Setting up Windows Server 2019
Active Directory Using Virtual Machines

Setting up a Windows Server 2019 Active Directory

For this project, we will use virtual machines, specifically Windows 10 and Windows Server 2019 VMs in Virtual Box.

First, let us rename the VM PC. Right-Click the start menu and click System. Click Rename PC. We will rename the PC Domain Controller. At this point, the system will restart.

We will open the Windows Server 2019 VM and begin configuring the server. First, start with accessing the network connections; Click Change Network Adapter. once the network connections have opened, figure out which connection is connected to the internet; it is usually the one on the left.

Changing the Network Adapter Settings

Go to the Network Adapter by:

Right-clicking the network icon in the bottom-right corner>Open Network Adapter Settings.

Now let us rename the connection. I am going to rename the connections **EXTERNAL CONNECTION.** The other connection should be the internal connection.

Now we will rename this one **INTERNAL CONNECTION**. Now Right-Click the **INTERNAL CONNECTIONS** one and Click Properties. Select the IPv4 option and Properties. Use the following IP address: 172.16.0.1 and subnet mask 255.255.255.0. No default gateway. Now select Use the following DNS server address. Change that to 127.0.0.1.

This address is a loopback address to recognize itself.

Configuring Domain Admin

Now let's configure the Active Domain Direct Services (ADDS). Using the Windows Server 2019 program to create a domain. Click Add Roles and Features. Click Next>Next>Now pick the server (Should Only be One)>Next. Now choose Active Domain Direct Services and Add the features>Next>Next>Next>Install. This may take a while.

Now that configuration is complete, look in the top-right corner for a flag with caution next; this is for post-deployment configuration. We have installed the software, so we are configuring the post-deployment configuration by clicking Promote this server to a domain controller. First, click Add New Forest and rename the domain we will use, **mydomain.com**. Now enter a password using **Password1**. Now Next out and install. A prompt will restart the VM. The login screen will now be changed, and re-login. When the VM restarts, you will notice that the name has changed on the screen; this is OK; login in with the same password: **Password1.**

Setting up a Domain Admin

Now let us create a domain admin:

Click Start>Windows Administrative Tools>Active Directory Users and Computers.

Now let's make an organizational unit:

Right-Click the domain site>New, and select Organizational Unit.

Now name it **_ADMIN_**, uncheck it to protect the container from accidental deletion, and click OK.

Now double-click the mydomain.com that has appeared in the right panel.

Now right-click the _ADMIN_ folder>New>User.

Use whatever name you want. For example, I will use **A-Chris Sexton**.
For the user name, businesses typically use first name initial and last name, for instance, **A-CSexton**>Next.

For the password, use the same password that has been used, **Password1,** and uncheck any boxes (Normally, in real situations, you would make a user change the password on login, but for this lab, we are not)>Next>Finish.

Now the user is created. Chris Sexton is an admin, so we have to change the privileges. Right-click the user>select properties>Member of tab>Add>Type in Domain Admins>Check>OK>Apply>OK.

Now we have created an admin domain account. Now restart the VM, and we will log in with the new admin account.

So now we will restart the VM and log back in with our newly created admin account. When signing back in, select Other User in the bottom-left corner and enter the credentials for the newly created admin **(A-CSexton/Password1)**.

Configuring a RAS/NAT

We will install a Remote Access Server (RAS) and Network Access Translation (NAT). We are doing this to allow the Windows 10 client to access the server.

Click Add Roles and Features>Next>Next>Select the server> Next>Select Remote Access>Next>Next>Next>Select Routing>Add Features and then Next out and Install. This may take a while.

Now click tools (Top-Right) and select Routing and Remote access.

When the window opens, Right-click the Local network>Select Configure and Enable Routing and Remote Access, and begin the configuration wizard. Next>Network Address Translation (NAT)>Use the public interface to connect to the internet>Next>Finish.

Now that the NAT/RAS has been configured, we will set up a Dynamic Host Configuration Protocol (DHCP).

The DHCP will allow when the user connects to the server; the DHCP will assign an IP address to the user.

Next, select Roles and Features>Next>Select the Server>Next>Select DHCP Server>Add Features and Next out and Install and close the window.

Now select tools (Top-Right) and select DHCP.

Creating Scope for DHCP

The DHCP assigns IP addresses to the users. In this window, we are creating the range of IP addresses that will be given.

Right-Click the server (mydomain.com)>New Scope>Name the Scope whatever you want>Next>Make the range 172.16.0.100-200>Next>Start IP addresses 172.16.0.100 and end 172.16.0.200>Length 24>subnet mask 255.255.255.0.

The next screen is for IP addresses you don't want to be used; if there is an IP address you don't wish to use, then input it in the box.

Next will be the lease duration; this is how long a client can use their assigned IP address. This is mainly used for public access, like an internet cafe or a coffee shop, to limit how long a client can use that IP address. For example, this location may set it for 2 hours or 8 hours, but we are putting it for 8 days for this lab.

The next screen is to configure DHCP options and select the yes option. This allows the remote access user to use the internet from this server. Next, enter the domain controller's IP address (172.16.0.1)>Add.

The next screen lists the specified domain site, the IP address is correct, and the next out and finish. Select activate the scope.

When the window minimizes, right-click and refresh the servers. Now authorize the server by right-clicking and authorize, then refresh, and the DNS is set up.

For this lab, we are allowing the users to browse the internet. We usually don't want to enable this in a production business and the IT environment, but we are for this lab.

From the primary server screen, select Configure This Local Server. Now we need to select enhanced security and disable this security feature. Click ON and disable these features.

Also, we need to add users for this lab, but installing each user separately would take a while, so we will use a code to input the users for us. The code I got from GitHub. Exact the Zip file and move the folder to the desktop.

Adding Users From a PowerShell Script

Now we will be using PowerShell. Click Start>Right-Click PowerShell ISE and run as administrator, and PowerShell will open.

Next, put in the Command:

```
C:\Users\A-CSexton\Desktop\AD_PS-master\1_CREATE_USERS.ps1
```

See how it did not allow the script to be loaded. This is because we have to disable a security feature. To disable this security feature, type in the command:

```
Set-ExecutionPolicy Unrestricted
```

A caution will appear, and click yes to all.

Now we must change the directory in PowerShell. Use the command line:

```
cd C:\Users\A-CSexton\Desktop\AD_PS-master
```

Now the command line in PowerShell will look something like this:

```
C:\Users\A-CSexton\Desktop\AD_PS-master>
```

Now we will look for the names.txt file using the command:

```
Ls
```

And it will look like this:

```
Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        2/10/2023   2:12 PM           1811 .gitignore
-a----        2/10/2023   2:12 PM           1025 1_CREATE_USERS.ps1
-a----        2/10/2023   2:12 PM           1532 Generate-Names-Create-Users.ps1
-a----        2/10/2023   2:21 PM          15582 names.txt
```

Now we will run our script using the command:

```
PS C:\Users\A-CSexton\Desktop\AD_PS-master>
C:\Users\A-CSexton\Desktop\AD_PS-master\1_CREATE_USERS.ps1
```

The script should be running, and the text (depending on your PowerShell settings) should be a cyan color. Remember, it is one thousand names so it will take a while.

Of course, we would not do this in situations, but for this lab, we are so we can see what it will look like with multiple users on our server.

Brief Explanation of the Script

`$PASSWORD_FOR_USERS = "Password1"` is the password all the users will have

The script line

`$USER_FIRST_LAST_LIST` = Get Content .\names.txt is the list of names in the text file and copying and pasting them for the learning for this lab.

I could explain more about the coding, but this lab is setting up an Active Directory so we can move on.

Configuring and Using Windows 10 VM with our Windows Server

In the VM screen, go into the settings of the Windows 10 VM and change the network adapter from NAT to Internal.

When the Windows 10 VM is up and running, open the Command line and check and see if the internet is working. Type in:

```
ipconfig
```

Look for the IP address on the IPv4 line and say 172.16.0.100, and the default gateway should communicate 172.16.0.1. Next, ping the domain site. Enter Command:

```
ping mydomain.com
```

A ping response should occur.

Now we are going to check the hostname. Use Command:

Hostname

And the name will appear, but it will have a default name, so let us change that. Right-click Start>system>scroll down and select advanced rename.

Don't enter a computer description; click the Change icon and rename the computer. For example, I will change the name to Chris-Sextons-Computer, and in the Member of area, enter the domain website mydomain.com>OK in the Member of area. Now enter the domain admin credentials.

CSexton
Password1

A restart prompt will appear if the prompt does not restart the Windows 10 VM.

Now go back to Tools (Top-Left Corner)>DHCP. Select Scope>Address Leases, and one lease should be assigned.

Now let us go to the Windows Server Users and Computers.

Click Start>Windows Administrative Tools>Active Directory Users, and Computers>Select Computer in the left panel, and the computer should appear, and the computer has joined the domain.

Now go back to the Windows 10 VM:

The Windows 10 VM has restarted, and instead of logging in with the same account, let's try a different account. Put in the name of the user and enter the password:

CSexton
Password1

The Windows 10 VM should log in and begin a new setup for a new Windows 10 VM.

Now let's go to the command line and enter the command:

```
Whoami
```

This will display which domain the computer is accessed to and the name of the user:

```
mydomain/CSexton
```