Jesse Russell
IA462 – Advanced Operating Systems
Christopher Krieger
Assignment Two

Part One: Building Windows and Active Directory

The first thing I did to set up this section of the lab environment was to create a virtual machine for my Windows Server 2019 ISO to sit on. This VM will be used as a domain controller for the rest of the lab network, and will be responsible for the DHCP, DNS, and Active Directory services. When creating this virtual machine, I left most of the settings default as they're not too important to change. Then, I ensured that the VM was connected to the Windows virtual network I created in the last lab and booted the machine up. Upon finishing the installation of Windows Server 2019, I then went into the Network and Sharing Center and changed the IPv4 adapter's IP settings to statically assign the IP addresses as shown in the table below. For the DNS settings, I assigned the box to use local DNS, by inputting 192.168.156.5 as the DNS server, then setting the alternate DNS server to the PFsense box (192.168.156.1).

IP Address	Subnet Mask	Default Gateway
192.168.156.5	255.255.255.128	192.168.156.1 (PFsense)

Once the networking was set up, I then installed the tools required for Active Directory Domain Services, DHCP, and DNS by clicking "Add Roles and Features" and following the prompts, selecting the services I needed. I decided to rename my server at this point to "2k19-DC", just to keep the network organized and the name easy to remember. Once those services were installed, I then had to configure them. In order to do so, I created a forest called "ia462.com." I then added some OUs to the domain and configured the group policies for all the OUs. I also added firewall rules for RDP, and NTP. For DHCP, I added scopes for both the WindowsNet and the LinuxNet. These scopes allow for any machine in those subnets to be assigned an IP address from the DHCP server, so that I don't have to manually create static IP addresses for every machine on the network. The address pool for the WindowsNet is 192.168.156.2 through 192.168.156.119. I then configured the LinuxNet pool in the same way but ranging from 192.168.156.130 to 192.168.156.249. These pools should provide more than enough addresses for my needs. For the DNS server, I allowed the server to point to the PFSense box as well as the server. I also configured forward and reverse lookup zones in order to navigate the network quickly. In PFSense, I added the DNS zones to BIND after installing the package, as well as set up access control lists for my two subnets.

Part Two: Install a Management/Monitoring Method for Windows Updates

In order to preserve space on my VM drive, I decided it would be best to use my AD server as my utility server. This way, I only have one server running instead of two separate instances. For the purposes of the lab, I believe this should be sufficient to handle both the functions mentioned in part one and the server management outlined in part two. However, due to the nature of WSUS, I decided to add a new 60 GB drive to the server. I also bumped the RAM of the server to 4GB, opposed to the standard 2 GB. This should allow the server enough memory to handle the tasks I've outlined. My host machine only has 8 GB, so any more memory allocated to the VM would probably slow down the machine and make it unusable.

To make managing this server easy, I created an MMC called MGMT and saved it to my documents. This allows me to configure the settings I need in one place, opposed to finding them in in the server manager. I can also transport and backup the MMC in case anything happens to the server. This way, I don't have to reconfigure the entire network should disaster strike, and I can just restore from backup. Here, I created my OU structure and linked any existing GPOs I hadn't done before. The structure is as follows: under the Domains tree, I created an OU called is462. Then, I created 3 sub-OUs called Compute, Groups, and Users. With these OUs, I can add specific GPOs to certain groups and computers, which is exactly what I did when I set up the WSUS GPOs (discussed further).

For this task, I set up a WSUS server to manage Windows Updates for the organization. To do this, I first had to install the feature using the "Add Roles and Features" wizard. I then used all the defaults the wizard provided for the installation process. I added the update services snap-in to the MGMT MMC I created for the server, using Microsoft Update as location to get the updates from.

I also made sure to configure the APP-RDP and APP-WINRM GPOs to allow those services to function properly. I had to create predefined rules for both in the Windows Defender and Firewall section of the GPO editors, and had to ensure the firewall was turned on for them From this point, all I had to do was configure the GPOs to properly receive updates from the server I just created. Under updates in the MMC, I selected the 2k19-DC server, then added three new computer groups under the "computers" tree, Laptops, Servers, and VDI. Then I created the same groups in the Group Policy management tree under Forest: ia462.com > Domains > ia462 > Compute. I created multiple GPOs to configure certain aspects of the server: APP-WSUS, APP-GRP-WSUS-Servers, and APP-WSUS-GRP-VDI. Under APP-WSUS, which is linked to the domain to apply to the entire organization, I enabled automatic update detection, set the automatic updates to check every day at 3:00 pm, and specified the intranet for detecting updates to http://2k-19-DC.ia462.com:8530. I enabled both Client Side Targeting for both the APP-GRP-WSUS-Servers and APP-GRP-WSUS-VDI objects, and then linked those objects to their respective groups in the Group Policy Management tree. After all of this was completed, I started syncing the server again and left it running overnight.

Part Three: Troubleshooting

When I tried to set up my server to sit on the network correctly, and assign IP addresses to the IPv4 interface, I was unable to get an internet connection and failed to reach any IP outside of the network. In order to fix this issue, I disabled the Ethernet0 adapter in the Network and Sharing Center, then reenabled it. At this point, I verified that the symbol at the bottom had changed from a yellow exclamation point (indicating there was no internet) to a connected symbol. I was then able to ping outside IP addresses, such as google.com.

Upon renaming the server, I had forgotten that the machine needs to restart when the name is changed. Due to this, I had to repeat the steps outlined in the section where I discussed installing Active Directory Domain Services. Renaming your machine prior to completing these steps may save time and frustration.

As I promoted the server to a DC, I received an error at the prerequisites check. This error specified that the local administrator password was blank and did not meet the minimum password requirements. This confused me, because I did set the password for my account. This was when I realized that I was not signed into the administrator account, I was signed into my own account. Upon this realization, I signed into the administrator account and changed the password.

During part two, when I tried to add a second disk to the server, File Manager wouldn’t see the drive I added. I was able to look in Device Manager and see the extra drive, but it was currently unusable in this state. This was a simple fix. All I had to do was go into the computer management settings, then allocate a volume to the drive and assign a drive letter (I chose E:).

I ran into multiple issues with my network connection, which prevented me from getting a lot of the work done in a timely manner. Usually, a restart to the router did the trick. I’m assuming this is because I was using a lot of bandwidth in accomplishing these tasks, and the router was overloaded or my apartment building throttled the connection. Either way, I was able to get back up and running within a short period of time.