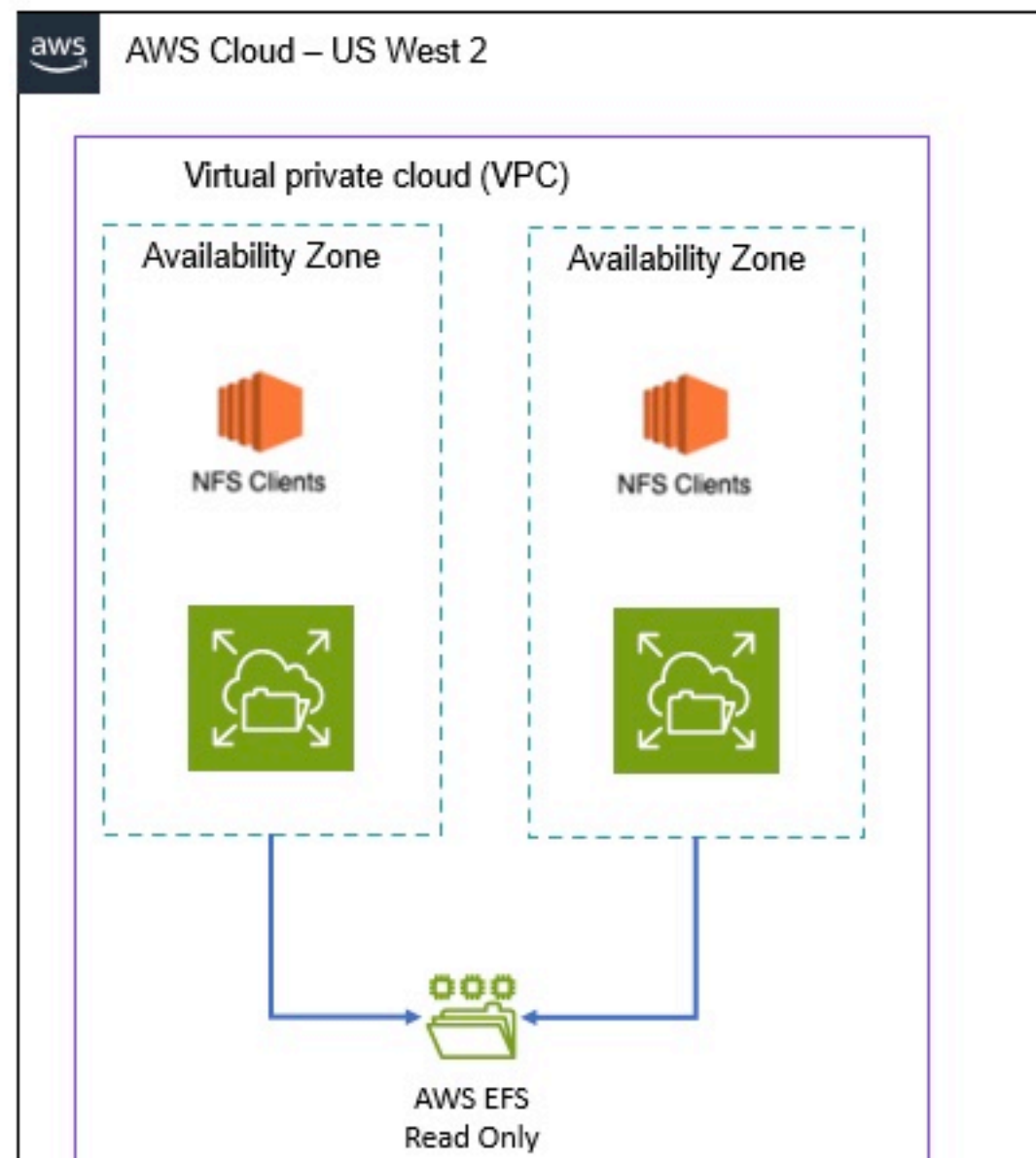
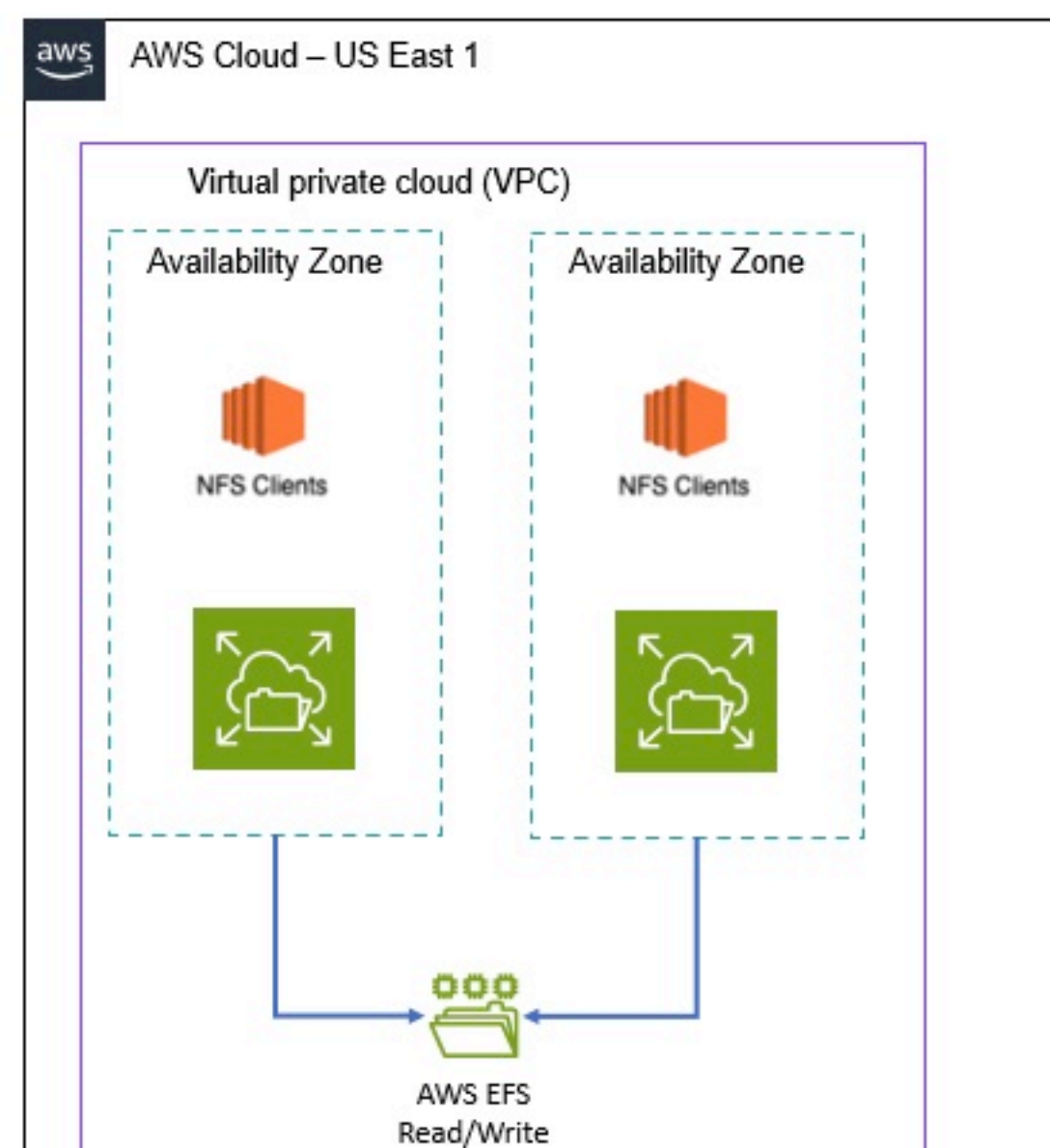


With replication, you can automatically maintain copies of your EFS file systems for continuity or to help you to meet compliance requirements as part of DR strategy. You can set up replication either in the same AWS region as the primary file system, or in a different region, known as [cross-region replication \(CRR\)](#).

## How it works

1. Create a file system in the primary region where you want your file system saved.
2. Enable replication on the file system, which will create a read only version of the primary file system in a different (secondary) region.
3. Create a mount target and access point in the secondary region.
4. The secondary file system will sync with the primary file system, making sure it is an up-to-date copy of the primary.

The following is a diagram showing a basic CRR configuration.





# EFS cross-region replication

Create copies of your file systems in different regions

Contact ▾

 Submit Edit

Cross-region replication (CRR) allows you to create replicas of EFS file systems in different regions. The following guide will show you how to create a file system in the primary region and create a read-only replica file system in a secondary region that syncs with the primary file system.

## Before you begin

- Complete the [onboarding process for EFS](#) and other AWS public cloud offerings.
- Read the [Terraform setup Tech Primer](#) to learn the basics of logging into Terraform and building a workspace.
- Make sure you are using the latest version of every EFS module referenced in this guide by visiting [go/tfe](#).

## Configure CRR

### Create primary file system

In this section you will create an EFS file system in the primary region (in this example, us-east-1) and enable replication to the secondary region (us-east-2).

1. [Create an EFS file system](#) in the main.tf of your primary workspace. The EFS module section of the main.tf file should look like this:

```
module "efs" {
```

```
}
```



In addition to creating the EFS file system with replication enabled, it creates a [customer-managed security group](#) to enable the traffic between EC2 instance and EFS file system on port 2049.

The following inputs need to be configured to enable cross-region replication:

- `multi_region_replication`: You need to set this input to "true" in order to enable replication for your file system.
- `repl_region`: The region where the replica file system will be created. Note that you can choose to make this the same region as the primary file system.
- `efs_fs_id_for_replica`: The primary EFS file system's ID.

## Create KMS key for secondary file system

This section will create a KMS key for the secondary file system:

1. Copy the code for the KMS key module and insert it into the main.tf file:

```
module "kms" {
```

## Run EC2 launch template

This section will use the EC2 launch template to create an EC2 instance and auto scaling group, as well as a role that tkktk. Learn more about [configuring the launch template for EC2](#).

1. Run the EC2 launch template will mount the file system on the EC2 instance:

```
module "role" {
```

## Create access point and mount target for the secondary file system

In this section you will create an access point and mount target for the replica file system in the secondary region.

Note: These steps must be done in a different Terraform workspace than the one you used to create the primary file system.

1. Copy the code to create a new EFS file system and insert it in the main.tf file. The EFS module should look like this:

```
module "efs" {
```

2. You must configure these inputs yourself:

- `fs_id_of_replica`: This should be the file system ID of the primary file system.
- `create_replication_ap_mt_only`: By setting this to "true" you will create the access point and mount target for the replica file system in the secondary region.



# EFS cross-region replication failover and failback

Configure your replica file systems to automatically restore

Contact ▾

 Submit Edit

After configuring an EFS file system for cross-region replication (CRR), you can manually failover to the replica file system in the secondary region, then switch back to the primary region afterward.

## Before you begin

- Complete the [onboarding process for EFS](#) and other AWS public cloud offerings.
- Read the [Terraform setup Tech Primer](#) to learn the basics of logging into Terraform and building a workspace.
- Make sure you are using the latest version of every EFS module referenced in this guide by visiting [go/tfe](#).
- Configured your EFS file system for CRR, so that now you have a replica file system in another secondary region.

## How it works

This is a brief outline of the failover/failback process:

1. You destroy the file system in the primary region, failing over to the file system in the secondary region.
2. The secondary file system will transition from read only to read/write.
3. A new file system in the primary region is created as a replica of the file system in the secondary region.
4. The secondary file system is deleted, causing failback to the new file system in the primary region.

## Failover initiation

To begin failover, go to the workspace where you created the primary file system, then run `tf1 destroy` on the primary file system. This will delete the Terraform workspace of the primary file system, which deletes the file system along with replication between it and the secondary file system. The secondary file system will then switch to read/write mode, allowing users to access the file system in the secondary region.

# Failback initiation

At this point, the file system in the secondary region is now the sole file system. In this section you will create a new file system in the primary region, enable replication from the secondary region to the primary region.

## Create KMS key in primary region

Insert the KMS module in the main.tf of primary workspace. This will create a new KMS key for the new primary file system. You will use it in the next section of this guide.

```
module "kms" {
```

## Create new primary file system

In this section you will create a new file system in the primary region to replace the one you deleted during failover.

1. Insert this into the main.tf file of your primary workspace:

```
module "efs" {
```

2. Configure the following inputs yourself:

- `efs_fs_id_for_replica`: The secondary file system's ID.
- `kms_key_arn`: The KMS key arn of the KMS key you created earlier in this section.
- `multi_region_replication`: This must be set to true to enable replication in the secondary region.
- `repl_region`: The region of the primary file system.

## Destroy secondary file system

You will now run `tfl destroy` in the secondary Terraform workspace. This will destroy the secondary file system and make the primary file system switch to read/write.

## Post-failback steps

At this point, the file system in the secondary region has been destroyed, leaving the file system in the primary region as the sole one left. CRR will not be enabled on this file system by default; you will need to repeat the steps to [configure replication](#) and create a replica file system in the secondary region.