Project 9: File system Management

Note: For this homework, mount a disk of 20G your system if you like to use same system for all Part

Part1: Working with MBR Partitions

1. Start a Red Hat 8 (rocky 8) lab server and mount a new disk of 2G. Once completed, log in to the system and navigate to the */dev* directory.
2. Create a primary Linux partition that is 500M in size on the attached disk.
3. Set the partition type for a basic Linux volume.
4. Write changes and exit.
5. Create an XFS filesystem on the disk.
6. Issue the command to list the block device and its UUID.
7. Mount the partition to */mnt/mymount*.
8. Configure the disk to mount to the */mnt/mymount* mount point automatically during system boot.

Part2: Working with GPT Partitions

1. Start a Red Hat 8 (rocky 8) lab server and "mount" a new disk 2G. Once completed, log in to the system and navigate to the */dev* directory.
2. Create a primary Linux partition that is 500M in size on the attached disk and define the partition type during the create process.
3. Write changes and exit.
4. Create an XFS filesystem on the disk.
5. Issue the command to list the block device and its UUID.
6. Mount the partition to */mnt/mymount2*.
7. Configure the disk to mount automatically during system boot to the */mnt/mymount* mount point.

Part3: Creating Persistent Mounts with File System Labels

1. Attach a disk external device of 2G to rhel 8 lab server.
2. Create two partitions, 500M each, on the device.
3. On the "xvdf1" partition, create an XFS file system.
4. On the "xvdf2" partition, create an EXT4 file system.
5. On the "xvdf1" partition, label the filesystem as "filesystem1".
6. On the "xvdf2"partition, label the filesystem as "filesystem2".
7. Create two directories, /mnt/mount1 and /mnt/mount2, to be used as mount points.
8. Create a persistent mount in the*/etc/fstab file* with defaults as the mount option for the xvdf1 partition and use the mount point */mnt/mount1*.
9. Create a persistent mount in the */etc/fstab* file with defaults as the mount option for the xvdf2 partition and use the mount */mnt/mount2*. Set the fsck check to 2 and the dump to 1.

Part4: Creating Swap Space

1. Create a lab server and mount a disk 2G (rhel 8).
2. If the lab servers already have swap enabled, so first run swapoff to disable the */swapfile* in the */etc/fstab* file.
3. Remove the entry */swapfile* from the */etc/fstab* file.
4. Verify there is no current swap space available.
5. Using fdisk, create a partition that uses the entire */dev/xvdf* disk; make sure to label the device correctly.
6. Format the device with the swap signature.
7. Activate the swap space.
8. Add swap space to the */etc/fstab* so that it is a persistent mount.
9. Activate the swap space that is added in the */etc/fstab* file.

Part5: Creating Swap Space from a LVM Volume

1. Start the RHEL 8 lab server and mount a new disk; ensure LVM utilities are installed on the system.
2. Create a partition that uses the entire disk space on */dev/xvdf*.
3. Create a physical volume and a new volume group named "battlestar" out of the */dev/xvdf1* partition.
4. Create a new LVM volume that is approximately 500M in size (generally, you want it to be 2 to 2.5 times the size of your physical memory; however, this is a lab environment).
5. Format the new LVM volume with a swap signature.
6. Add an entry into the */etc/fstab* file using the UUID of the LVM device to create a persistent swap mount.
7. Run the swapon command that enables all swap entries in the */etc/fstab* file.
8. Verify swap space is working.
9. Reboot and test.

Part6: Filesystem ACLs

1. Create a second user on the system called "starbuck".  Open a second terminal window for the lab server, and connect as the user *starbuck*. Ensure you're working as a privileged user (sudo) or root user while performing the lab. The *starbuck* user will be used to test setting the permissions.
2. Navigate into the */tmp* directory, create two new directories named "dir1" and "dir2" and two files called "file1" and "file2".
3. Identity if any of the files currently have extended access control lists associated with them.
4. Set an ACL for the *starbuck* user to read and write for *file1*.
5. Set the mask on the *file1* to read only, then as the *starbuck* user, attempt to execute the following command: echo "test" > /tmp/file1. Why does this fail?
6. Set the default permissions on *dir1* to read, write and execute for the *starbuck* user; execute only if it is a directory.
7. Using setfacl, change the "other" permissions to *none* on *file1*.
8. Remove the default permissions from *dir1.*
9. Remove all ACLs on *file1*.

Part7:

1. Build an advance RHEL8 server with bellow characteristic:

* RAM; 1800 MB
* Disk: 30GB with below partition
* / =10GB
* /boot= 1GB
* /var= 4GB
* /home = 8GB
* User: student
* Password: school1

1. Create a Linux partition that is 500M in size on the attached main disk using fdisk tool with VFAT filesystem and mount it /data/class4 and make it permanent
2. Create a new LVM volume that is approximately 3GB in size (VG Name= devops, LV Name= lv\_linux) using ext3 filesystem and mount it with /data/platform. Make sure it is permanent
3. Extend the current swap if exist with another 1GB using LVM strategy.

Note: All this partitioning needs to be completed using the main disk added when built the server. It means no need to added extra disk to complete these tasks and only using fdisk tool