**Connecting your existing EC2 instances to your Amazon EFS file system**

## Prerequisite

For this tutorial, I'll assume you already have an AWS account. If not please create one.

## Steps we will follow

A. First, we will launch two Amazon Linux 2 EC2 instances with SSH access.  
B. We will then create an EFS file system.  
C. Next, we will SSH into the instances and install the amazon-efs-utils package. We will also create a directory that we will use as the file system mount point.  
D. ✨Bonus step✨ We will use EFS mount helper to automatically remount the EFS file system every time we reboot our instances.

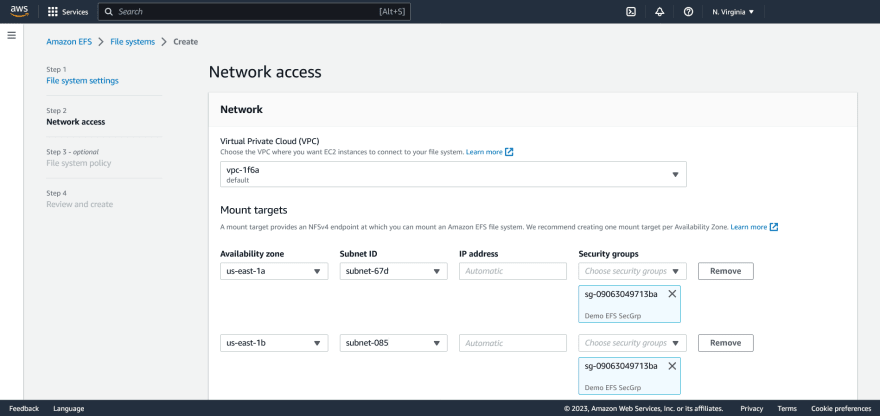
Please note that in-order for this setup to function, all of these resources must be launched in the same region.

Let's get started! 🏁

### A. Launching Amazon Linux 2 EC2 Instance

1. Log into the AWS Management Console.
2. Search for the Amazon EC2 service and open the EC2 console.
3. Firstly, we'll create two security groups. Choose **Security Groups** on left-hand panel.
4. Click **Create Security Group**.
5. Under **Security Group Name** and **Description** enter Demo EC2 SecGrp.
6. Under **VPC** select the default VPC.
7. For the security group inbound rules, create a new rule that allows SSH(Port 22) and set the source to Anywhere IPv4 (0.0.0.0/0).
8. We'll keep the outbound rules as default. Once done, choose **Create security group**.
9. Once the security group has been created, click **Create Security Group** again.
10. Under **Security Group Name** and **Description** enter Demo EFS SecGrp.
11. Under **VPC** select the default VPC.
12. For the security group inbound rules, create a new rule with NFS(Port 2049) and set the source to the Demo EC2 SecGrp.
13. Again, we'll leave the outbound rules as is then choose **Create security group**.
14. On the left hand menu, choose **Instances**.
15. Next, choose **Launch instance**.
16. Under Name, enter the name **"Demo"** to identify your instance.
17. Under Application and OS Images (Amazon Machine Image), choose the **Amazon Linux 2 AMI**.
18. Under Instance type, select **t2.micro**.
19. Under Key pair, for Key pair name, choose an existing key pair or create a new one.
20. Under Network settings, make sure the default VPC is selected. Under security groups, select from existing security groups and choose the Demo EC2 SecGrp that we previously created.
21. We can leave all other configurations as is, and then choose **Launch instance**.

### B. Setting Up Amazon EFS

1. From the search bar type in **"Amazon EFS"** then select the service from the search results.
2. On the Amazon EFS dashboard, choose **Create file system** to open a dialogue box.
3. Enter the Name **"Demo EFS"** for your file system.
4. For Virtual Private Cloud (VPC), choose the same VPC that you launched your instance into.
5. For Storage class, choose the **Standard** option.
6. To customize settings manually, select the **Customize** option.
7. We will stick with most of the default service configurations. When you get to the **Network access** section, clear the **default sg** from all the Availability Zones and add the **EFS SecGrp** instead. Click **Next**.
8. On all other sections leave the default file systems configurations. Then review the settings and select **Create**.
9. Once the file system has been created, click the Demo EFS name entry. You will then be redirected to the **File systems** page. Choose the **Attach** option and on the pop up window that appears, copy the script seen under **"Using the EFS mount helper"** option. It will have the same format as shown below. Save this for later.

sudo mount -t efs -o tls fs-xxxxxx:/ efs

### C. Configuring your Linux instance

1. Open a terminal window and use **ssh** to connect to the instance.
2. ssh -i /path/key-pair-name.pem ec2-user@instance-public-ipv4-dns-name
3. Next, run this command to install the amazon-efs-utils package.
4. sudo yum install -y amazon-efs-utils
5. We now need to create a directory efs that will serve as the file system mount point.
6. sudo mkdir efs
7. Lastly, paste in the command you copied before to mount your file system.
8. sudo mount -t efs -o tls fs-xxxxxx:/ efs

Your EC2 instance is now mounted to the EFS file system. However, you'll notice that if you reboot your instance, the file system does not remount to your instance. We can use the /etc/fstab file with EFS mount helper to automatically remount the file system. The /etc/fstab file contains information about file systems that should be mounted during instance booting.  
If you run the cat /etc/fstab command on your terminal, you'll notice there are no references to the EFS file system we have mounted. We have to manually update the /etc/fstab file so that the instance uses the EFS mount helper to automatically remount the file system whenever the instance restarts.

### D. Setting up Automatic Mounting using /etc/fstab with the EFS Mount Helper

1. Run the following command to open the /etc/fstab file in the nano editor.
2. sudo nano /etc/fstab
3. On a new line, paste in the following (remember to use the file system ID):
4. fs-xxxxxx:/ /home/ec2-user/efs efs tls,\_netdev
5. Save the changes to the file by holding down the ctrl and O keys (press ENTER when prompted to save with the file name) then exit the editor by holding down the ctrl and x keys.
6. Let's test the new entry to see if everything was setup correctly.
7. sudo mount -fav

All done! We now have the EC2 instance remounting the file system on reboot.