terminal on your client.
2. Determine which kernel is currently running on the compute instance. The Lustre client requires Amazon Linux kernel 4.14, version 104 or higher. Run the following command to determine which kernel is running.

```
uname -r
```

   If the command returns `4.14.104-95.84.amzn2.x86_64` or a higher version of 4.14, continue to step 4 to download and install the Lustre client. If the result is less than 4.14.104, continue to the next step to install the supported kernel.
3. Update the kernel and reboot your EC2 instance by running the following command.

```
sudo yum -y update kernel && sudo reboot
```

4. Download and install the Lustre client with the following command.

```
sudo amazon-linux-extras install -y lustre2.10
```

To install the Lustre client as an RPM package (CentOS and RedHat 7.5 or 7.6)

1. Open a terminal on your client.
2. Determine which kernel is currently running on the compute instance with the following command.

```
uname -r
```

   - If the instance is running kernel version `3.10.0-862.*`, then continue with step 3 (p. 19) to download and install the Lustre 2.10.5 client.
   - If the instance is running kernel version `3.10.0-957.*`, then continue with step 4 (p. 19) to download and install the Lustre 2.10.6 client.
3. For instances running linux kernel 3.10.0-862.* – download and install the Lustre 2.10.5 client with the following commands. The client comes in two packages that must be downloaded and installed.

```
sudo yum -y install https://downloads.whamcloud.com/public/lustre/lustre-2.10.5/el7/
client/RPMS/x86_64/kmod-lustre-client-2.10.5-1.el7.x86_64.rpm
sudo yum -y install https://downloads.whamcloud.com/public/lustre/lustre-2.10.5/el7/
client/RPMS/x86_64/lustre-client-2.10.5-1.el7.x86_64.rpm
```

4. For instances running linux kernel 3.10.0-957.* – download and install the Lustre 2.10.6 client with the following commands. The client comes in two packages that you need to download and install.

```
sudo yum -y install https://downloads.whamcloud.com/public/lustre/lustre-2.10.6/el7/
client/RPMS/x86_64/kmod-lustre-client-2.10.6-1.el7.x86_64.rpm
```

```
sudo yum -y install https://downloads.whamcloud.com/public/lustre/lustre-2.10.6/el7/
client/RPMS/x86_64/lustre-client-2.10.6-1.el7.x86_64.rpm
```

Note
You might need to reboot your compute instance for the client to finish installing.

To install the Lustre client as an RPM package (SUSE Linux 12 SP3)

1. Open a terminal on your client.
2. Download and install the Lustre client with the following commands. The client comes in two packages that must be downloaded and installed.

```
sudo rpm -ivh https://downloads.whamcloud.com/public/lustre/lustre-2.10.6/sles12sp3/
client/RPMS/x86_64/lustre-client-kmp-default-2.10.6_k4.4.155_94.50-1.x86_64.rpm
sudo rpm -ivh https://downloads.whamcloud.com/public/lustre/lustre-2.10.6/sles12sp3/
client/RPMS/x86_64/lustre-client-2.10.6-1.x86_64.rpm
```

Note
You might need to reboot your compute instance for the client to finish installing.

To install the Lustre client as a .deb package (Ubuntu 16.04)

1. Open a terminal on your client.
2. Determine which kernel is currently running on the compute instance. The Lustre client requires kernel 4.4.0-131-generic. Run the following command to determine which kernel is running.

```
uname -r
```

   If the command returns `4.4.0-131-generic`, then continue with step 7 (p. 20) to download the Lustre client. If the result is not `4.4.0-131-generic`, continue to the next step to install the supported kernel and headers.
3. Install the supported kernel and associated Linux kernel headers with the following commands.

```
sudo apt-get install linux-image-4.4.0-131-generic
sudo apt-get install -y linux-headers-4.4.0-131-generic
```

4. After installation completes, edit the `/etc/default/grub` file with the following command.

```
sudo sed -i 's/GRUB_DEFAULT=.\+/GRUB\_DEFAULT="Advanced options for Ubuntu>Ubuntu, with
 Linux 4.4.0-131-generic"/' /etc/default/grub
```

5. Update the file with the following command.

```
sudo update-grub
```

6. Reboot your instance with the following command.

```
sudo reboot
```

7. Download the Lustre client with the following commands. The client comes in two packages that must be downloaded.

```
wget https://downloads.whamcloud.com/public/lustre/lustre-2.10.6/ubuntu1604/client/
lustre-client-modules-4.4.0-131-generic_2.10.6-1_amd64.deb
wget https://downloads.whamcloud.com/public/lustre/lustre-2.10.6/ubuntu1604/client/
lustre-utils_2.10.6-1_amd64.deb
```

8. Install the Lustre client packages with the following command.

```
sudo apt-get install -y ./lustre-*_2.10.6*.deb
```

Mounting from an Amazon EC2 Instance

You can mount your file system from an Amazon EC2 instance.

To mount your file system from Amazon EC2

1. Make a directory on your Amazon FSx for Lustre file system for the mount point with the following command.

```
$ sudo mkdir -p /mnt/fsx
```

2. Connect to your Amazon EC2 instance.
3. Mount the Amazon FSx for Lustre file system to the directory that you created. Use the following command and replace *file_system_dns_name* with the actual file system's DNS name.

```
sudo mount -t lustre file_system_dns_name@tcp:/fsx /mnt/fsx
```

4. View the contents of your data repository in your file system by using the following command.

```
ls /mnt/fsx
```

Mounting from On-Premises or a Peered Amazon VPC

You can access your Amazon FSx for Lustre file system in two ways. One is from Amazon EC2 instances located in an Amazon VPC that's peered to the file system's VPC. The other is from on-premises clients that are connected to your file system's VPC using AWS Direct Connect or VPN.

You can mount your file system from outside its VPC using the IP address of its primary network interface. The primary network interface is the first network interface returned when you run the `aws fsx describe-file-systems` AWS CLI command. You can also get this IP address from the AWS Management Console.

To get the IP address of the primary network interface for a file system

1. Open the Amazon FSx console at https://console.aws.amazon.com/fsx/.
2. In the navigation pane, choose **File systems**.
3. Choose your file system from the dashboard.
4. From the file system details page, choose **Network & security**.
5. For **Network interface**, choose the ID for your primary elastic network interface. Doing this takes you to the Amazon EC2 console.

6. On the **Details** tab, find the **Primary private IPv4 IP**. This is the IP address for your primary network interface.

   Note
   You can't use Domain Name System (DNS) name resolution when mounting an Amazon FSx for Lustre file system from outside the VPC it is associated with.

Mounting Your Amazon FSx for Lustre File System Automatically

You can use `fstab` to automatically mount your Amazon FSx for Lustre file system when the Amazon EC2 instance it is mounted on reboots. You can set up automatic mounting in two ways. You can update the `/etc/fstab` file in your Amazon EC2 instance after you connect to the instance for the first time, or you can configure automatic mounting of your Amazon FSx for Lustre file system when you create your Amazon EC2 instance.

Updating an Existing EC2 Instance to Mount Automatically

To automatically remount your Amazon FSx for Lustre file system directory when the Amazon EC2 instance reboots, you can use the `fstab` file. The `fstab` file contains information about file systems. The command `mount -a`, which runs during instance startup, mounts the file systems listed in the `fstab` file.

Note
Before you can update the `/etc/fstab` file of your EC2 instance, make sure that you've already created your Amazon FSx for Lustre file system. For more information, see Step 1: Create Your Amazon FSx for Lustre File System (p. 6) in the Getting Started exercise.

To update the /etc/fstab file in your EC2 instance

1. Connect to your EC2 instance, and open the `/etc/fstab` file in an editor.
2. Add the following line to the `/etc/fstab` file.

```
file_system_dns_name@tcp:/fsx /mnt/fsx lustre defaults,_netdev 0 0
```

   Warning
   Use the `_netdev` option, used to identify network file systems, when mounting your file system automatically. If `_netdev` is missing, your EC2 instance might stop responding. This result is because network file systems need to be initialized after the compute instance starts its networking. For more information, see Automatic Mounting Fails and the Instance Is Unresponsive (p. 40).
3. Save the changes to the file.

Your EC2 instance is now configured to mount the Amazon FSx for Lustre file system whenever it restarts.

Note
In some cases, your Amazon EC2 instance might need to start regardless of the status of your mounted Amazon FSx for Lustre file system. In these cases, add the `nofail` option to your file system's entry in your `/etc/fstab` file.

The fields in the line of code that you added to the `/etc/fstab` file do the following.

Field	Description
file_system_dns_name@tcp:/fsx	The DNS name for your Amazon FSx for Lustre file system, which identifies the file system. You can get this name from the console or programmatically from the AWS CLI or an AWS SDK.
/mnt/fsx	The mount point for the Amazon FSx for Lustre file system on your EC2 instance.
`lustre`	Describes the type of file system, Amazon FSx for Lustre.
`mount options`	Mount options for the file system, presented as a comma-separated list of the following options: • `defaults` – This value tells the operating system to use the default mount options. You can list the default mount options after the file system has been mounted by viewing the output of the `mount` command. • `_netdev` – The value tells the operating system that the file system resides on a device that requires network access. This option prevents the instance from mounting the file system until the network has been enabled on the client.
`0`	A value that indicates whether the file system should be backed up by `dump`. For Amazon FSx, this value should be `0`.
`0`	A value that indicates the order in which `fsck` checks file systems at boot. For Amazon FSx file systems, this value should be `0` to indicate that `fsck` should not run at startup.

Unmounting File Systems

Before you delete a file system, we recommend that you unmount it from every Amazon EC2 instance that it's connected to. You can unmount a file system on your Amazon EC2 instance by running the `umount` command on the instance itself. You can't unmount an Amazon FSx for Lustre file system through the AWS CLI, the AWS Management Console, or through any of the AWS SDKs. To unmount an Amazon FSx for Lustre file system connected to an Amazon EC2 instance running Linux, use the `umount` command as follows:

```
umount /mnt/fsx
```

We recommend that you do not specify any other `umount` options. Avoid setting any other `umount` options that are different from the defaults.

You can verify that your Amazon FSx for Lustre file system has been unmounted by running the `df` command. This command displays the disk usage statistics for the file systems currently mounted on your Linux-based Amazon EC2 instance. If the Amazon FSx for Lustre file system that you want to unmount isn't listed in the `df` command output, this means that the file system is unmounted.

Example – Identify the Mount Status of an Amazon FSx for Lustre File System and Unmount It

```
$ df -T
```

```
Filesystem Type 1K-blocks Used Available Use% Mounted on
file-system-id.fsx.aws-region.amazonaws.com@tcp:/fsx /mnt/fsx 3547708416 61440 3547622400
 1% /mnt/fsx
```

```
$ umount /mnt/fsx
```

```
$ df -T
```

```
Filesystem Type 1K-blocks Used Available Use% Mounted on
/dev/sda1 ext4 8123812 1138920 6884644 15% /
```

Monitoring Amazon FSx for Lustre

With Amazon FSx for Lustre, you can monitor activity for your file systems using Amazon CloudWatch metrics.

Monitoring with Amazon CloudWatch

You can monitor file systems using Amazon CloudWatch, which collects and processes raw data from Amazon FSx for Lustre into readable, near real-time metrics. These statistics are retained for a period of 15 months, so that you can access historical information and gain a better perspective on how your web application or service is performing. By default, Amazon FSx for Lustre metric data is automatically sent to CloudWatch at 1-minute periods. For more information about CloudWatch, see What Are Amazon CloudWatch, Amazon CloudWatch Events, and Amazon CloudWatch Logs? in the *Amazon CloudWatch User Guide*.

As with Amazon S3 and Amazon EBS, Amazon FSx for Lustre CloudWatch metrics are reported as raw *Bytes*. Bytes are not rounded to either a decimal or binary multiple of the unit.

Amazon FSx for Lustre publishes the following metrics into the `AWS/FSx` namespace in CloudWatch. For each metric, Amazon FSx for Lustre emits a data point per disk per minute. To view aggregate file system details, you can use the `Sum` statistic. Note that the file servers behind your Amazon FSx for Lustre file systems are spread across multiple disks.

Metric	Description
`DataReadBytes`	The number of bytes for each file system read operation. The `Sum` statistic is the total number of bytes associated with read operations. The `Minimum` statistic is the minimum number of bytes associated with read operations on a single disk. The `Maximum` statistic is the maximum number of bytes associated with read operations on the disk. The `Average` statistic is the average number of bytes associated with read operations per disk. The `SampleCount` statistic is the number of disks. Units: • Bytes for `Sum`, `Minimum`, `Maximum`, and `Average`. • Count for `SampleCount`. Valid statistics: `Sum`, `Minimum`, `Maximum`, `Average`, `SampleCount`
`DataWriteBytes`	The number of bytes for each file system write operation. The `Sum` statistic is the total number of bytes associated with write operations. The `Minimum` statistic is the minimum number of bytes associated with write operations on a single disk. The `Maximum` statistic is the maximum number of bytes associated with write operations on the disk. The `Average` statistic is the average number of bytes associated with write operations per disk. The `SampleCount` statistic is the number of disks. Units: • Bytes for `Sum`, `Minimum`, `Maximum`, and `Average`.

Metric	Description
	• Count for `SampleCount`. Valid statistics: `Sum`, `Minimum`, `Maximum`, `Average`, `SampleCount`
`DataReadOperations`	The number of read operations. The `Sum` statistic is the total number of read operations. The `Minimum` statistic is the minimum number of read operations on a single disk. The `Maximum` statistic is the maximum number read operations on the disk. The `Average` statistic is the average number of read operations per disk. The `SampleCount` statistic is the number of disks. Units: • Bytes for `Sum`, `Minimum`, `Maximum`, and `Average`. • Count for `SampleCount`. Valid statistics: `Sum`, `Minimum`, `Maximum`, `Average`, `SampleCount`
`DataWriteOperations`	The number of write operations. The `Sum` statistic is the total number of write operations. The `Minimum` statistic is the minimum number of write operations on a single disk. The `Maximum` statistic is the maximum number write operations on the disk. The `Average` statistic is the average number of write operations per disk. The `SampleCount` statistic is the number of disks. Units: • Bytes for `Sum`, `Minimum`, `Maximum`, and `Average`. • Count for `SampleCount`. Valid statistics: `Sum`, `Minimum`, `Maximum`, `Average`, `SampleCount`
`MetadataOperations`	The number of metadata operations. The `Sum` statistic is the count of metadata operations. The `Minimum` statistic is the minimum number of metadata operations per disk. The `Maximum` statistic is the maximum number of metadata operations per disk. The `Average` statistic is the average number of metadata operations per disk. The `SampleCount` statistic is the number of disks. Units: • Count for `Sum`, `Minimum`, `Maximum`, `Average`, and `SampleCount`. Valid statistics: `Sum`, `Minimum`, `Maximum`, `Average`, `SampleCount`

Metric	Description
`FreeDataStorageCapacity`	The total bytes available. The `Sum` statistic is the total number of bytes available. The `Minimum` statistic is the total number bytes available in the fullest disk. The `Maximum` statistic is the total number of bytes available in the disk with the most remaining available storage. The `Average` statistic is the average number of bytes available per disk. The `SampleCount` statistic is the number of disks. Units: • Bytes for `Sum` and `Minimum`. Valid statistics: `Sum`, `Minimum`, `Maximum`, `Average`, `SampleCount`

Amazon FSx for Lustre Dimensions

Amazon FSx for Lustre metrics use the `FSx` namespace and provide metrics for a single dimension, `FileSystemId`. A file system's ID can be found using the `aws fsx describe-file-systems` AWS CLI command, and it takes the form of *`fs-01234567890123456`*.

How to Use Amazon FSx for Lustre Metrics

The metrics reported by Amazon FSx for Lustre provide information that you can analyze in different ways. The list following shows some common uses for the metrics. These are suggestions to get you started, not a comprehensive list.

How Do I Determine...	Relevant Metrics
My file system's throughput?	SUM(DataReadBytes + DataWriteBytes)
My file system's IOPS?	Total IOPS = SUM(DataReadOperations + DataWriteOperations + MetadataOperations)

Accessing CloudWatch Metrics

You can see Amazon FSx for Lustre metrics for CloudWatch in many ways. You can view them through the CloudWatch console, or you can access them using the CloudWatch CLI or the CloudWatch API. The following procedures show you how to access the metrics using these various tools.

To view metrics using the CloudWatch console

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose **Metrics**.
3. Select the **FSx** namespace.
4. (Optional) To view a metric, type its name in the search field.
5. (Optional) To filter by dimension, select **FileSystemId**.

To access metrics from the AWS CLI

- Use the `list-metrics` command with the `--namespace "AWS/FSx"` namespace. For more information, see the AWS CLI Command Reference.

To access metrics from the CloudWatch API

- Call `GetMetricStatistics`. For more information, see Amazon CloudWatch API Reference.

Creating CloudWatch Alarms to Monitor Amazon FSx for Lustre

You can create a CloudWatch alarm that sends an Amazon SNS message when the alarm changes state. An alarm watches a single metric over a time period you specify, and performs one or more actions based on the value of the metric relative to a given threshold over a number of time periods. The action is a notification sent to an Amazon SNS topic or Auto Scaling policy.

Alarms invoke actions for sustained state changes only. CloudWatch alarms don't invoke actions simply because they are in a particular state; the state must have changed and been maintained for a specified number of periods.

The following procedures outline how to create alarms for Amazon FSx for Lustre.

To set alarms using the CloudWatch console

1. Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. Choose **Create Alarm**. This launches the **Create Alarm Wizard**.
3. Choose **FSx Metrics** and scroll through the Amazon FSx for Lustre metrics to locate the metric you want to place an alarm on. To display just the Amazon FSx for Lustre metrics in this dialog box, search on the file system id of your file system. Select the metric to create an alarm on and choose **Next**.
4. Fill in the **Name**, **Description**, **Whenever** values for the metric.
5. If you want CloudWatch to send you an email when the alarm state is reached, in the **Whenever this alarm:** field, choose **State is ALARM**. In the **Send notification to:** field, choose an existing SNS topic. If you select **Create topic**, you can set the name and email addresses for a new email subscription list. This list is saved and appears in the field for future alarms.

   Note
   If you use **Create topic** to create a new Amazon SNS topic, the email addresses must be verified before they receive notifications. Emails are only sent when the alarm enters an alarm state. If this alarm state change happens before the email addresses are verified, they do not receive a notification.
6. At this point, the **Alarm Preview** area gives you a chance to preview the alarm you're about to create. Choose **Create Alarm**.

To set an alarm using the AWS CLI

- Call `put-metric-alarm`. For more information, see *AWS CLI Command Reference*.

To set an alarm using the CloudWatch API

- Call `PutMetricAlarm`. For more information, see *Amazon CloudWatch API Reference*.

Security

Amazon FSx for Lustre provides various features to secure your file systems. In the following sections, you can find information on how to secure your file system data and configure access controls for your Amazon FSx for Lustre file systems.

Amazon FSx for Lustre is PCI-DSS and ISO compliant and HIPAA eligible.

File System Access Control with Amazon VPC

An Amazon FSx for Lustre file system is accessible through an elastic network interface that resides in the Amazon Virtual Private Cloud (Amazon VPC) that you associate with your file system. You access your Amazon FSx for Lustre file system through its DNS name, which maps to the file system's network interface. Only resources within the associated VPC, or a peered VPC, can access your file system's network interface. For more information, see What is Amazon VPC? in the *Amazon VPC User Guide.*

Amazon VPC Security Groups

To further control network traffic going through your file system's network interface within your VPC, you use security groups to limit access to your file systems. A *security group* acts as a virtual firewall to control the traffic for its associated resources. In this case, the associated resource is your file system's network interface.

To use a security group to control access to your Amazon FSx for Lustre file system, you add the inbound rules to control incoming traffic and outbound rules to control the outgoing traffic from your file system. Make sure to have the right network traffic rules in your security group to map your Amazon FSx for Lustre file system's file share to a folder on your supported compute instance.

For more information on security group rules, see Security Group Rules in the *Amazon EC2 User Guide for Linux Instances.*

To create a security group for Amazon FSx for Lustre

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2.
2. In the navigation pane, choose **Security Groups**.
3. Choose **Create Security Group**.
4. Specify a name and description for the security group.
5. For **VPC**, choose the Amazon VPC associated with your file system to create the security group within that VPC.
6. Add the following inbound rules.

Type	Protocol	Port Range	Source	Description
Lustre	TCP	988	Anywhere 0.0.0.0/0	Lustre traffic over TCP for Amazon FSx for Lustre

To associate a security group with your Amazon FSx for Lustre file system

1. Open the Amazon FSx console at https://console.aws.amazon.com/fsx/.

2. On the console dashboard, select your file system to view its details.
3. From the menu, locate the **Network & Security** tab, and choose your file system's network interface IDs (for example, ENI-01234567890123456). Doing this redirects you to the Amazon EC2 console.
4. Choose each network interface ID. Each action opens a new instance of the Amazon EC2 console in your browser. For each security group, choose **Actions**, **Change Security Groups**.
5. In the **Change Security Groups** dialog box, select the security groups to use, and choose **Save**.

Amazon VPC Network ACLs

Another option for securing access to the file system within your VPC is to establish network access control lists (network ACLs). Network ACLs are separate from security groups, but have similar functionality to add an additional layer of security to the resources in your VPC. For more information on network ACLs, see Network ACLs in the *Amazon VPC User Guide.*

Administration Access Control with IAM for Amazon FSx for Lustre Resources

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 AWS Identity and Access Management (IAM) identities (that is, users, groups, and roles). 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

Resources and Operations for Amazon FSx for Lustre

In Amazon FSx for Lustre, the primary resource is a *file system*.

These resources have unique Amazon Resource Names (ARNs) associated with them as shown in the following table.

Amazon FSx for Lustre provides a set of operations to work with Amazon FSx for Lustre resources. For a list of available operations, see the Amazon FSx for Lustre API Reference.

Using Service-Linked Roles for Amazon FSx for Lustre

Amazon FSx for Lustre uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that is linked directly to Amazon FSx for Lustre. Service-linked

roles are predefined by Amazon FSx for Lustre and include all the permissions that the service requires to call other AWS services on your behalf.

Amazon FSx for Lustre defines the permissions of its service-linked roles, and unless defined otherwise, only Amazon FSx for Lustre can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

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

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

Service-Linked Role Permissions for Amazon FSx for Lustre

Amazon FSx for Lustre uses two service-linked roles named `AWSServiceRoleForAmazonFSx` and `AWSServiceRoleForFSxS3Access_`*`fs-01234567890`* that perform certain actions in your account. Examples of these actions are creating elastic network interfaces for your file systems in your VPC and accessing your data repository in an Amazon S3 bucket. For `AWSServiceRoleForFSxS3Access_`*`fs-01234567890`*, this service-linked role is created for each Amazon FSx for Lustre file system you create that is linked to an S3 bucket.

For `AWSServiceRoleForAmazonFSx`, the role permissions policy allows Amazon FSx for Lustre to complete the following actions on the all applicable AWS resources:

- `ec2:CreateNetworkInterface`

For `AWSServiceRoleForFSxS3Access_`*`fs-01234567890`*, the role permissions policy allows Amazon FSx for Lustre to complete the following actions on your Amazon S3 bucket hosting your data repository.

- `s3:AbortMultipartUpload`
- `s3:DeleteObject`
- `s3:PutObject`
- `s3:Get*`
- `s3:List*`

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see Service-Linked Role Permissions in the *IAM User Guide*.

Creating a Service-Linked Role for Amazon FSx for Lustre

You don't need to manually create a service-linked role. When you create a file system in the AWS Management Console, the IAM CLI, or the IAM API, Amazon FSx for Lustre creates the service-linked roles for you.

> **Important**
> The service-linked roles can appear in your account if you completed an action in another service that uses the features supported by this role. To learn more, see A New Role Appeared in My IAM Account.

If you delete these service-linked roles, and then need to create them again, you can use the same process to recreate the role in your account. When you create a file system, Amazon FSx for Lustre creates the service-linked role for you again.

Editing a Service-Linked Role for Amazon FSx for Lustre

Amazon FSx for Lustre does not allow you to edit these service-linked roles. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a Service-Linked Role in the *IAM User Guide*.

Deleting a Service-Linked Role for Amazon FSx for Lustre

If you no longer need to use a feature or service that requires a service-linked role, we recommend that you delete that role. That way you don't have an unused entity that is not actively monitored or maintained. However, you must delete all of your file systems and backups before you can manually delete the service-linked role.

> **Note**
> If the Amazon FSx for Lustre service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

To manually delete a service-linked role using IAM

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

Supported Regions for Amazon FSx for Lustre Service-Linked Roles

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

Understanding Resource Ownership

The AWS account owns the resources that are created in the account, regardless of who created the resources. Specifically, the resource owner is the AWS account of the principal entity (that is, the root account, an IAM user, or an IAM role) that authenticates the resource creation request. The following examples illustrate how this works:

- If you use the root account credentials of your AWS account to create a file system, your AWS account is the owner of the resource. In Amazon FSx for Lustre, the resource is the file system.
- If you create an IAM user in your AWS account and grant permissions to create a file system to that user, the user can create a file system. However, your AWS account, to which the user belongs, owns the file system resource.
- If you create an IAM role in your AWS account with permissions to create a file system, anyone who can assume the role can create a file system. Your AWS account, to which the role belongs, owns the file system 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 Amazon FSx for Lustre. It doesn't provide detailed information about the IAM service. For complete IAM documentation, see What Is IAM? in the *IAM User Guide*. For information about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the *IAM User Guide*.

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

Amazon FSx for Lustre API Permissions: Actions, Resources, and Conditions Reference

When you are setting up access control and writing a permissions policy that you can attach to an IAM identity (identity-based policies), you can use the following as a reference. The each Amazon FSx for Lustre 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 Amazon FSx for Lustre policies to express conditions. For a complete list of AWS-wide keys, see Available Keys in the *IAM User Guide*.

To specify an action, use the `fsx:` prefix followed by the API operation name (for example, `fsx:CreateFileSystem`). Each action applies to either a single Amazon FSx for Lustre file system, to all Amazon FSx for Lustre file systems owned by an AWS account.

Amazon FSx for Lustre API and Required Permissions for Actions

Amazon FSx for Lustre API Operations	Required Permissions (API Actions)	Resource
CreateFileSystem	`fsx:*`	`arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/*`
DeleteFileSystem	`fsx:DeleteFileSystem`	`arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/*` `arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/`*`filesystem-id`*
DescribeFileSystems	`fsx:DescribeFileSystems`	N/A
UpdateFileSystem	`fsx:UpdateFileSystem`	`arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/*` `arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/`*`filesystem-id`*
ListTagsForResource	`fsx:ListTagsForResource`	`arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/*` `arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/`*`filesystem-id`*
TagResource	`fsx:TagResource`	`arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/*` `arn:aws:fsx:`*`region`*`:`*`account-id`*`:file-system/`*`filesystem-id`*

Amazon FSx for Lustre API Operations	Required Permissions (API Actions)	Resource

UntagResource	`fsx:UntagResource`	`arn:aws:fsx:region:account-id:file-system/*`
		
		`arn:aws:fsx:region:account-id:file-system/filesystem-id`

Encryption

All Amazon FSx for Lustre file systems are encrypted at rest with keys managed by the service. Data is encrypted using an XTS-AES-256 block cipher. Data is automatically encrypted before being written to the file system, and automatically decrypted as it is read. These processes are handled transparently by Amazon FSx for Lustre, so you don't have to modify your applications.

Logging Amazon FSx for Windows File Server API Calls with AWS CloudTrail

Amazon FSx is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon FSx. CloudTrail captures all API calls for Amazon FSx as events. Captured calls include calls from the Amazon FSx console and from code calls to Amazon FSx API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon FSx. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Amazon FSx. You can also determine the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Amazon FSx Information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When API activity occurs in Amazon FSx, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for Amazon FSx, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all AWS Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following topics in the *AWS CloudTrail User Guide:*

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
- Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

All Amazon FSx API calls are logged by CloudTrail. For example, calls to the `CreateFileSystem` and `TagResource` operations generate entries in the CloudTrail log files.

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

- Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.

- Whether the request was made by another AWS service.

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

Understanding Amazon FSx 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 `TagResource` operation when a tag for a file system is created from the console.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": “Root”,
        "principalId": “111122223333”,
        "arn": "arn:aws:sts::111122223333:root”,
        "accountId": “111122223333”,
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE”,
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-11-14T22:36:07Z"
            }
        }
    },
    "eventTime": "2018-11-14T22:36:07Z",
    "eventSource": "fsx.amazonaws.com",
    "eventName": "TagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": “192.0.2.0”,
    "userAgent": “console.amazonaws.com”,
    "requestParameters": {
        "resourceARN": "arn:aws:fsx:us-east-1:111122223333:file-system/fs-
ab12cd34ef56gh789”
    },
    "responseElements": null,
    "requestID": “aEXAMPLE-abcd-1234-56ef-b4cEXAMPLE51”,
    "eventID": “bEXAMPLE-gl12-3f5h-3sh4-ab6EXAMPLE9p”,
    "eventType": "AwsApiCall",
    "apiVersion": "2018-03-01",
    "recipientAccountId": “111122223333”
}
```

The following example shows a CloudTrail log entry that demonstrates the `UntagResource` action when a tag for a file system is deleted from the console.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": “Root”,
        "principalId": "111122223333",
        "arn": "arn:aws:sts::111122223333:root",
```

```
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-11-14T23:40:54Z"
            }
        }
    },
    "eventTime": "2018-11-14T23:40:54Z",
    "eventSource": "fsx.amazonaws.com",
    "eventName": "UntagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "resourceARN": "arn:aws:fsx:us-east-1:111122223333:file-system/fs-
ab12cd34ef56gh789"
    },
    "responseElements": null,
    "requestID": "aEXAMPLE-abcd-1234-56ef-b4cEXAMPLE51",
    "eventID": "bEXAMPLE-gl12-3f5h-3sh4-ab6EXAMPLE9p",
    "eventType": "AwsApiCall",
    "apiVersion": "2018-03-01",
    "recipientAccountId": "111122223333"
}
```

Amazon FSx for Lustre Maintenance Windows

Amazon FSx for Lustre performs routine software patching for the Lustre software it manages. The maintenance window is your opportunity to control what day and time of the week this software patching occurs.

Patching occurs infrequently, typically once every several weeks. Patching should require only a fraction of your 30-minute maintenance window. During these few minutes of time, your file system will be temporarily unavailable.

You choose the maintenance window during file system creation. If you have no time preference, then a 30-minute default window is assigned.

Changing the Maintenance Window of an Existing File System

You can use the AWS CLI or one of the AWS SDKs to change the maintenance window for your file systems.

To update the weekly maintenance window of your file system

1. Open a command prompt or terminal on your computer.
2. Run the following command, replacing the file system ID with the ID for your file system, and the date and time with when you want to begin the window.

```
aws fsx update-file-system --file-system-id fs-01234567890123456 --lustre-configuration
 WeeklyMaintenanceStartTime=1:01:30
```

Limits

Following, you can find out about limits when working with Amazon FSx for Lustre.

Limits That You Can Increase

Following are the limits for Amazon FSx for Lustre per AWS account, per AWS Region, which you can increase by contacting AWS Support.

Resource	Default limit	Can be increased up to
Number of file systems	100	Thousands
Total storage for all file systems	100,800 GiB	Petabytes

To request a limit increase

1. Open the AWS Support Center page, sign in, if necessary, and then choose **Create Case**.
2. Under **Regarding**, choose **Service Limit Increase**.
3. Under **Limit Type**, choose the type of limit to increase, fill in the necessary fields in the form, and then choose your preferred method of contact.

Resource Limits for Each File System

Following are the limits on Amazon FSx for Lustre resources for each file system in an AWS Region.

Resource	Limit per file system
Number of tags	50

For information on throughput capacity, see Amazon FSx for Lustre Performance (p. 15).

Troubleshooting

File System Mount Hangs and Then Fails with Timeout Error

The file system mount command hangs for a minute or two, and then fails with a timeout error.

The following code shows an example.

```
sudo mount -t lustre file_system_dns_name@tcp:/fsx /mnt/fsx

[2+ minute wait here]
Connection timed out
```

This error can occur because the security groups for the Amazon EC2 instance or the file system aren't configured properly.

Action to Take

Make sure that your security groups for the file system have the inbound rules specified in Amazon VPC Security Groups (p. 29).

Automatic Mounting Fails and the Instance Is Unresponsive

In some cases, automatic mounting might fail for a file system and your Amazon EC2 instance might stop responding.

This issue can occur if the `_netdev` option wasn't declared. If `_netdev` is missing, your Amazon EC2 instance can stop responding. This result is because network file systems need to be initialized after the compute instance starts its networking.

Action to Take

If this issue occurs, contact AWS Support.

File System Mount Using DNS Name Fails

A file system mount that is using a Domain Name Service (DNS) name fails. The following code shows an example.

```
sudo mount -t lustre file_system_dns_name@tcp:/fsx /mnt/fsx
mount.lustre: Can't parse NID
'file_system_dns_name@tcp:/fsx'
```

Action to Take

Check your virtual private cloud (VPC) configuration. If you are using a custom VPC, make sure that DNS settings are enabled. For more information, see Using DNS with Your VPC in the *Amazon VPC User Guide*.

To specify a DNS name in the `mount` command, do the following:

- Ensure that the Amazon EC2 instance is in the same VPC as your Amazon FSx for Lustre file system.
- Connect your Amazon EC2 instance inside a VPC configured to use the DNS server provided by Amazon. For more information, see DHCP Options Sets in the *Amazon VPC User Guide*.
- Ensure that the Amazon VPC of the connecting Amazon EC2 instance has DNS host names enabled. For more information, see Updating DNS Support for Your VPC in the *Amazon VPC User Guide*.

Creating a File System with Data Repository Fails

You can't create a file system linked to a data repository in Amazon S3 bucket, and encounter an error like the following.

```
User: arn:aws:iam::012345678901:user/username is not authorized to perform:
 iam:PutRolePolicy on resource: resource ARN
```

This error can happen if you try to create a file system linked to a data repository in an Amazon S3 bucket without the necessary IAM permissions. The required IAM permissions support the Amazon FSx for Lustre service-linked role that is used to access the specified Amazon S3 bucket on your behalf.

Action to Take

Ensure that your IAM entity (user, group, or role) has the appropriate permissions to create file systems. Doing this includes adding the permissions policy that supports the Amazon FSx for Lustre service-linked role. For more information, see Adding Permissions to Use Data Repositories in Amazon S3 (p. 4).

For more information about service-linked roles, see Using Service-Linked Roles for Amazon FSx for Lustre (p. 30).

Document History

- **API version:**
- **Latest documentation update:** March 11, 2019

The following table describes important changes to the *Amazon FSx for Lustre User Guide*. For notifications about documentation updates, you can subscribe to the RSS feed.

update-history-change	update-history-description	update-history-date
Lustre support for Amazon Linux and Amazon Linux 2 added (p. 42)	The Amazon FSx for Lustre client now supports Amazon EC2 instances running Amazon Linux and Amazon Linux 2. For more information see Installing the Lustre Client.	March 11, 2019
User-defined data export path support added (p. 42)	Users now have the option to overwrite the original objects in your Amazon S3 bucket or write the new or changed files to a prefix that you specify. With this option, you have additional flexibility to incorporate Amazon FSx for Lustre into your data processing workflows. For more information, see Exporting Data to Your Amazon S3 Bucket.	February 6, 2019
Total storage default limit increased (p. 42)	The default total storage for all Amazon FSx for Lustre file systems increased to 100,800 GiB. For more information, see Limits.	January 11, 2019
Amazon FSx for Lustre is now generally available (p. 42)	Amazon FSx for Lustre is a fully managed file system that is optimized for compute-intensive workloads, such as high-performance computing, machine learning, and media processing workflows.	November 28, 2018