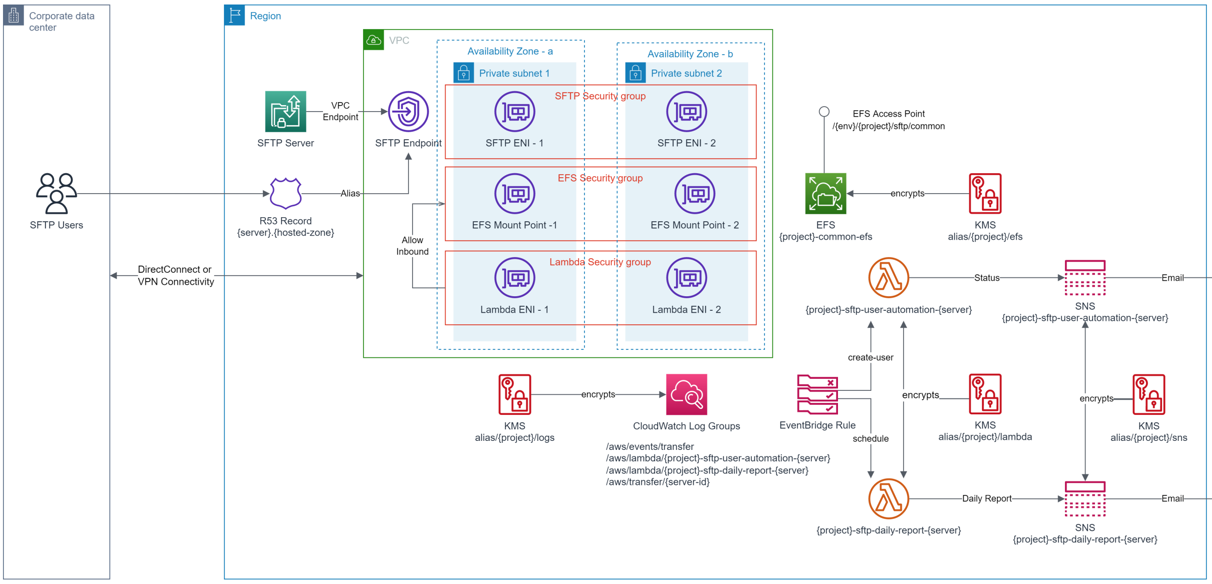
AWS Transfer Family SFTP Server using Amazon EFS domain with automated operations

Large organizations are migrating existing on-premises workloads to the AWS. Many of these workloads are legacy applications that are running on the Amazon EC2 instances. Often these legacy applications exchange files with other applications or trading partners via SFTP. Each SFTP client requires their own secured area for `inbound` and `outbound` files. The legacy application, however, requires access to all the files exchanged with all the SFTP clients.

This solution is a set of Terraform modules and examples. It provisions an AWS Transfer Family SFTP server that uses an Amazon EFS File System via an Amazon EFS Access Point as the storage backend. It also provisions automation for the SFTP client folder maintenance on the Amazon EFS File System. Optionally, the solution can send SFTP client folder status email and SFTP activity email to different subscribers.

The Amazon EFS File System can be mounted on the Amazon EC2 instance(s) hosting the target application(s) needing access to the `inbound` and `outbound` files.



Features and Benefits

The solution has following features and benefits:

- Uses existing VPC and Subnets, identified via tags, for the SFTP server and the EFS mount points.
- Creates a VPC hosted internal SFTP Server with service managed identity provider.
  - Optionally creates a DNS record for the SFTP server via providing the Route 53 hosted zone name.
- Manages SFTP clients along with POSIX profile and public `ssh` key in Terraform state.
- Automatically creates the SFTP client folders on the target Amazon EFS access point with given POSIX profile.
  - Manages subscription to the email for each folder creation status.
- Uses an existing EFS and/or EFS access point or provision new EFS and/or EFS access point.
- Uses an existing IAM roles for the SFTP Users, Lambda execution, and CloudWatch Logging or provision new IAM Roles.

- Uses an existing customer managed keys (CMKs) to optionally encrypt the EFS, CloudWatch Logs, Lambda, and SNS or provision new CMKs.
- Uses an existing Security Groups for the SFTP server, EFS, and Lambda or provision new Security Groups.
- Enables SFTP activity report via email on a given schedule.
    - Manages subscription to the SFTP activity report email.
- Uniformly names and tags the provisioned resources.

Prerequisites

- The target AWS Account and AWS Region are identified.

- The AWS User/Role executing the Terraform scripts must have permissions to provision the target resources.

- The Terraform CLI (`version = ">= 1.0.4"`) is installed.

- Terraform backend provider and state locking providers are identified and bootstrapped.

    - An example bootstrap module/example is provided that provisions an Amazon S3 bucket for Terraform state storage and Amazon DynamoDB table for Terraform state locking.
        - The Amazon S3 bucket name has to be globally unique.

- The target VPC along with the target Subnets exist and identified via tags.

    - The examples use the following tags to identify the target VPC and Subnets.

      ```
      "transfer/sftp/efs" = "1"
      "Env"               = "DEV"
      ```

- A unique project code name e.g. `appx-sftp` is identified that will be used in naming all the resources.

- Uniform resource tagging scheme is identified.

    - The examples use only two tags: `Env` and `Project`

- Optionally, Route 53 Hosted zone is identified.

Usage

- Use the module via GitHub source or copy the module into your repository.
- Incorporate the module in your CI/CD pipeline as appropriate.

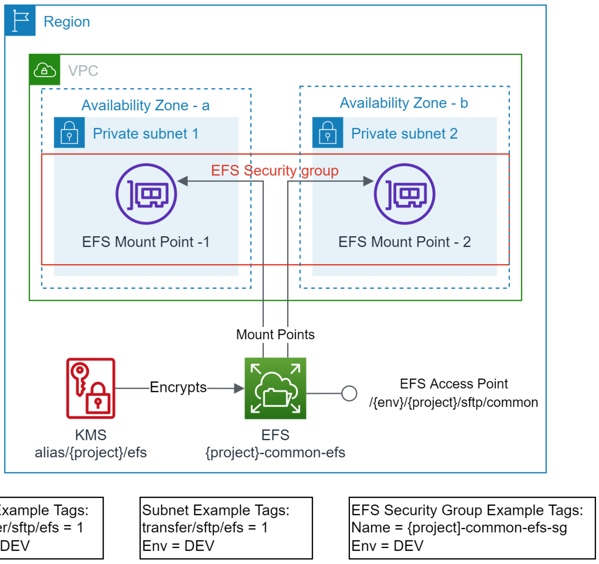
Scenarios

This solution primarily supports the following scenarios though many other scenarios are possible.

Scenario 1: Shared EFS and Shared EFS Access Point

- EFS file system exists and optionally encrypted using KMS.

- EFS access point exists.
- EFS mount points exist in the target VPC Subnets.
- EFS Security Group exists and attached to the EFS mount points.

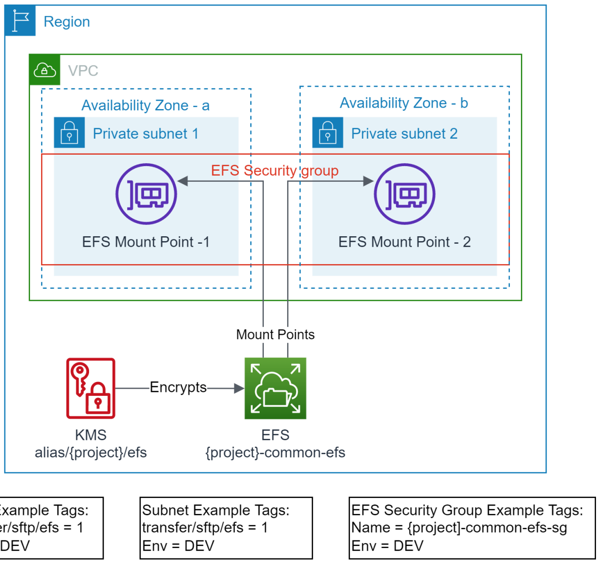


Use examples/efs/scenario1 to setup this scenario.

Use examples/sftp/scenario1 to execute this scenario.

Scenario 2: Shared EFS and Owned EFS Access Point

- EFS file system exists and optionally encrypted using KMS.
- EFS access point does not exist. It is owned by the SFTP server.
- EFS mount points exist in the target VPC Subnets.
- EFS Security Group exists and attached to the EFS mount points.

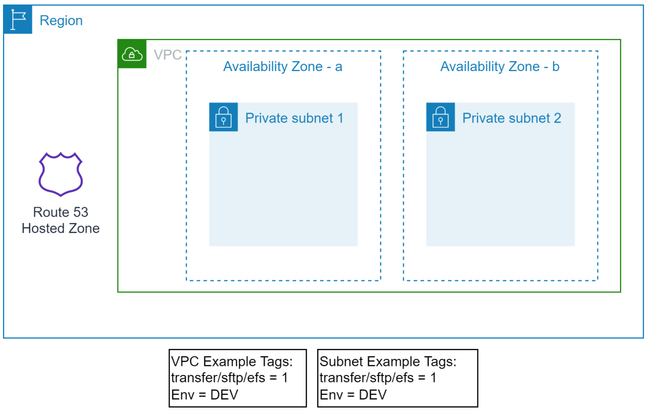


Use examples/efs/scenario2 or examples/efs/scenario1 to setup this scenario.

Use examples/sftp/scenario2 to execute this scenario.

Scenario 3: Owned EFS and Owned EFS Access Point

- EFS file system does not exist. It is owned by the SFTP server.
- EFS access point does not exist. It is owned by the SFTP server.
- EFS mount points do not exist. It will be created along with the EFS.
- EFS Security Group does not exist. It will be created along with the EFS.



Use examples/sftp/scenario3 to execute this scenario.

Future Enhancements

- The current solution only takes actions for the `CreateUser` event. The solution can be enhanced to support `UpdateUser` and `DeleteUser` events.
- The solution can be enhanced to support the VPC hosted `internet-facing` SFTP Server using Elastic IP addresses.
- The solution be can be enhanced to support other identity providers such as AWS Directory Service and Custom Identity Provider.
- The current solution proposes one SFTP server per application. The solution can be enhanced to support single SFTP server for multiple applications that would allow a single SFTP client to securely exchange files with multiple applications.
- The current solution sends basic email alert and report. It can be enhanced to send better looking emails.
- The current solution assumes a single AZ in a single region. It could leverage multiple AZs and / or use EFS replication to provide multi-region availability.
- The current solution does not employ any form of backup. EFS has the ability to do automatic backups and this could be leveraged.
- EFS performance and throughput should be configured based on desired behavior from regular usage.

Security

See CONTRIBUTING for more information.

License