

How to Create an AD Test Environment

 raxis.com/blog/create-ad-environment

April 27, 2023

Lead Pentester **Andrew Trexler** walks us through creating a simple AD environment.

Whether you use the environment to test new hacks before trying them on a pentest, or you use it while learning to pentest and study for the OSCP exam, it is a useful tool to have in your arsenal.

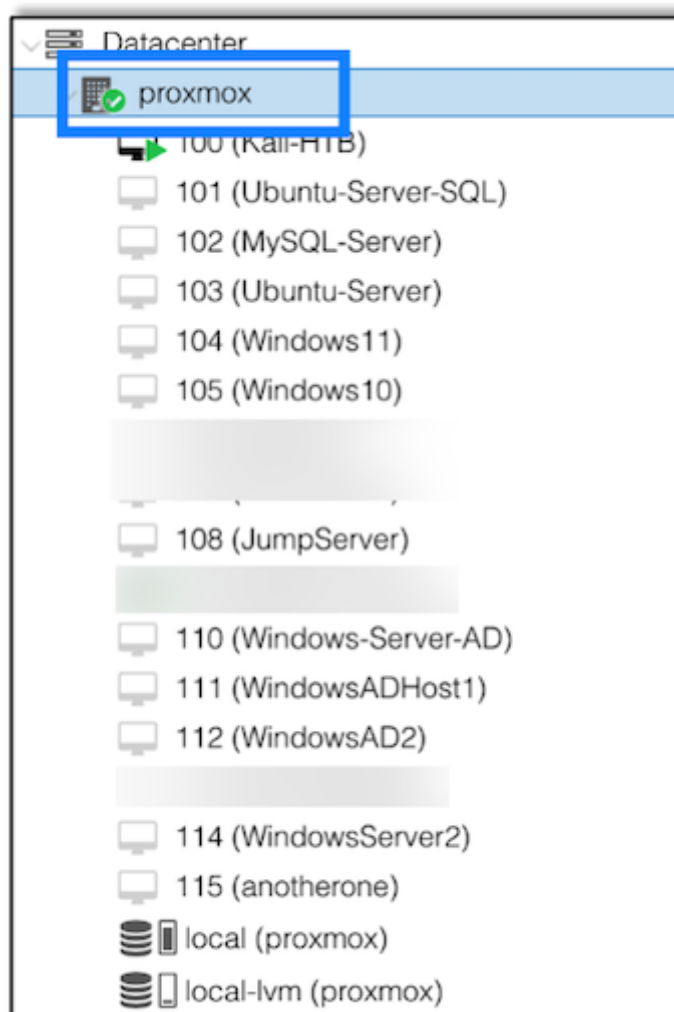
The Basics

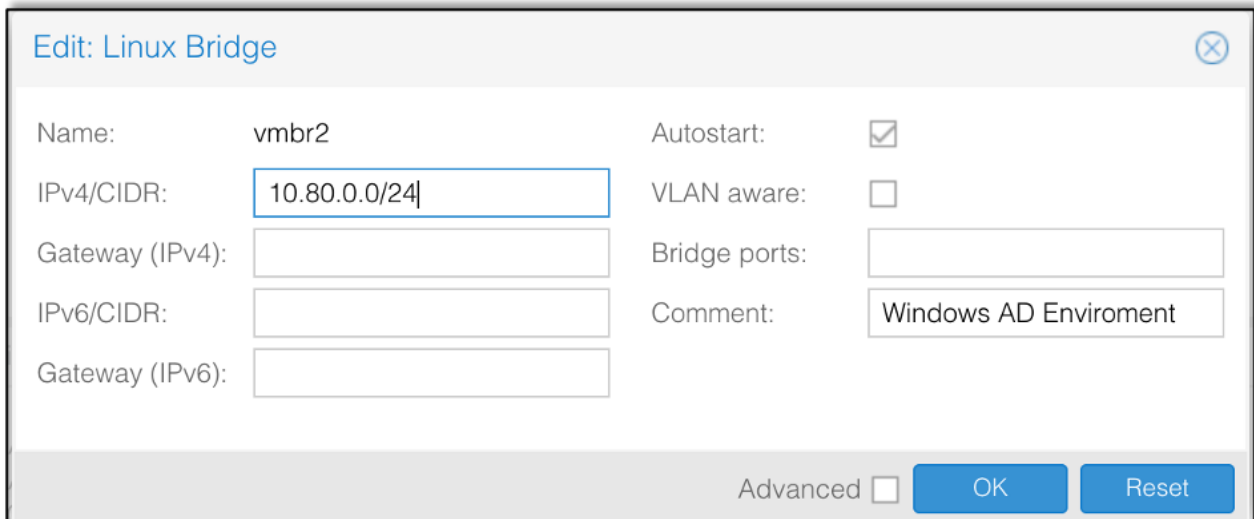
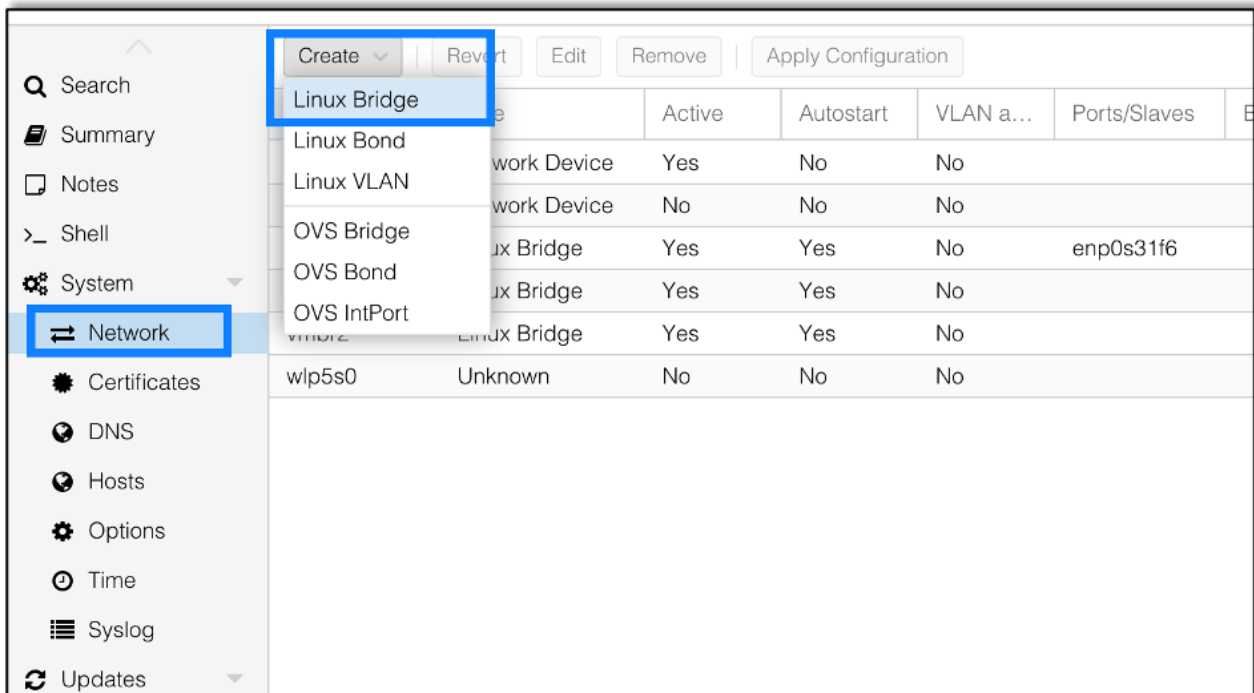
Today we'll go through the steps to set up a Windows Active Directory test environment using Proxmox to virtualize the computers. In the end, we'll have a total of three connected systems, one Domain Controller and two other computers joined to the domain.

Setting up the Domain Controller (DC)

The first step is to setup a new virtualized network that will contain the Windows Active Directory environment. Select your virtualization server on the left:

This is a Windows based environment, but we're using a Linux hypervisor to handle the underlying network architecture, so under *System*, select *Network*, and then create a Linux Bridge, as shown in Figures 2 and 3:





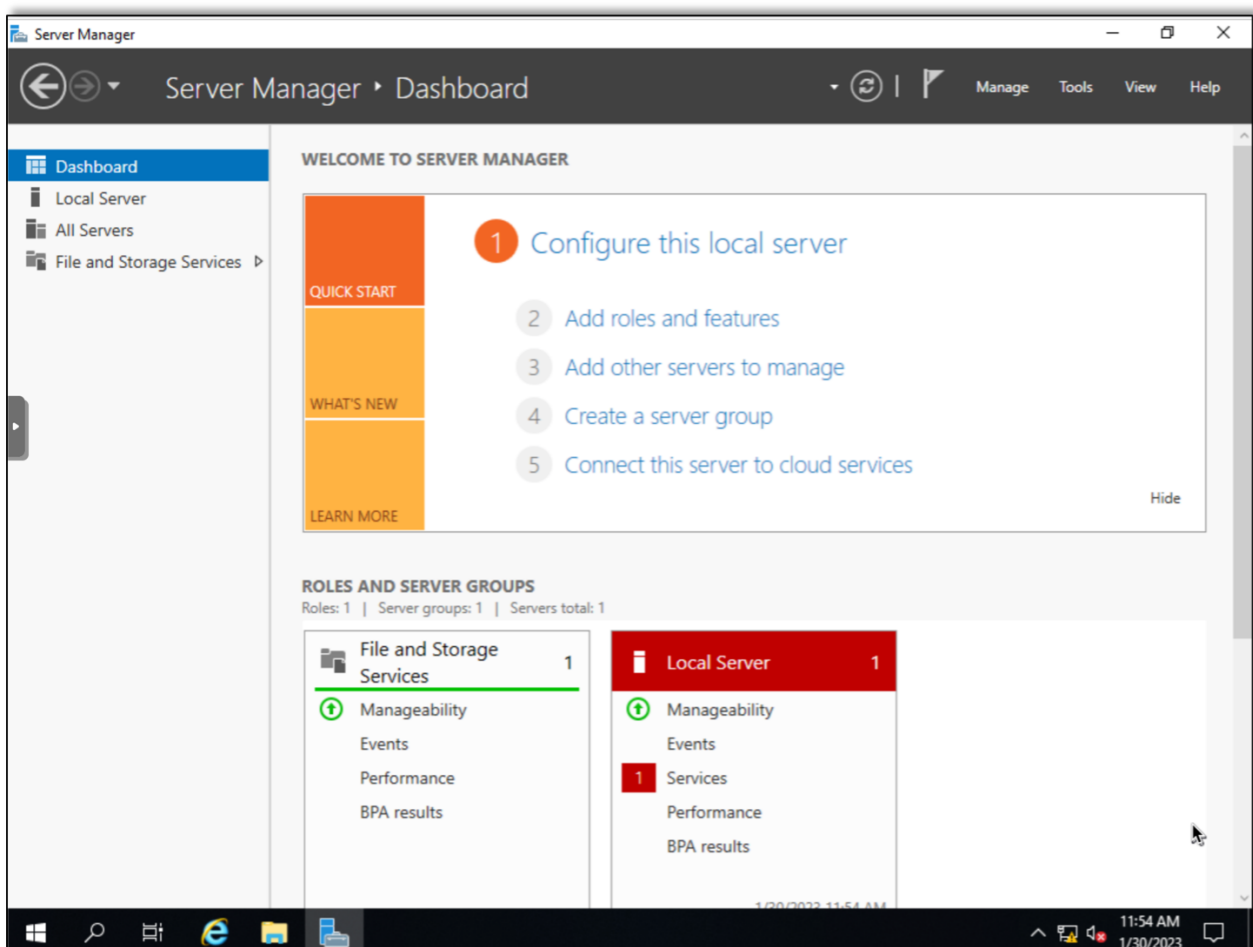
After setting up the network, we provision a new virtual machine where we will install Windows 2019 Server. Figure 4 shows the final configuration of the provisioned machine:

Key ↑	Value
cores	2
ide2	local:iso/Windows_Server_2019_17763.737.190906-2324.rs5_release_svc_refresh_SER...
memory	4096
name	Windows-Server-AD
net0	e1000,bridge=vmbr2,firewall=1
nodename	proxmox
numa	0
ostype	win10
sata0	local-lvm:40
scsihw	virtio-scsi-single
sockets	1
vmid	110

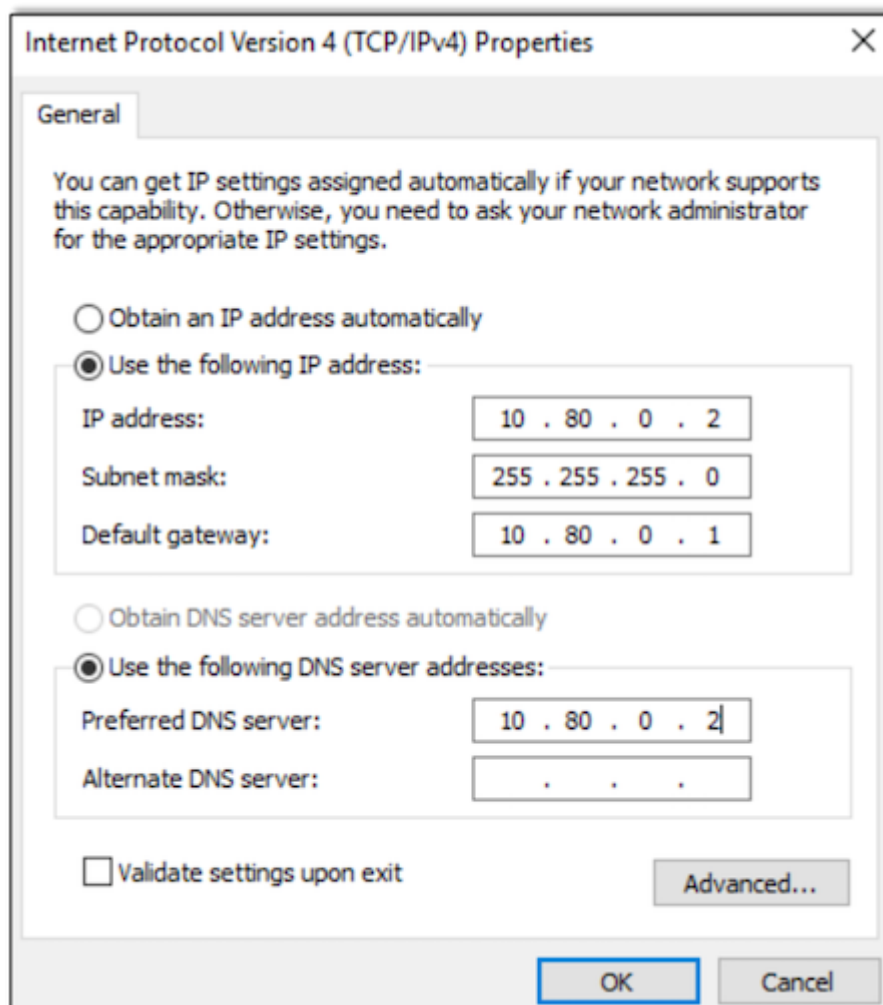
☐ Start after created

Advanced ☐ [Back](#) [Finish](#)

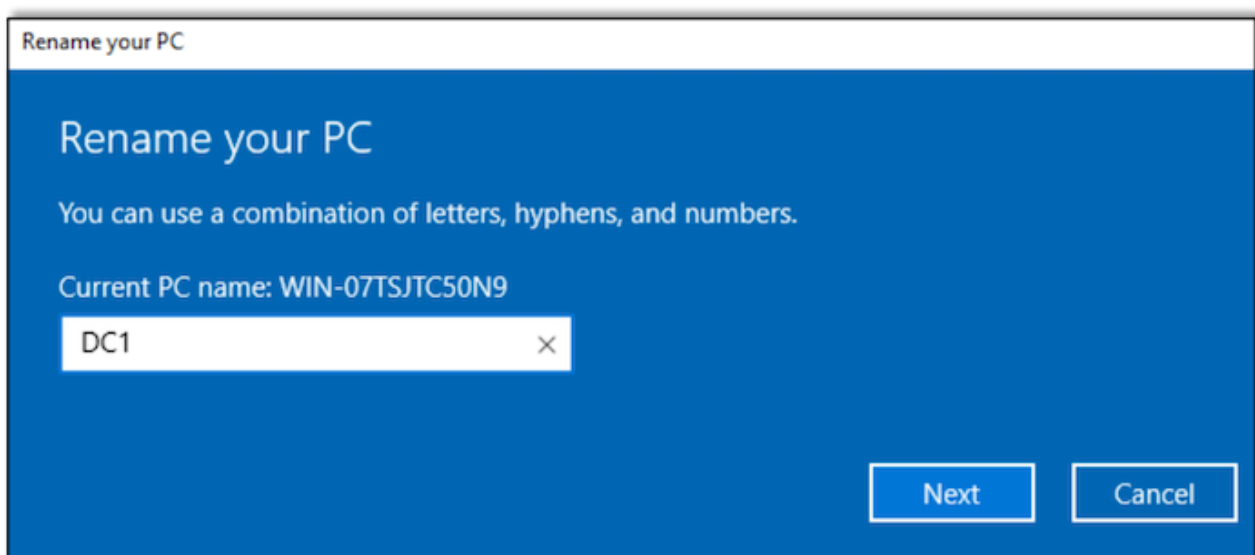
The next step is to install Windows 2019 Server. While installing the operating system make sure to install the Desktop Experience version of the operating system. This will make sure a GUI is installed, making it easier to configure the system.



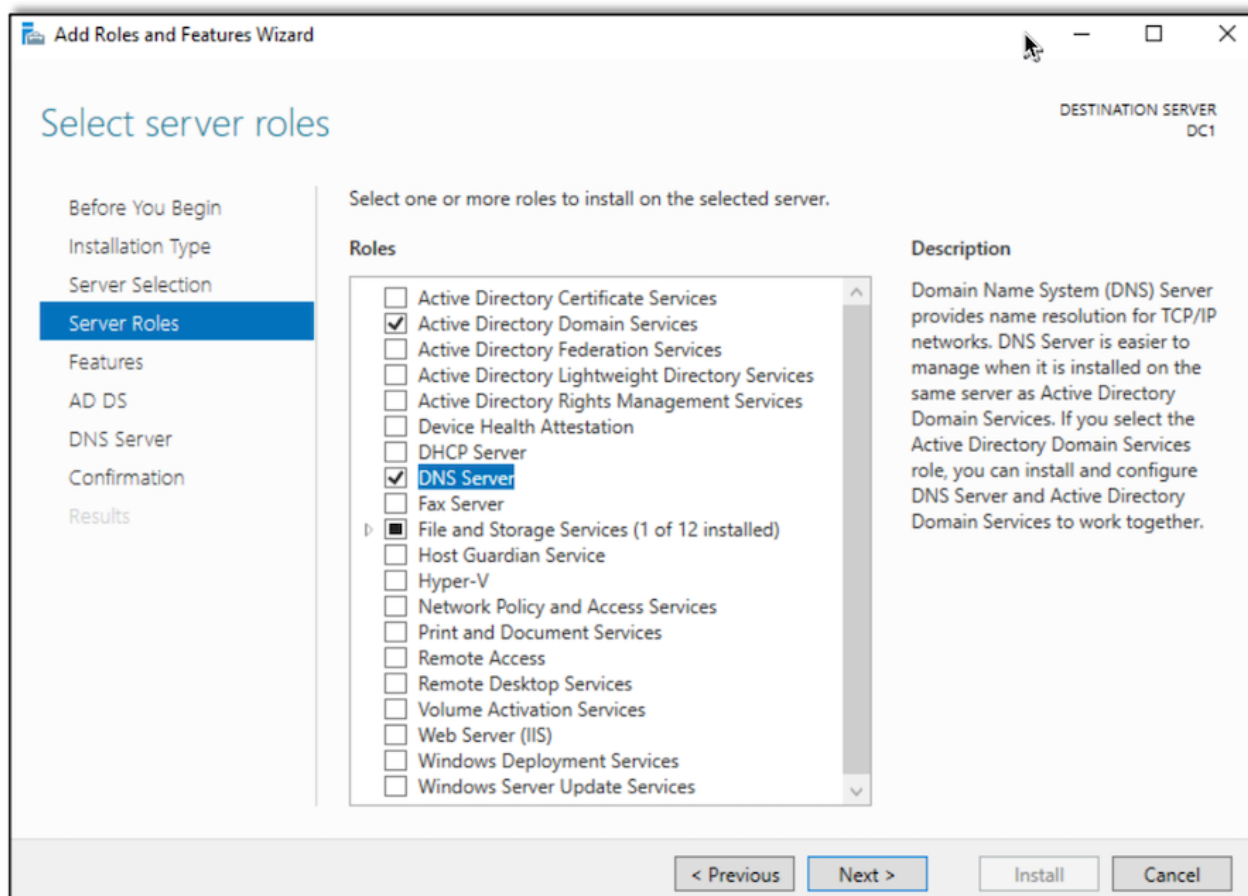
Now that we have a fresh install, the next step is to configure the domain controller with a static IP address. This will allow the server to function as the DHCP server. Also make sure to set the same IP as the DNS server since the system will be configured later as the domain's DNS server.



In order to make things easier to follow and understand later, let's rename the computer to DC1 since it will be acting as our domain controller on the Active Directory domain.



Next, configure the system as a domain controller by using the *Add Roles and Features Wizard* to add the *Active Directory Domain Services* and *DNS Server* roles. This configuration will allow the server to fulfill the roles of a domain controller and DNS server.



After the roles are installed, we can configure the server and provision the new Active Directory environment. In this lab we will use the domain *ad.lab*. Other than creating a new forest and setting the name, the default options will be fine.

The screenshot shows the 'Active Directory Domain Services Configuration Wizard' window. The title bar includes the Microsoft logo and the text 'Active Directory Domain Services Configuration Wizard'. The window has standard Windows window controls (minimize, maximize, close) in the top right corner. The main area is titled 'Deployment Configuration'. On the left, there is a vertical list of steps: 'Deployment Configuration' (highlighted in blue), 'Domain Controller Options', 'Additional Options', 'Paths', 'Review Options', 'Prerequisites Check', 'Installation', and 'Results'. On the right, the text 'Select the deployment operation' is followed by three radio button options: 'Add a domain controller to an existing domain', 'Add a new domain to an existing forest', and 'Add a new forest' (which is selected). Below this, the text 'Specify the domain information for this operation' is followed by a label 'Root domain name:' and a text box containing 'ad.lab'. At the bottom right of the main area, there is a link that says 'More about deployment configurations'. At the bottom of the window, there is a grey bar containing four buttons: '< Previous', 'Next >' (highlighted in blue), 'Install', and 'Cancel'. In the top right corner of the main area, the text 'TARGET SERVER DC1' is displayed.

Active Directory Domain Services Configuration Wizard

Deployment Configuration

TARGET SERVER
DC1

Deployment Configuration

Domain Controller Options

Additional Options

Paths

Review Options

Prerequisites Check

Installation

Results

Select the deployment operation

☐ Add a domain controller to an existing domain

☐ Add a new domain to an existing forest

☒ Add a new forest

Specify the domain information for this operation

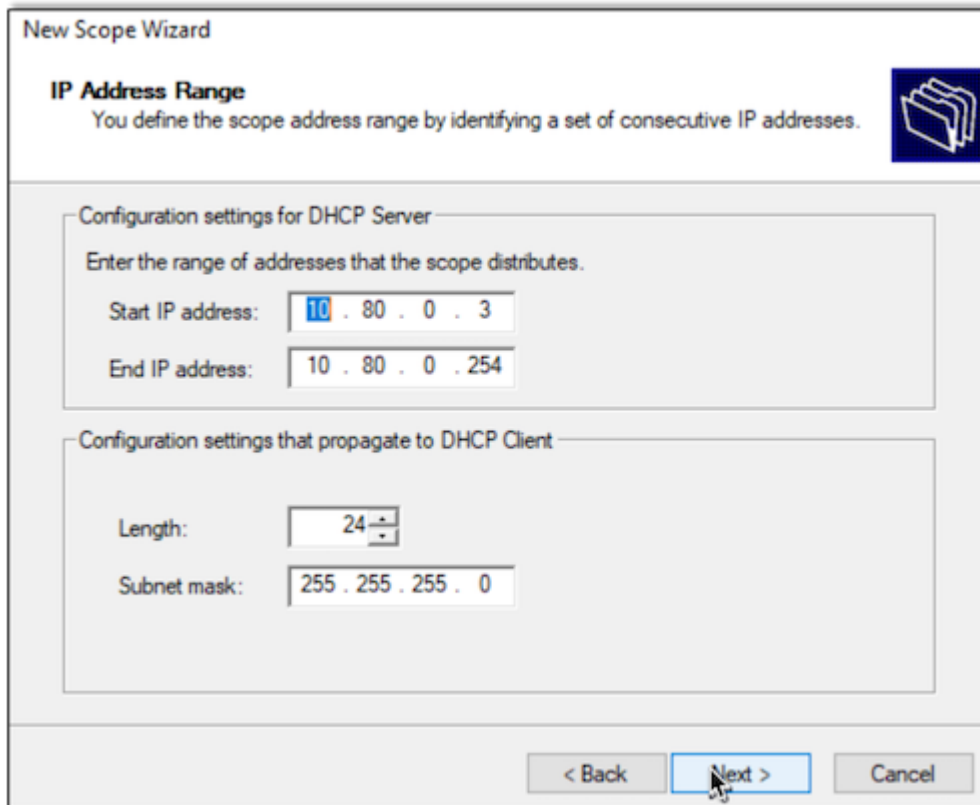
Root domain name:

[More about deployment configurations](#)

< Previous Next > Install Cancel

Setting Up the DHCP Service

The next step is to configure the DHCP service. Here we are using a portion of the 10.80.0.0/24 network space, leaving enough addresses available to accommodate static IP addressing where necessary.



The image shows a 'New Scope Wizard' window with the title 'IP Address Range'. Below the title is a description: 'You define the scope address range by identifying a set of consecutive IP addresses.' To the right of this text is a blue icon of a folder with a document. The window is divided into two main sections. The first section is titled 'Configuration settings for DHCP Server' and contains the instruction 'Enter the range of addresses that the scope distributes.' It has two input fields: 'Start IP address:' with the value '10 . 80 . 0 . 3' and 'End IP address:' with the value '10 . 80 . 0 . 254'. The second section is titled 'Configuration settings that propagate to DHCP Client' and contains two input fields: 'Length:' with the value '24' and 'Subnet mask:' with the value '255 . 255 . 255 . 0'. At the bottom of the window are three buttons: '< Back', 'Next >', and 'Cancel'. A mouse cursor is pointing at the 'Next >' button.

New Scope Wizard

IP Address Range
You define the scope address range by identifying a set of consecutive IP addresses.

Configuration settings for DHCP Server

Enter the range of addresses that the scope distributes.

Start IP address: 10 . 80 . 0 . 3

End IP address: 10 . 80 . 0 . 254

Configuration settings that propagate to DHCP Client

Length: 24

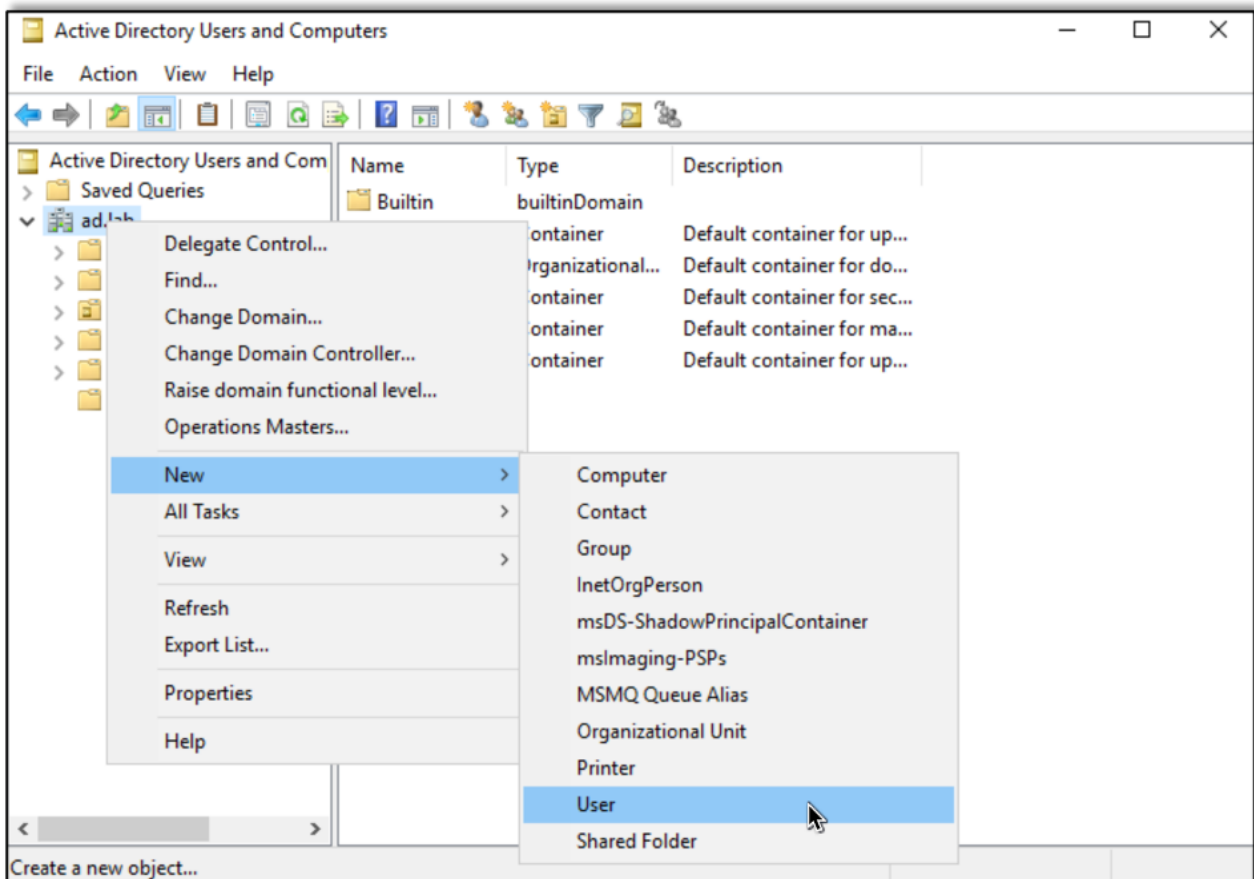
Subnet mask: 255 . 255 . 255 . 0

< Back Next > Cancel

There is no need for any exclusions on the network, and we will set the lease to be valid for an entire year.

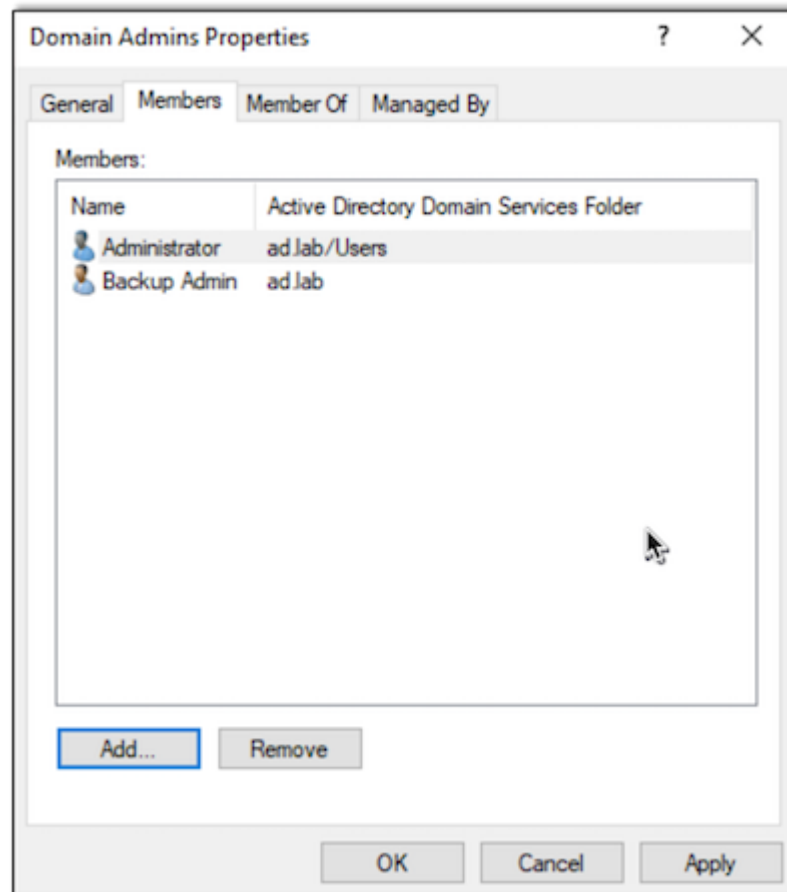
Adding a Domain Administrator and Users

Additional configuration is now required within the domain. Let's add a new domain administrator and some new domain users. Their names and passwords can be anything you want, just make sure to remember them.

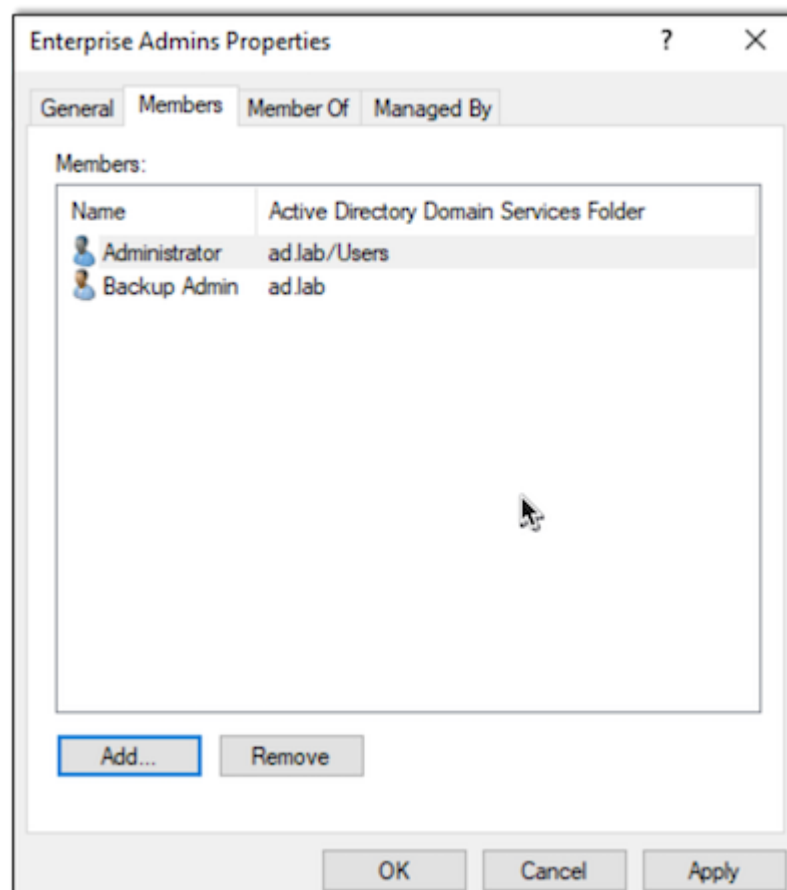


First we create the Domain Administrator (DA):

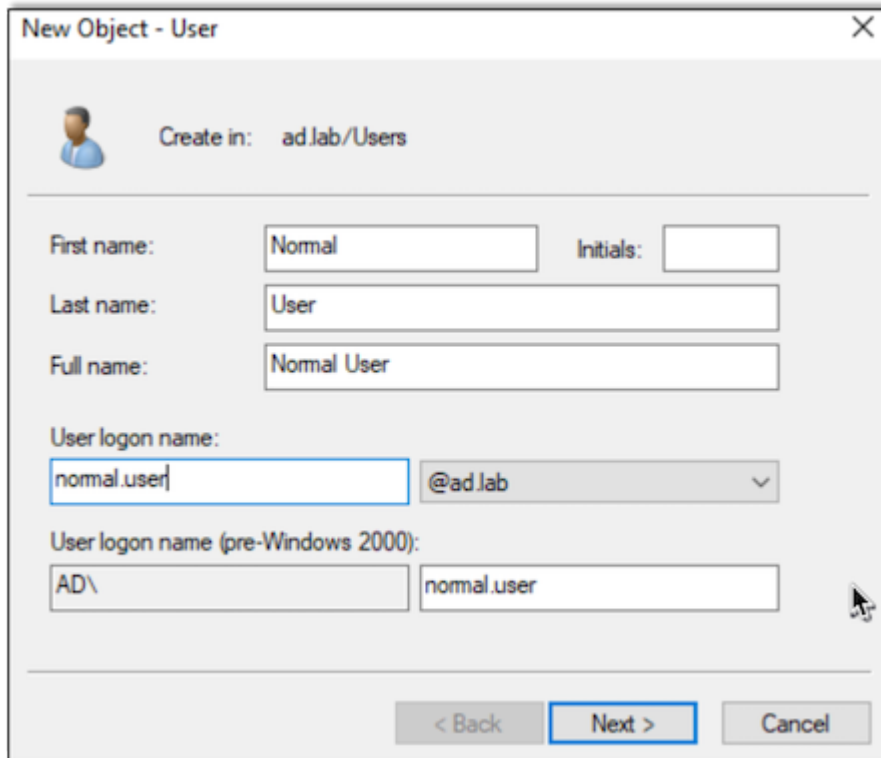
The 'New Object - User' dialog box is shown. It has a title bar with 'New Object - User' and a close button. Below the title bar is a user icon and the text 'Create in: ad.lab/'. The dialog contains several input fields: 'First name:' with 'Backup' entered, 'Initials:' (empty), 'Last name:' with 'Admin' entered, and 'Full name:' with 'Backup Admin' entered. Below these is 'User logon name:' with 'backup.admin' entered and a dropdown menu showing '@ad.lab'. Underneath is 'User logon name (pre-Windows 2000):' with 'AD\' entered in the first box and 'backup.admin' in the second box. At the bottom are three buttons: '< Back' (disabled), 'Next >' (active/highlighted), and 'Cancel' (disabled).



Here we also make this user an Enterprise Admin (EA) by adding them to the Enterprise Admins group:



Next we will add a normal user to the domain:



The screenshot shows the 'New Object - User' dialog box in Active Directory. The 'Create in' field is set to 'ad.lab/Users'. The 'First name' field contains 'Normal', the 'Last name' field contains 'User', and the 'Full name' field contains 'Normal User'. The 'User logon name' field contains 'normal.user' and the domain dropdown is set to '@ad.lab'. The 'User logon name (pre-Windows 2000)' field contains 'AD\' and the 'normal.user' field contains 'normal.user'. The 'Next >' button is highlighted.

Creating Windows PC

At this point we should have a functional Active Directory domain with active DHCP and DNS services. Next, we will setup and configure two other Windows 10 machines and join them to the domain.

The first step is to provision the resources on the Proxmox server. Since our test environment requires only moderate resources, we will only provision the machines with two processor cores and two gigabytes of RAM.

Create: Virtual Machine

General

OS

System

Disks

CPU

Memory

Network

Confirm

Key ↑	Value
cores	2
ide2	local:iso/Windows_10_Enterprise_2022.iso,media=cdrom
memory	2048
name	WindowsHost1
net0	e1000,bridge=vbr2,firewall=1
nodename	proxmox
numa	0
ostype	win10
sata0	local-lvm:40
scsihw	virtio-scsi-single
sockets	1
vmid	111

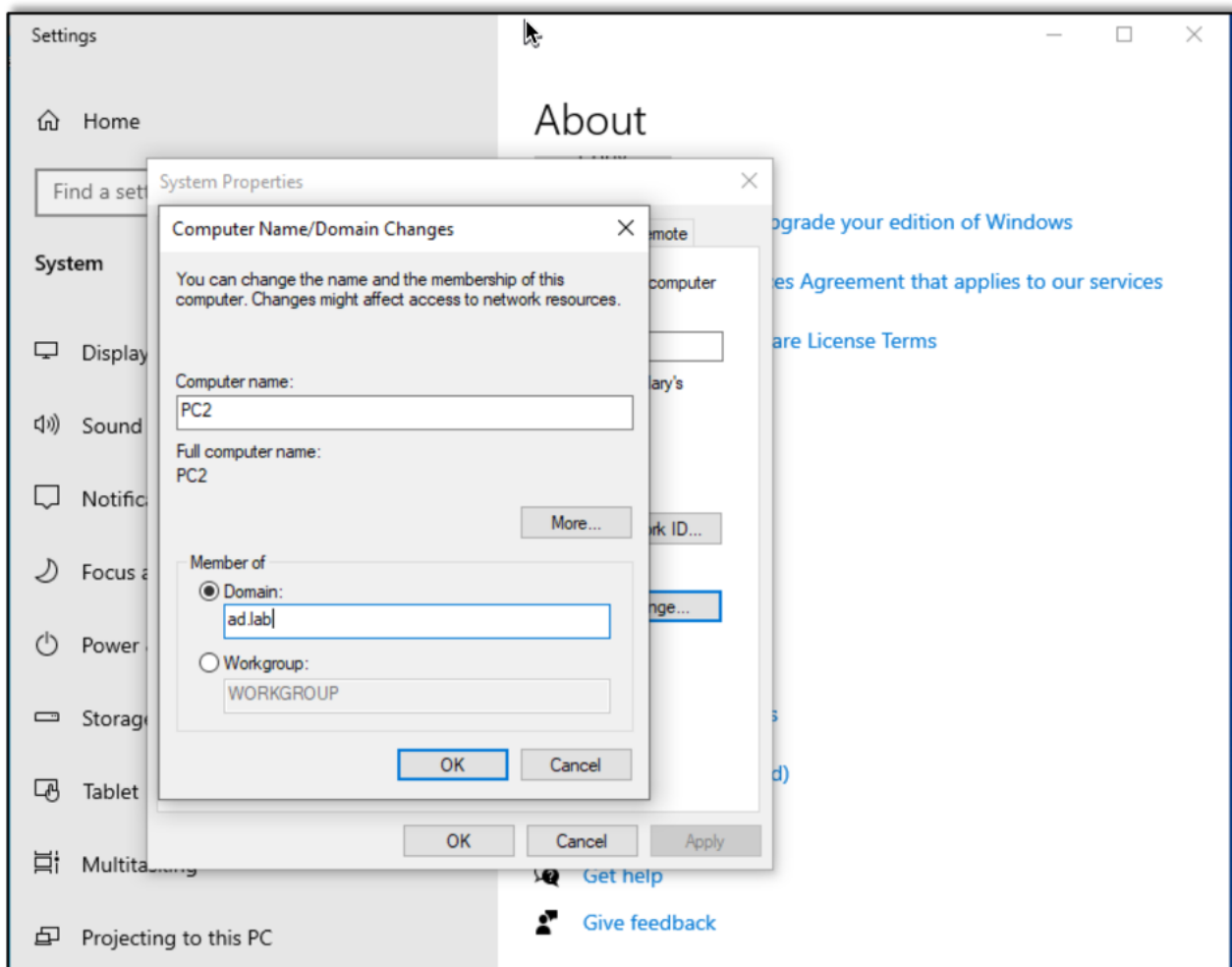
☐ Start after created

Advanced ☐

Back

Finish

Then we install Windows 10 using the default settings. Once Windows is installed, we can open the *Settings* page and join the system to the *ad.lab* domain, changing the computer name to something easy to remember if called for.



Adding the system to the domain will require us to enter a domain admin's password. After a reboot we should be able to login with a domain user's account.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

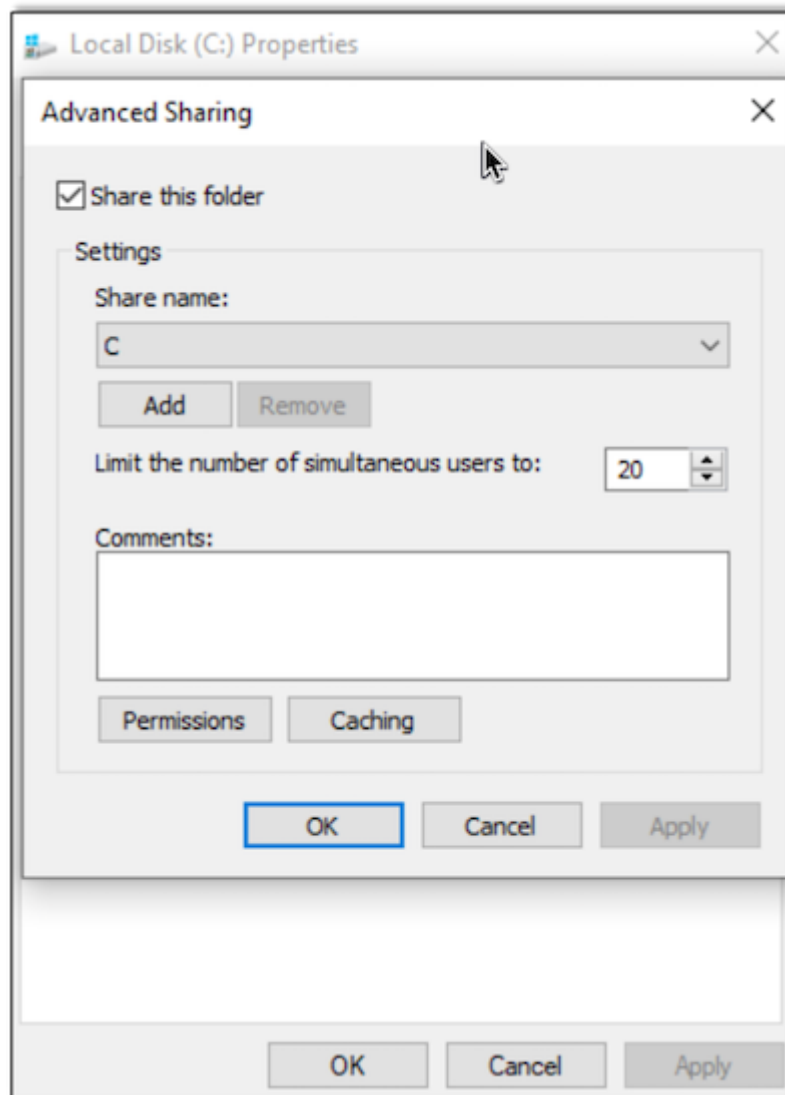
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Raxis> whoami
ad\raxis
PS C:\Users\Raxis>
```

SMB Share

At this point, there should be three computers joined to the Active Directory domain. Using CrackMapExec, we can see the SMB server running on the domain controller but no other systems are visible via SMB. So let's add a new network share. Open *Explorer.exe*, select *Advance Sharing*, and share the C drive.

I don't recommend sharing the entire drive in an environment not used for testing, as it's not secure: the entire contents of the machine would be visible. Since this is a pentest lab environment, though, this is exactly what we are looking for.



Creating the share resulted in the system exposing the SMB service to the network. In Figure 20 we verified this by using CrackMapExec to enumerate the two SMB servers:

```
~ crackmapexec smb 10.80.0.0/24 Mon 30 Jan 2023 03:14:49 PM EST
/usr/lib/python3/dist-packages/pywerview/requester.py:144: SyntaxWarning: "is not" with a literal. Did you mean "!="?
if result['type'] is not 'searchResEntry':
SMB 10.80.0.4 445 PC2 [*] Windows 10.0 Build 19041 x64 (name:PC2) (domain:ad.lab) (si
gning:False) (SMBv1:False)
SMB 10.80.0.2 445 DC1 [*] Windows 10.0 Build 17763 x64 (name:DC1) (domain:ad.lab) (si
gning:True) (SMBv1:False)
~ 11.1s < Mon 30 Jan 2023 03:15:02 PM EST
```

Conclusion

At this point, our environment should be provisioned, and we are ready to test out different AD test cases, attacks, and other shenanigans. This environment is a great tool for ethically learning different exploits and refining pentesting techniques. Using a virtual infrastructure such as this also provides rollback capability for running multiple test cases with minimal downtime.

I hope you'll come back to see my next posts in this series, which will show how to use this environment to test common exploits that we find during penetration testing.

Want to learn more? Take a look at the [next part in our Active Directory Series](#).