Installing and Configuring CentOS 8 Stream VM

For my server configuration, I decided to follow the walkthrough posted in the canvas shell and create a PXE configuration to install CentOS Linux onto a machine located on the Linux LAN. First, I created a new virtual machine on the subnet with 2 GB of RAM, and 20 GB of disk space. I also downloaded the CentOS 8 Stream image from the CentOS website and imported it into the VM. Upon entering the installation menu of CentOS, I chose a minimal install and made sure all preliminary settings were correct for the installation. After verification, was able to install CentOS 8 stream onto this box and set the root password. The machine was named 'centos.ia462.com' so it could easily be identified on the network. To make sure the VM was working properly and communicating with the network, I verified the IP address was correctly configured by the DHCP server and pinged various devices around the virtual network. From my host Windows machine, I was able to SSH into this box and input commands by pasting instead of typing them all out individually. In order for the box to be properly configured for our purposes, various packages needed to be installed: realmd net-tools, SSSD, and Samba. After these packages are installed, the box can now be added to the domain using the command "realm join ia462.com". The SSSD config file needed to be changed slightly: use_fully_qualified_names = false, cache_credentials = false, fallback_homedir = /home/%d/%u. Upon configuration, SSSD was enabled and restarted, and the machine was rebooted. At this point, I was able to sign into the AD account I created previously and verify by using the command "id jesse". This account is not a super user, so it needed to be added to /etc/sudoers.d/10_ADROOT (a file I had to create). I copied the contents of the 10_ADROOT file from the class GitHub to this file. Once the contents were copied and the file was saved, I moved to the Windows Server machine to create the "linux-sudo-root" administrator group for the Jesse account to reside in. I rebooted the machine and verified I could sudo from the command line.

Configure Linux for Automated Installation, Configuration, and Testing

The process of actually configuring the PXE server requires the installation of a few packages: HTTPD, TFTP, syslinux-tftpboot.noarch, and tftp-server.x86_64. I then copied the tftp.service file to the systems folder and moved the tftpboot directory to /opt/. Within the tftp.service file, /var/lib/tftpboot was changed to /opt/tftpboot to make sure that the server points to the right directory. The TFTP service was then enabled and restarted. I also allowed it through the firewall by using "firewall-cmd --add-service=tftp –permanent". Inside this directory, I created a new directory for the PXE server to point to called pxelinux.cfg. I copied the default and Centos8 files from the GitHub repo to this folder. The default file remained unchanged, but the Centos8 was altered to point to my repository on the Windows DNS server. I then logged into the Windows Server to set the DHCP scope options for the LinuxNet. I added both options #66 (bootserver) and #67 (bootfile). For the boot server, I made sure to set the hostname to ia462.com. For the boot file, I set the value to lpxelinux.0. I then tested the PXE connection from a test VM on the Linux subnet. The only option that worked at this point wthe install directly from HTTP on the server on campus. I did receive a few issues, but this will be discussed later.

Install and Configure Apache

The final part of this process was to configure the apache server for the PXE box to access and install from the network (not from the EMU server.) First, I created a file at /etc/httpd/conf.d/repo.ia462.com and copied the first part of the code from the class GitHub repo. This file had to be altered to the following specifications: server name = repo.ia462.com, DocumentRoot = /opt/html/ia462/, ServerAdmin = jrusse31@emich.edu, Directory = opt/html/ia462/. I saved this file, and then created the directory for the server to point to. The next phase of this portion was to install semanage and use the command “semanage fcontext -a -t httpd_sys_content_t "/opt/html(/.*)?" to set up SELinux management. Once this was all set up, I changed the C8s-server-ks.cfg file from the GitHub repository and deleted all instances of /centos/ from the URLs to correctly point to the image location. I created a directory called KS and moved the C8s-server-ks.cfg to it. Then I changed the owner to the root.apache profile and the permissions of the directory to read. On the Windows server, I was able to create a repo for the PXE server to boot from.

Issues

The first issue I ran into prevented me from proceeding with the rest of the installation, as the CentOS box could not receive an IP address from the DHCP server residing on the Windows subnet. After much trial and error, I found that the problem lied within the PFSense configuration. I had not configured the Linux subnet on the PFSense Linux interface, so this network was essentially invisible to the others. This PFSense box also needed the BIND server enabled, VM tools installed, and DHCP Relay enabled for the Linux subnet. I was able to receive an IP address from the DHCP after I accomplished these tasks. Another issue I ran into was when logging into the AD server from the CentOS box. I had found a typo in the SSSD configuration file that prevented me from accessing the server. After this was corrected, the login worked as expected. An issue that may arise is forgetting to restart and enable the service. Upon testing the TFTP functionality in the PXE test boot, I was greeted with a timeout error. To correct this error, I looked at the DHCP server configuration and noticed that the boot server was spelled incorrectly. Once this was corrected, I received another error that said “operating system not found.” I went back into the DHCP configuration and found that the boot file was also misspelled. I fixed the spelling and rebooted again only to be greeted by yet another error message. This time, it was able to boot into the PXE server, but could not find the Centos8 file. This was due to the fact that I had it named centos8 and not Centos8. I rebooted again and was still unable to install the image. I went into the Centos8 file and removed all instances of /centos/ from the URLs, which is the incorrect location. When it tried to install after fixing this, the installer began but told me it had insufficient disk space. To fix this, I changed the RAM of the test machine to 4GB, which solved the issue.I also ran into a slew of permission issues, which was solved by either logging into root or adding sudo to every command. When installing the Apache server, I had forgotten to set up the DNS record on the Windows server, which meant the PXE server had no repository to point to. I also had to move the C8s-server-ks.cfg to a new directory called ks. I still had issues, and realized that the C8s-server-ks.cfg file was spelled wrong. I also noticed the Kickstart file had typos that needed to be fixed. Fixing these mistakes allowed the process to run as intended.