

---

# Simple File Manager for Amazon EFS **Implementation Guide**

---

## **Simple File Manager for Amazon EFS: Implementation Guide**

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

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

## Table of Contents

|   |    |
|---|----|
| Welcome .....   | 1  |
| Cost .....  | 2  |
| Architecture overview .....                           | 3  |
| Solution components .....                             | 4  |
| Web UI .....  | 4  |
| Security .....  | 5  |
| API Security .....                                    | 5  |
| IAM roles .....                                       | 5  |
| Amazon CloudFront .....                               | 5  |
| Design considerations .....                           | 6  |
| Limitations .....                                     | 6  |
| Regional deployments .....                            | 6  |
| AWS CloudFormation templates .....                    | 7  |
| Automated deployment .....                            | 8  |
| Deployment overview .....                             | 8  |
| Step 1. Launch the stack .....                        | 8  |
| Step 2. Reset auto-generated password .....           | 9  |
| Step 3. Create a file manager Lambda function .....   | 9  |
| Step 4. Create Amazon Cognito users .....             | 10 |
| Using the solution .....                              | 12 |
| Additional resources .....                            | 13 |
| Update the stack .....                                | 14 |
| Update post deployment resources .....                | 14 |
| Uninstall the solution .....                          | 15 |
| Delete file manager Lambda and EFS access point ..... | 15 |
| Using the AWS Management Console .....                | 15 |
| Using AWS Command Line Interface .....                | 15 |
| Source code .....                                     | 16 |
| Revisions .....                                       | 17 |
| Contributors .....                                    | 18 |
| Notices .....   | 19 |

# Deploy a web user interface to manage your Amazon EFS file systems

Publication date: *July 2021*

Simple File Manager for Amazon EFS helps you to directly interact with data in your [Amazon Elastic File System](#) (Amazon EFS) file systems without deploying an [Amazon Elastic Compute Cloud](#) (Amazon EC2) instance. It features a web user interface (web UI) where you can browse, upload, and download files in existing EFS file systems, without specialized Linux knowledge. For example, you can use this solution to upload a machine learning (ML) model file to an Amazon EFS file system that is used by an AWS Lambda function for ML inference.

This guide provides infrastructure and configuration information for planning and deploying the Simple File Manager for Amazon EFS in the Amazon Web Services (AWS) Cloud. It includes links to an [AWS CloudFormation](#) template that launches and configures the AWS services required to deploy this solution using AWS best practices for security and availability.

The guide is intended for IT architects, developers, DevOps, data analysts, and marketing technology professionals who have practical experience architecting in the AWS Cloud.

# Cost

You are responsible for the cost of the AWS services used while running the Simple File Manager for Amazon EFS solution, which can vary based on the following factors:

- Number of Amazon API Gateway requests per month.
- Number of AWS Lambda invocations per month.
- Volume of web traffic delivered from Amazon CloudFront per month. (The solution uses CloudFront to deliver the web UI with ultra-low latency performance and high availability to your end users.)
- Number of active users per month authenticated with Amazon Cognito.

As of July 2021, the cost for running this solution with the default settings in the US East (N. Virginia) Region is approximately **\$0.78/month**. This solution is based entirely on serverless AWS services. Therefore, when the solution is not in use, you only pay for data storage.

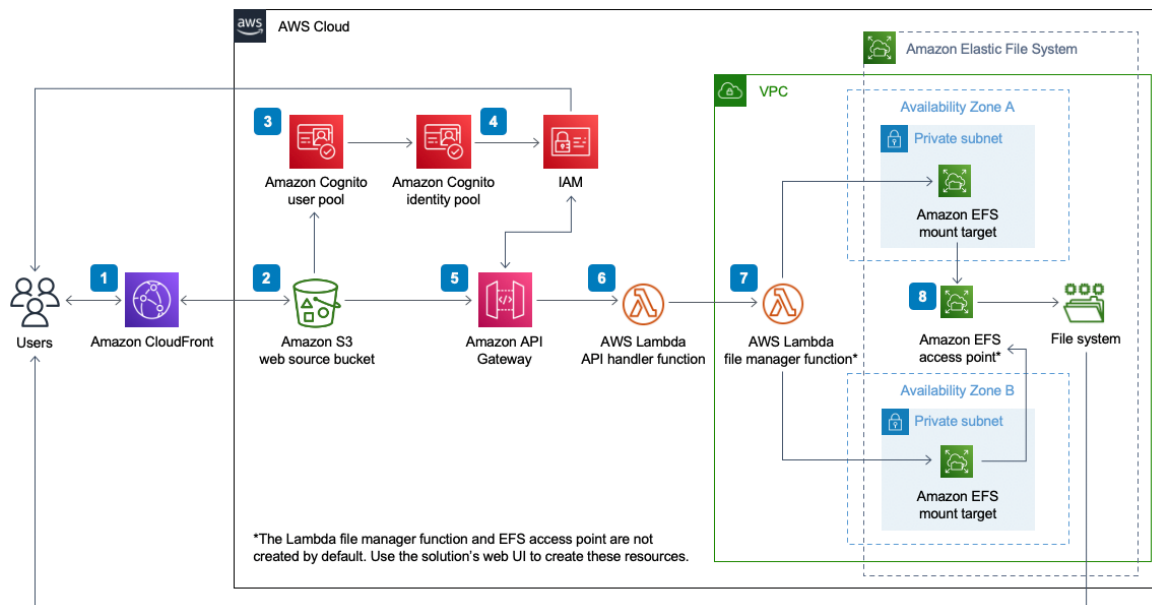
We recommend creating a [budget](#) through [AWS Cost Explorer](#) to help manage costs.

| AWS service        | Dimensions   | Monthly cost                         |
|--------------------|--|--------------------------------------|
| Amazon EFS         | Standard storage/GB-month  | \$0.30                               |
| Amazon API Gateway | 50,000 requests/month  | \$0.17                               |
| AWS Lambda         | 100,000 invocations/month (avg 300 ms duration and 128 MB memory)  | \$0.08                               |
| Amazon CloudFront  | Regional data transfer out to internet: first 10 TB<br><br>Regional data transfer out to origin: all data transfer<br><br>HTTPS Requests: 50,000 requests/month X (\$0.01/10,000 requests) | \$0.085<br><br>\$0.020<br><br>\$0.05 |
| Amazon S3          | Storage (0.04 GB) and 50,000 get requests/month  | \$0.02                               |
| Amazon Cognito     | 10 user x (\$0.0055/monthly active users (MAUs))   | \$0.055                              |
| <b>Total:</b>      |  | <b>\$0.78</b>                        |

This cost estimate does not account for resources related to [Amazon VPC](#) that the solution interacts with and for which you may incur additional charges. Prices are subject to change. For full details, refer to the pricing webpage for each AWS service used in this solution.

# Architecture overview

Deploying this solution with the default parameters builds the following environment in the AWS Cloud.



**Figure 1: Simple File Manager for Amazon EFS architecture on AWS**

The AWS CloudFormation template deploys the following infrastructure:

1. An [Amazon CloudFront](#) distribution to serve the Simple File Manager web UI.
2. An [Amazon Simple Storage Service](#) (Amazon S3) web source bucket for hosting the CloudFront distribution.
3. An [Amazon Cognito user pool](#) to provide a user directory.
4. An Amazon Cognito [identity pool](#) to provide federation with [AWS Identity and Access Management](#) (IAM) for authentication and authorization to the web UI.
5. An [Amazon API Gateway](#) file manager REST API to proxy file system operations from the web UI to your Amazon EFS file system. AWS IAM roles are created for the API to operate.
6. An [AWS Lambda](#) API handler function to support the file manager API.
7. An AWS Lambda file manager function to connect to the Amazon EFS file system.
8. An [Amazon EFS](#) access point to allow Amazon EFS file system access from AWS Lambda.

**Note**

The AWS Lambda file manager function (7) and Amazon EFS access point (8) are not automatically deployed by this solution's CloudFormation template. [Create these resources after deployment](#) (p. 9) with the solution's web UI.

# Solution components

## Web UI

The solution provides a web UI that you can use to interact with your EFS file systems. It is designed to allow you to create AWS Lambda file manager functions that connect to your EFS file system. The file manager API sends file system operations to this Lambda function. Currently, the web UI supports the following file operations:

- Create a directory
- List files
- Delete files
- Upload files
- Download files

# Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This [shared model](#) reduces your operational burden because AWS operates, manages, and controls the components including the host operating system, the virtualization layer, and the physical security of the facilities in which the services operate. For more information about AWS security, visit [AWS Cloud Security](#).

## API Security

The solution's Amazon API Gateway file manager REST API is secured with an [IAM authorizer](#). Valid AWS Identity and Access Management (IAM) credentials are granted to a user via AWS Security Token Service (AWS STS) after they successfully authenticate with the solution-deployed Amazon Cognito user pool and if they belong to an Amazon Cognito group that has an adequately scoped IAM role associated with it. Refer to the [sequence diagram on GitHub](#) for further details.

## IAM roles

IAM roles allow customers to assign granular access policies and permissions to services and users on the AWS Cloud. This solution creates IAM roles that grant the solution's AWS Lambda functions access to create Regional resources. The solution also creates the `EFSFileManagerIamRole` IAM role that has Amazon API Gateway `execute-api` permissions on all file manager API endpoints.

## Amazon CloudFront

This solution deploys a web UI [hosted](#) in an Amazon Simple Storage Service (Amazon S3) bucket. To help reduce latency and improve security, this solution includes an Amazon CloudFront distribution with an origin access identity, which is a CloudFront user that provides public access to the solution's website bucket contents. For more information, refer to [Restricting Access to Amazon S3 Content by Using an Origin Access Identity](#) in the *Amazon CloudFront Developer Guide*.



# Design considerations

When you create AWS Lambda file manager functions, you must ensure that the default settings fit your use case and security requirements. By allowing the default settings, the users created in the Simple File Manager for Amazon EFS Amazon Cognito user pool will have full access to files present in the directory specified.

## Limitations

As of July 2021, uploads and downloads with this solution have been tested with files up to 1 GB in size. This solution supports larger file uploads, but anything larger than 1 GB has not been tested.

## Regional deployments

This solution uses Amazon Cognito, which is not currently available in all AWS Regions. You must launch this solution in a Region where Amazon Cognito is available. For the most current availability by Region, refer to the [AWS Regional Services List](#).

# AWS CloudFormation templates

To automate deployment, this solution uses AWS CloudFormation. It includes the following CloudFormation template, which you can download before deployment:

[View  
Template](#)

**simple-file-manager-for-amazon-efs.template:** Creates the solution's API Gateway REST API and instantiates the nested templates for the application. Use this template to launch the solution and all associated components. The default configuration deploys Amazon API Gateway, AWS Lambda, Amazon S3, Amazon CloudFront, and Amazon Cognito, but you can customize the template to meet your specific needs.

The following nested templates are automatically deployed:

**efs-file-manager-web.yaml:** This nested template creates the web UI resources: Amazon CloudFront distribution, Amazon S3 hosting bucket, AWS IAM roles, and AWS CloudFormation custom resources to support automated deployment.

**efs-file-manager-auth.yaml:** This nested template creates the authentication and authorization resources: Amazon Cognito User Pool, Amazon Cognito identity pool, Amazon Cognito Application Client, AWS IAM role for application access, and an AWS CloudFormation custom resource to perform Amazon Cognito Role Mapping.

The following template is integrated with the solution's web UI to create a file manager Lambda function, which provides the solution access to a file system:

**efs-file-manager-ap-lambda.template:** Creates a file manager AWS Lambda function and Amazon EFS access point for a specified Amazon EFS file system. The template does not require customizations; however, you can download this template from the [Simple File Manager for Amazon EFS GitHub repository](#) to customize or repurpose the template for your own needs.

# Automated deployment

Before you launch the solution, review the architecture, components, network security, and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy the solution into your account.

**Time to deploy:** Approximately 15 minutes.

## Deployment overview

Use the following steps to deploy this solution on AWS. For detailed instructions, follow the links for each step.

[Step 1. Launch the stack \(p. 8\)](#)

- Launch the AWS CloudFormation template into your AWS account.
- Review the template's parameters and enter or adjust the default values as needed.

[Step 2. Reset auto-generated password \(p. 9\)](#)

[Step 3. Create a file manager Lambda function \(p. 9\)](#)

- Use AWS Lambda to create a file manager function for your EFS file system.

[Step 4. Create Amazon Cognito users \(p. 10\)](#)

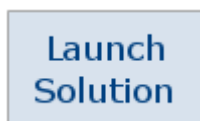
## Step 1. Launch the stack

This automated AWS CloudFormation template deploys Simple File Manager for Amazon EFS in the AWS Cloud.

### Note

You are responsible for the cost of the AWS services used while running this solution. For more details, visit the [Cost \(p. 2\)](#) section in this guide, and refer to the pricing webpage for each AWS service used in this solution.

1. Sign in to the AWS Management Console and select the button to launch the `simple-file-manager-for-amazon-efs.template` AWS CloudFormation template.



Alternatively, you can [download the template](#) as a starting point for your own implementation.

2. The template launches in the US East (N. Virginia) Region by default. To launch the solution in a different AWS Region, use the Region selector in the console navigation bar.

### Note

This solution uses Amazon Cognito, which is not currently available in all AWS Regions. You must launch this solution in an AWS Region where Amazon Cognito is available. For the most current availability by Region, refer to the [AWS Regional Services List](#).

3. On the **Create stack** page, verify that the correct template URL is in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, refer to [IAM and STS Limits](#) in the *AWS Identity and Access Management User Guide*.
5. Under **Parameters**, review the parameters for this solution template and modify them as necessary. This solution uses the following default values.

| Parameter         | Default          | Description  |
|-------------------|------------------|--|
| <b>AdminEmail</b> | <Requires input> | The email address of the user that will use the solution. This user will have access to the files and file systems within the AWS account. |

6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review** page, review and confirm the settings. Check the boxes acknowledging that the template will create AWS Identity and Access Management (IAM) resources and requires certain capabilities.
9. Choose **Create stack** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation Console in the **Status** column. You should receive a CREATE\_COMPLETE status in approximately 15 minutes.

## Step 2. Reset auto-generated password

During stack creation, the solution sends you an email containing your initial login credentials.

1. Sign in to the [AWS CloudFormation console](#).
2. Select the solution's root stack.
3. Select the **Outputs** tab and choose **EFSFileSimpleWebsiteUrl** in the **Value** column.
4. To sign in, use the initial credentials sent to the admin email.
5. Follow the prompts to create a new password.

Upon successful authentication, the application routes you to the web UI home page, where all the EFS file systems are displayed in your account for the Region where the stack was deployed.

6. For convenience, save the address.

## Step 3. Create a file manager Lambda function

To grant Simple File Manager for Amazon EFS access to a file system, create a file manager Lambda function.

1. Log in to the solution's web UI. For the site URL, refer to [Step 2. Reset auto-generated password \(p. 9\)](#).
2. Under **Filesystems**, select the **false** link.
3. On the **Create file manager lambda** page, enter the following information:

- **User ID** - The numeric POSIX user ID that Lambda will use to make file system requests.
- **Group ID** - The numeric POSIX group ID that Lambda will use to make file system requests.
- **Path** - The file system directory that the solution will use as the root directory. Any files present in this directory are accessible to the application.

If you are unsure what the options are, the default values work for most Simple File Manager for Amazon EFS use cases.

4. Choose **Submit** and wait for the application to complete the request.
5. After completion, you are routed back to the web UI's home page.

**Note**

Lambda can take several minutes to provision a new function. Allow 1-2 minutes if the managed state returns **Creating** and refresh the page.

6. The link previously labeled **false** now returns **true** and the file system ID is now an active link.
7. Select the file system ID link to access the file system.

## Step 4. Create Amazon Cognito users

This solution uses Amazon Cognito to manage all users and authentication. It creates a user for you during deployment and sends an email at the address provided with temporary credentials.

Use the following procedure to create additional users:

1. Sign in to the [AWS Cognito console](#).
2. Choose **Manage User Pools**.
3. Choose **SimpleFileManagerUserPool**.
4. In the navigation pane, under **General Settings**, choose **Users and groups**.
5. From the **Users** tab, choose **Create user**.
6. In the **Create user** box, enter values for all required fields.

| Form field                | Required?        | Description   |
|---------------------------|------------------|---|
| <b>Username</b>           | Yes              | The user name that you will use to log in to Simple File Manager for Amazon EFS. For convenience, it is best to match this value to the Email form field.                                     |
| <b>Send an invitation</b> | Yes (email only) | When selected, sends a notification as a reminder of the temporary password. Select <b>Email only</b> . If you select SMS (default) an error message displays, but the user is still created. |
| <b>Temporary Password</b> | Yes              | Enter a temporary password. The user is forced to change this when they log in to Simple File Manager for Amazon EFS for the first time.  |

Simple File Manager for Amazon  
EFS Implementation Guide  
Step 4. Create Amazon Cognito users

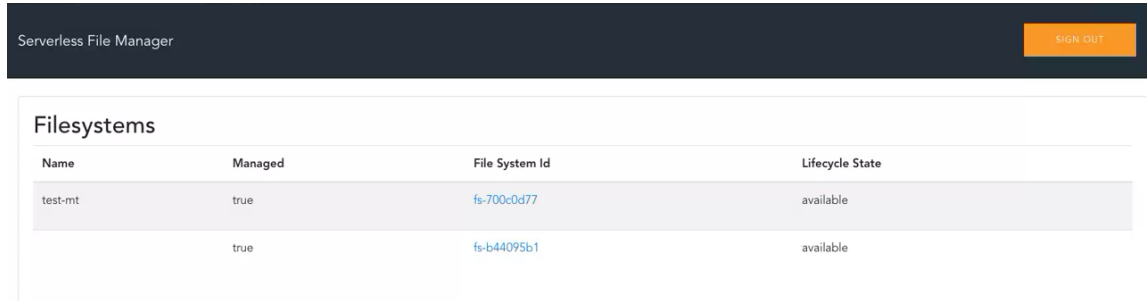
---

| Form field   | Required? | Description  |
|--------------|-----------|--|
| Phone Number | No        | Enter a phone number in international format, for example, +44. Ensure that the <b>Mark phone number as verified?</b> box is selected. |
| Email        | Yes       | Enter a valid email address. Ensure that the <b>Mark email as verified?</b> box is selected.   |

7. Choose **Create user**.
8. Choose **Groups**.
9. Choose the group with the description **User group for Simple File Manager Admins**.
10. Choose **Add users**.
11. Choose the **plus icon** next to the user name of the user you just created.
12. Repeat this process to create as many users as you need.

# Using the solution

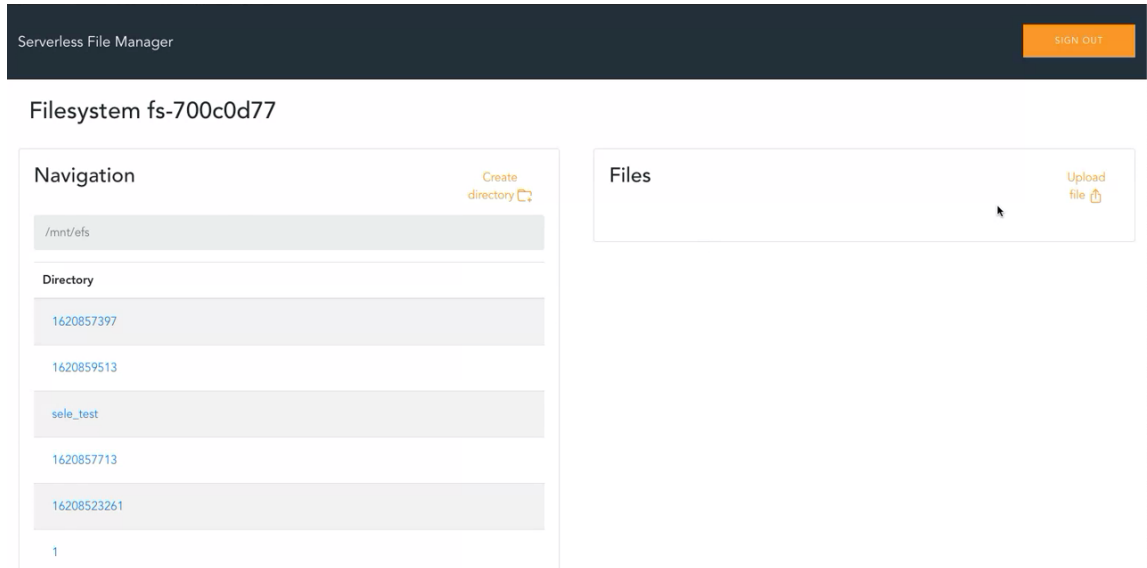
1. Log in to the solution's web UI. For the site URL, refer to [Step 2. Reset auto-generated password \(p. 9\)](#).
2. Under **Filesystems**, select the file system ID link to access the file system.



**Figure 3: Web UI home page**

3. Choose the relevant button or directory to perform the relevant operation:

- Create a directory
- List files
- Delete files
- Upload files
- Download files



**Figure 4: Web UI within selected file system**

4. To return to the home page, choose **Serverless File Manager**.

# Additional resources

## **AWS services**

- [AWS CloudFormation](#)
- [Amazon Elastic File System](#)
- [Amazon Route 53](#)
- [AWS Lambda](#)
- [Amazon API Gateway](#)
- [Amazon Cognito](#)

## **Related AWS documentation**

- [Using Amazon EFS with Lambda](#)
- [Amazon Cognito user pools](#)
- [Amazon Cognito identity pools](#)



# Update the stack

If you have previously deployed this solution, follow this procedure to update your AWS CloudFormation stack to the latest version.

1. Sign in to the [AWS CloudFormation console](#), select your existing **SimpleFileManager** CloudFormation stack, and choose **Update**.
2. Select **Replace current template**.
3. Under **Specify template**:
  - a. Select **Amazon S3 URL**.
  - b. Copy the link of the [latest template](#).
  - c. Paste the link in the **Amazon S3 URL** box.
  - d. Verify that the correct template URL shows in the **Amazon S3 URL** text box, and choose **Next**. Choose **Next** again.
4. Under **Parameters**, review the parameters for the template and modify them as necessary.
5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template might create AWS Identity and Access Management (IAM) resources.
8. Choose **View change set** and verify the changes.
9. Choose **Update stack** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of UPDATE\_COMPLETE in approximately 15 minutes.

## Update post deployment resources

For updates that include changes to post deployment resources, which are outlined in the release notes, you must follow these additional steps to update the solution. Complete the standard [update process \(p. 14\)](#) before proceeding with these steps.

1. Log in to the solution's web UI.
2. Choose the **true** link for a file system.
3. Choose **Delete**.
4. Wait for the **deleting** status to change to **false**.
5. Recreate the file manager Lambda function. For details, refer to [Step 3. Create a file manager Lambda function \(p. 9\)](#).
6. Repeat this process for each managed file system in the **Filesystems** table.

# Uninstall the solution

You can uninstall the Simple File Manager for Amazon EFS solution from the AWS Management Console or by using the AWS Command Line Interface. You must manually delete the file manager Lambda and EFS access point created by this solution before you delete the CloudFormation stack.

## Delete file manager Lambda and EFS access point

1. Log in to the solution's web UI.
2. Choose the **true** link for a filesystem.
3. Choose **Delete**.
4. Wait for the **deleting** status to change to **false**.
5. Repeat this process for each managed file system in the **Filesystems** table.

## Using the AWS Management Console

1. Sign in to the [AWS CloudFormation console](#).
2. Select this solution's installation stack.
3. Choose **Delete**.

## Using AWS Command Line Interface

Determine whether the AWS Command Line Interface (AWS CLI) is available in your environment. For installation instructions, refer to [What Is the AWS Command Line Interface](#) in the *AWS CLI User Guide*. After confirming that the AWS CLI is available, run the following command.

```
$ aws cloudformation delete-stack --stack-name <installation-stack-name>
```

# Source code

Visit the [Simple File Manager for Amazon EFS GitHub](#) repository to download the source files for this solution and to share your customizations with others. Refer to the [README.md file](#) for additional information.

# Revisions

| Date      | Change                   |
|-----------|--------------------------|
| July 2021 | Initial release (v1.4.0) |

# Contributors

- Brandon Dold
- Eddie Goynes
- Andrea Amorosi

# Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. AWS responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Simple File Manager for Amazon EFS is licensed under the terms of the of the Apache License Version 2.0 available at [The Apache Software Foundation](#).