SECURITY+ V4 LAB SERIES

Lab 11: Configuring a RADIUS Server

Document Version: 2023-02-27

Material in this Lab Aligns to the Following	
CompTIA Security+ (SY0-601) Exam Objectives	3.8: Given a scenario, implement authentication and authorization solutions
All-In-One CompTIA Security+ Sixth Edition ISBN-13: 978-1260464009 Chapters	24: Implement Authentication and Authorization

Contents

Introduction

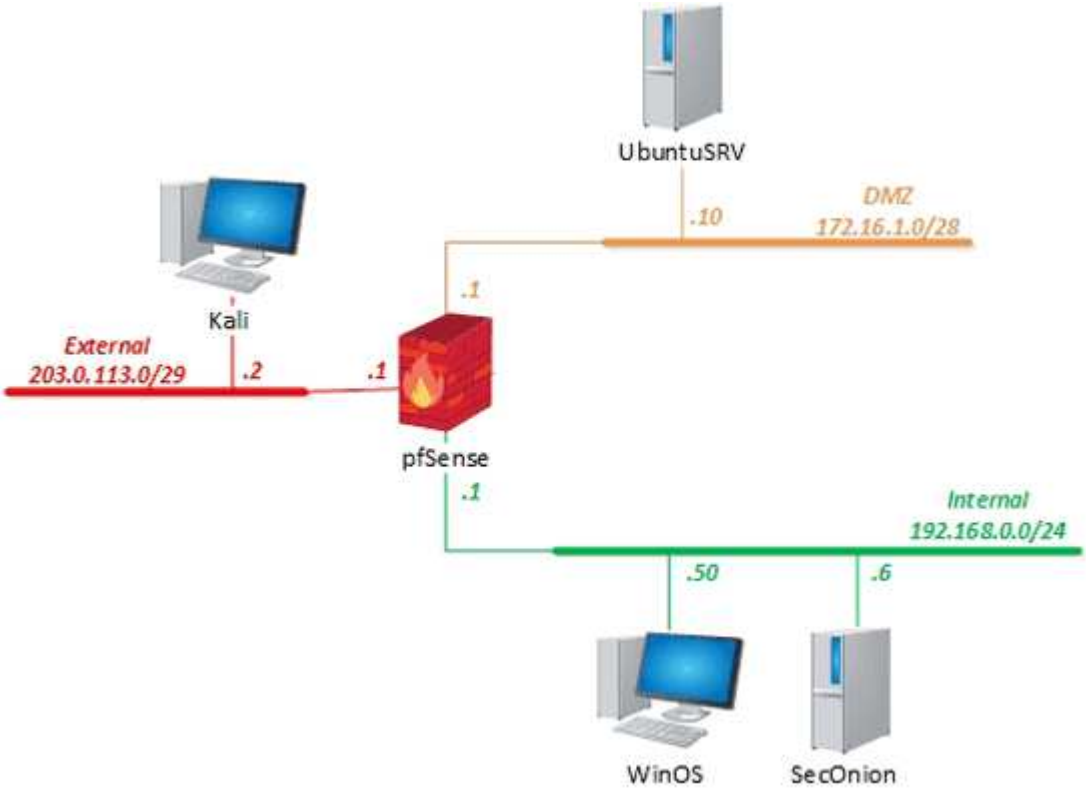
In this lab, you will install, configure, and deploy a RADIUS server within the Windows and Linux operating systems.

Objective

In this lab, you will perform the following tasks:

- Set up and configure RADIUS on Windows
- Set up and configure RADIUS on Linux

Lab Topology



Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

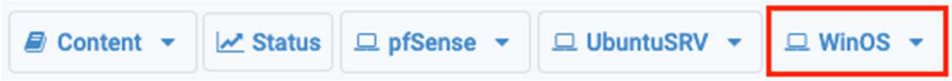
Virtual Machine	IP Address	Account (if needed)	Password (if needed)
Kali	203.0.113.2	kali	kali
pfSense	192.168.0.1	sysadmin	NDGlabpass123!
SecOnion	192.168.0.6	sysadmin	NDGlabpass123!
UbuntuSRV	172.16.1.10	sysadmin	NDGlabpass123!
WinOS	192.168.0.50	Administrator	NDGlabpass123!

1 Configure RADIUS on Windows

1.1 Add the Network Policy Role

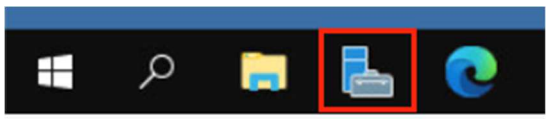
In this task, you will add the role of *Network Policy Server* to the *WinOS* server. You will then add the *Network Policy and Access Services Tools*.

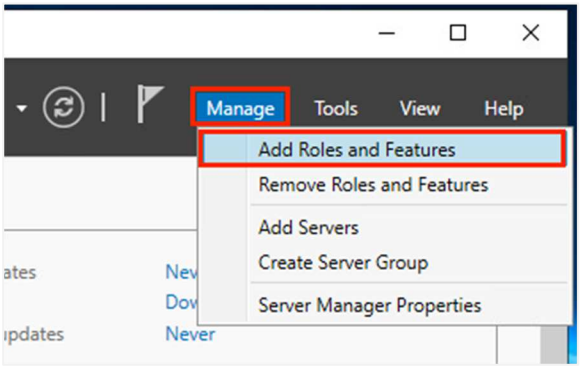
1. Launch the **WinOS** virtual machine to access the graphical login screen.



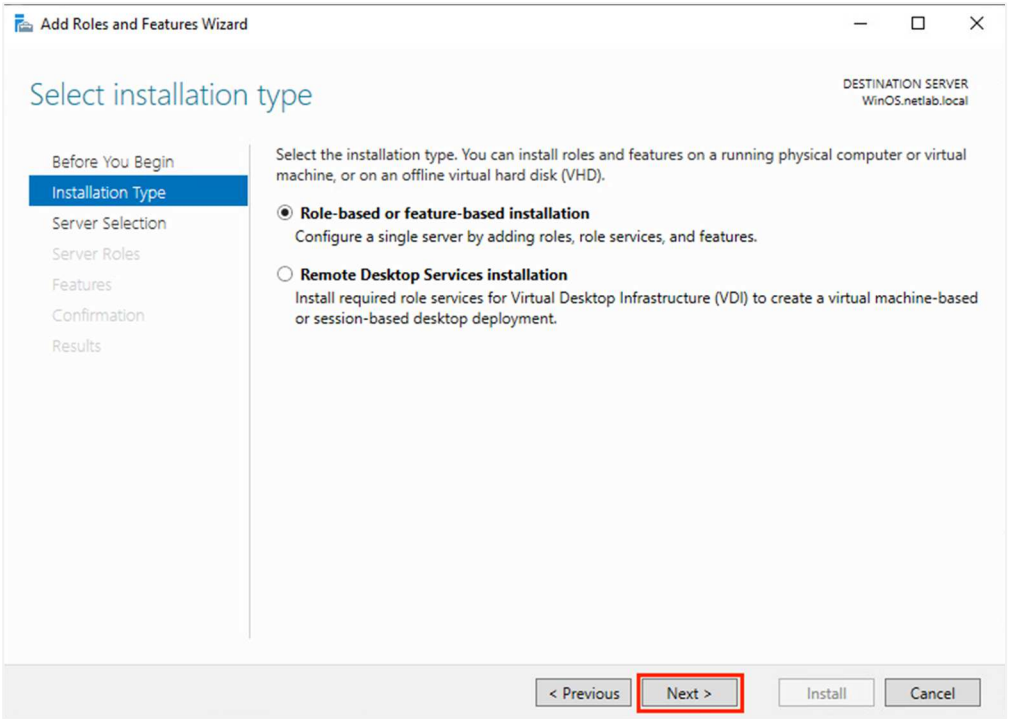
2. While on the splash screen, focus on the *NETLAB+* tabs. Click the dropdown menu for the **WinOS** tab and click on **Send CTRL+ALT+DEL**.
3. Log in as `Administrator` using the password `NDGlabpass123!`.

4. Once logged in, click the **Server Manager** icon to launch it. In the *Server Manager* window, click on **Manage** in the top-right corner and select **Add Roles and Features**.

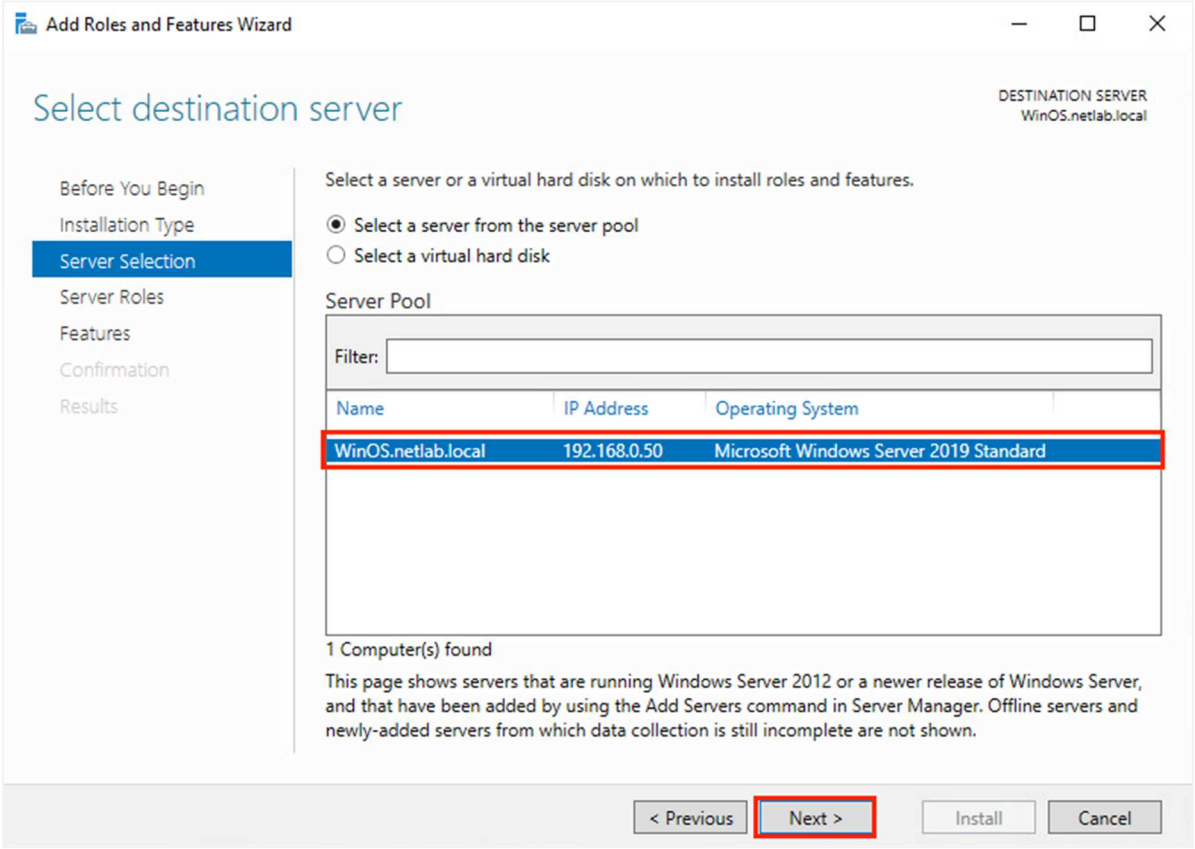




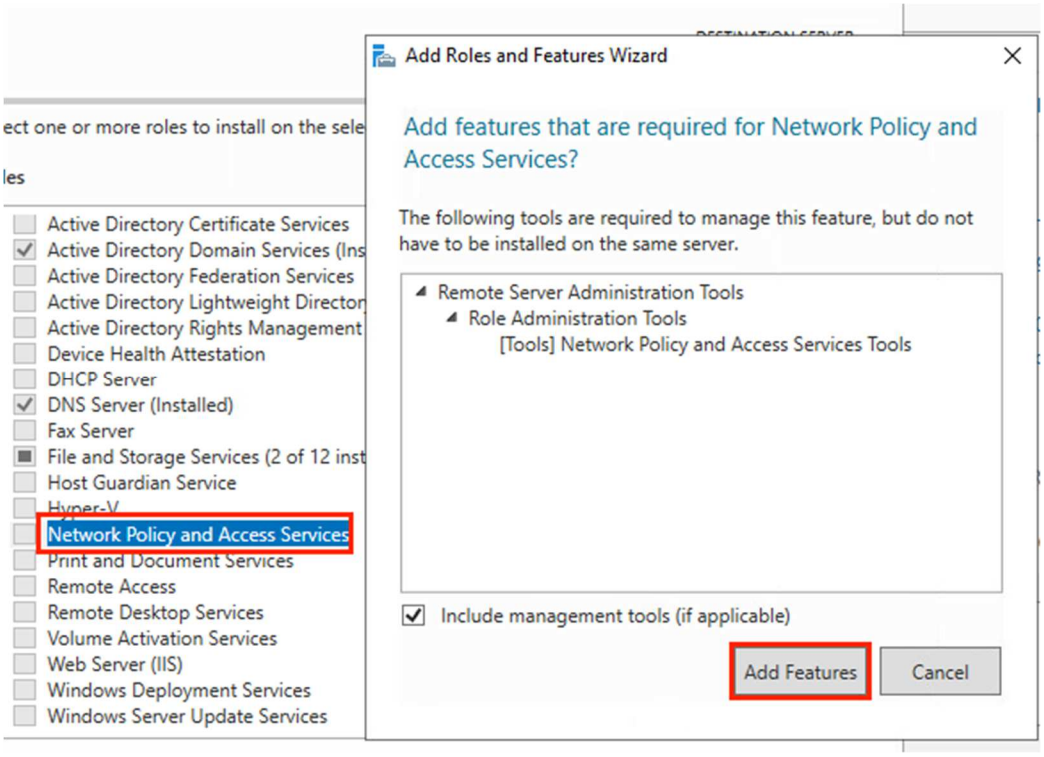
5. Notice the *Add Roles and Features Wizard* appears. On the *Installation Type* step, keep the default setting of **Role-based or feature-based Installation** and click **Next**.



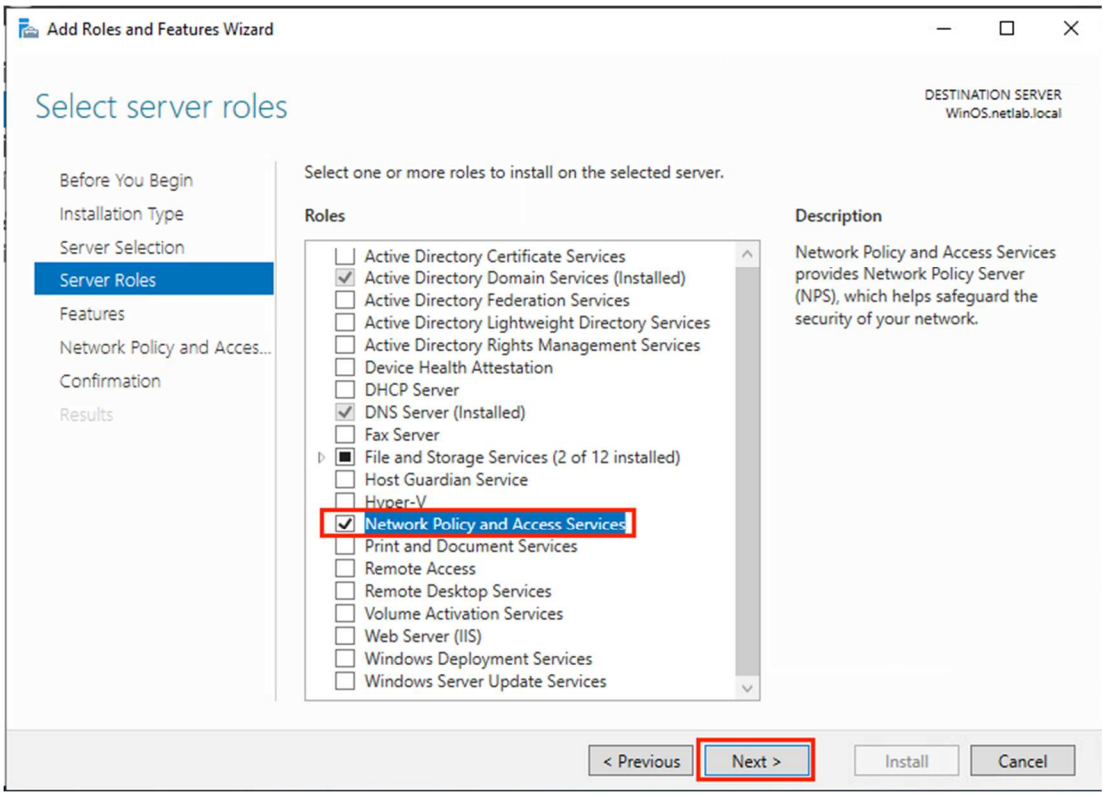
6. On the *Server Selection* step, select the **WinOS.netlab.local** server from the pool and click **Next.**



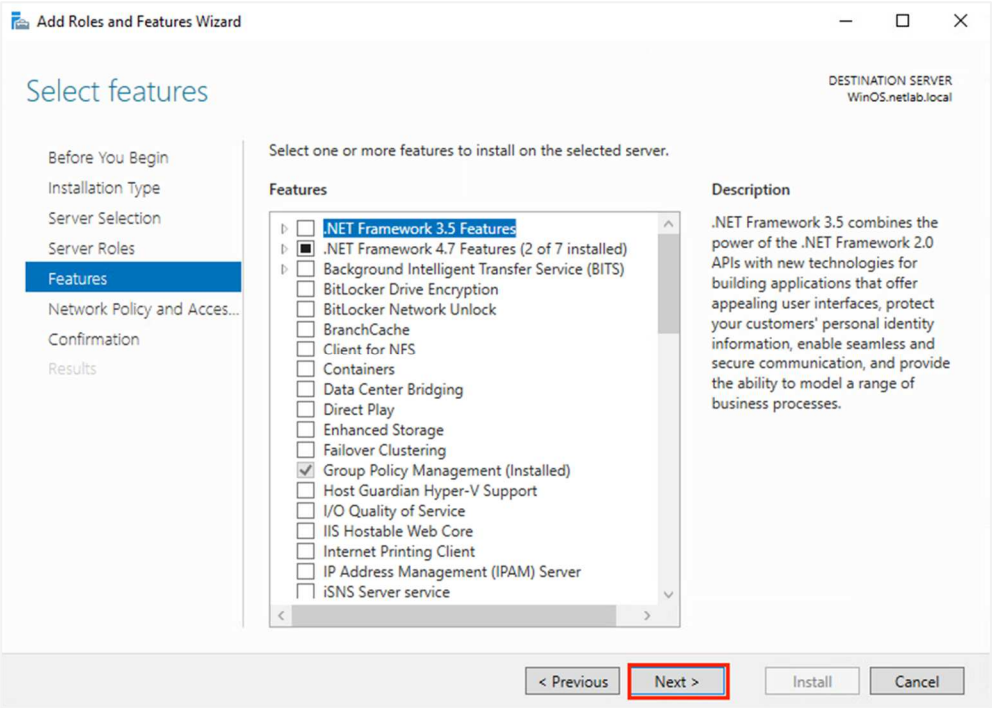
7. On the *Server Roles* step, check the checkbox for **Network Policy and Access Services** and notice a pop-up window appears. Click the **Add Features** button.



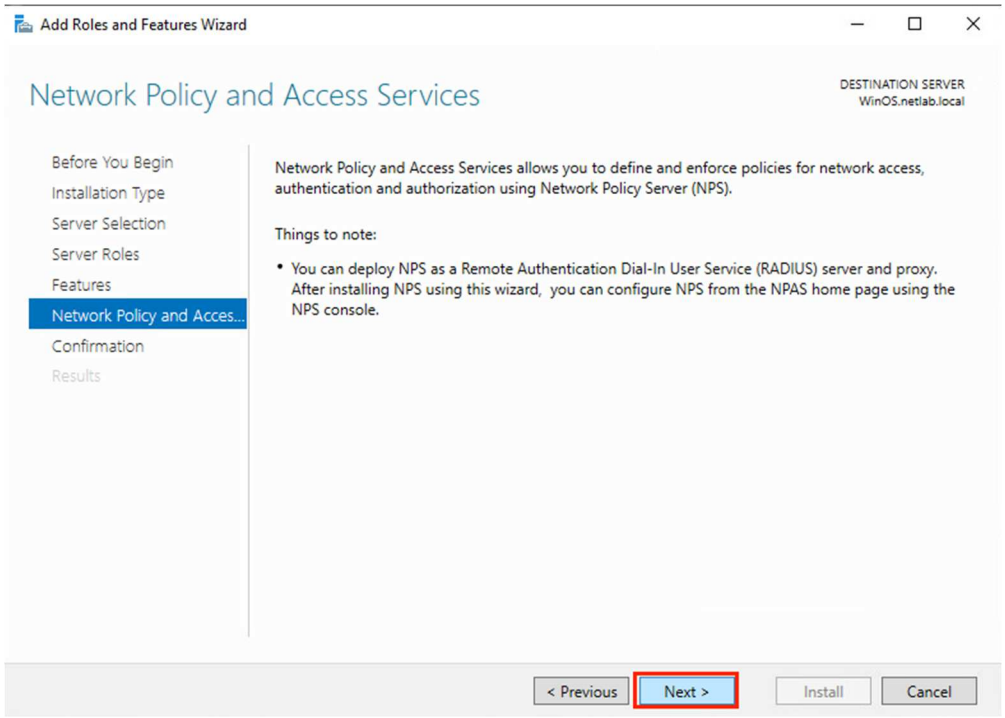
8. Back on the main wizard window, ensure that **Network Policy and Access Services** is checked, and click **Next.**



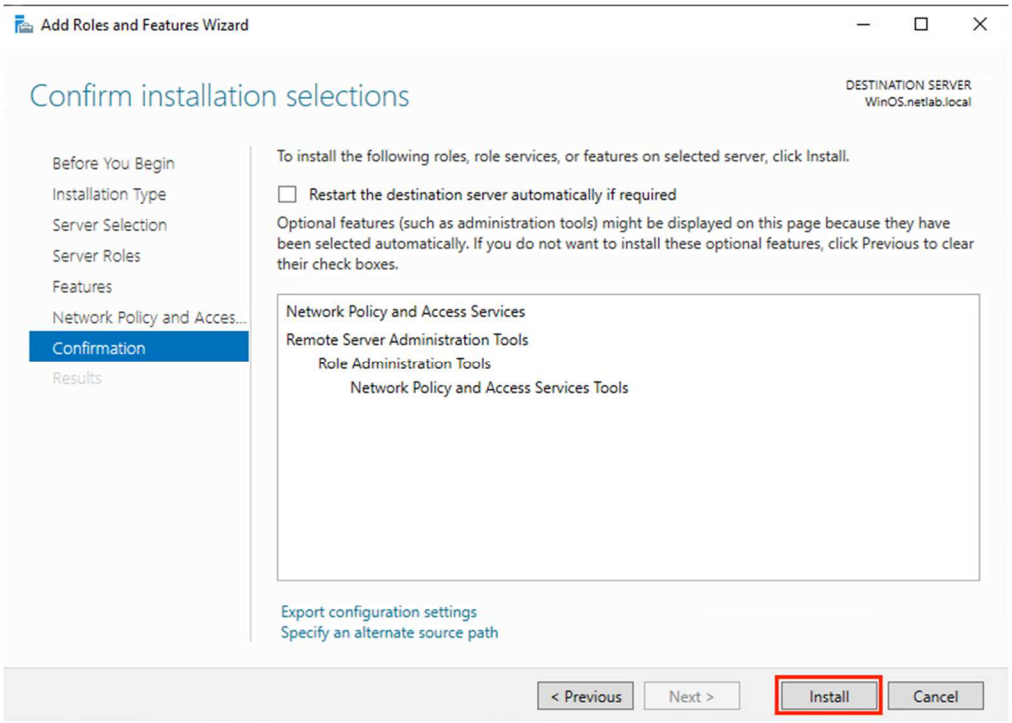
9. On the *Features* step, leave the defaults and click **Next.**



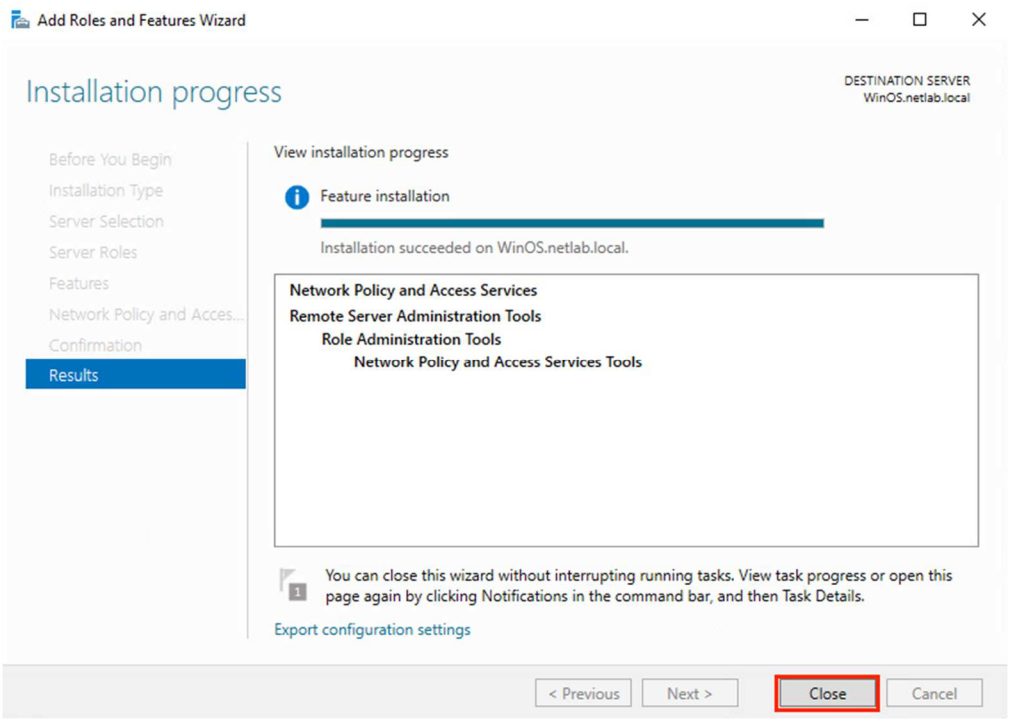
10. On the *Network Policy and Access Services* step, review the information and click **Next.**



11. On the *Confirmation* step, click **Install** to finish the installation of the *Network Policy Server.*

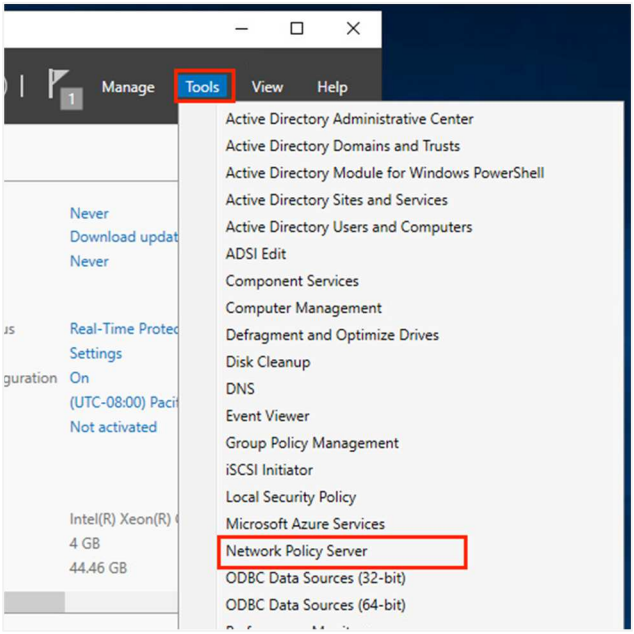


12. The installation will take about 3 minutes. After the install is complete, click **Close.**

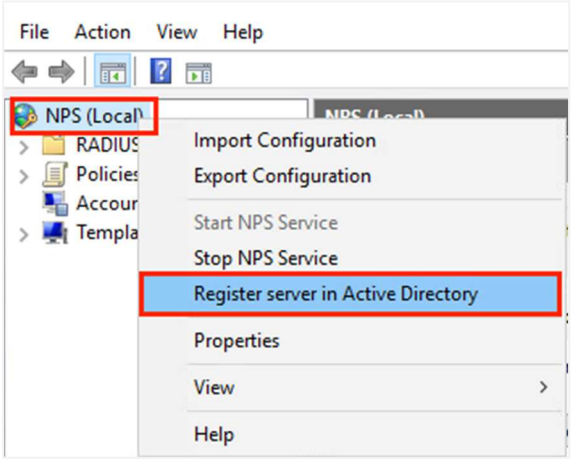


1.2 Configure the Network Policy Server

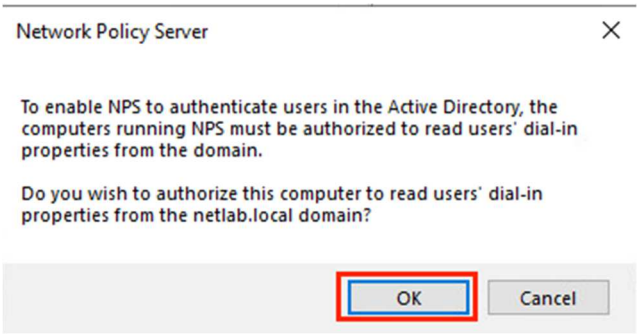
1. Back on the *Server Manager* window, navigate to **Tools > Network Policy Server**.



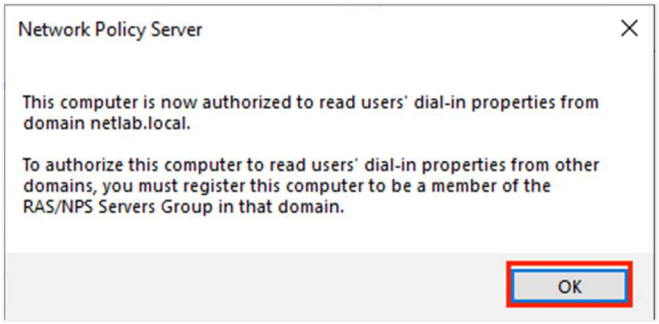
2. In the *Network Policy Server* window, in the left pane, right-click on **NPS (Local)** and select **Register Server in Active Directory.**



3. When prompted to authorize, click **OK** to continue.

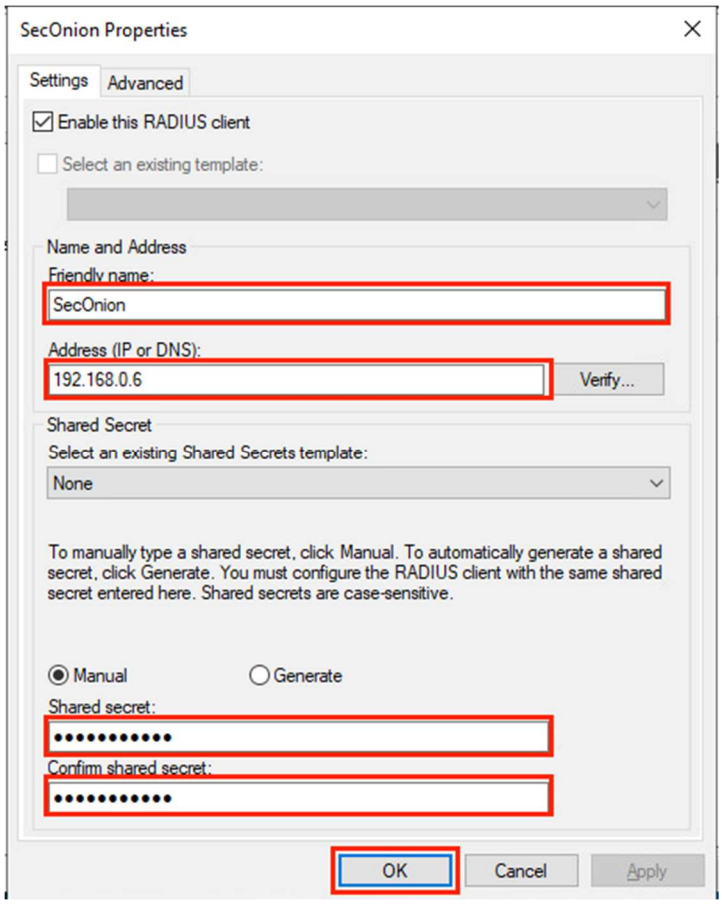


4. Click **OK** again to confirm that the system is now authorized.

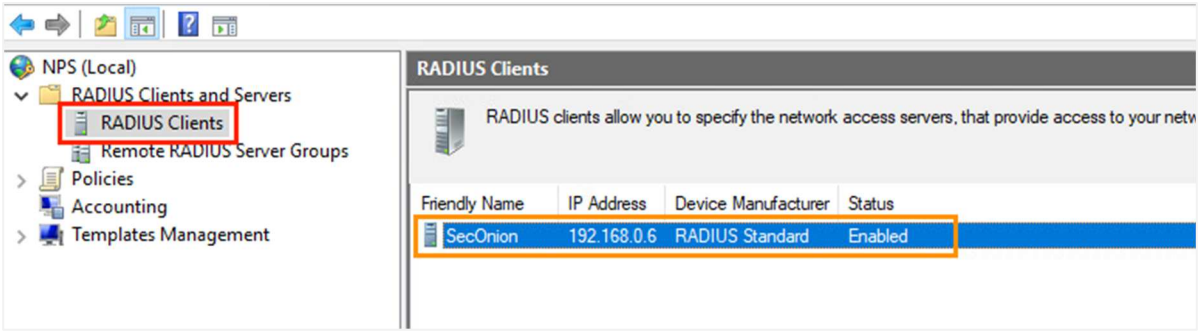


5. Expand the **RADIUS Clients and Servers** inventory object, right-click on **RADIUS Clients** and click **New.**

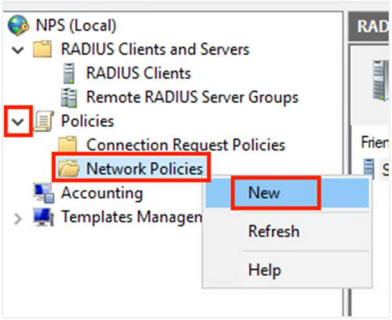
6. In the *New RADIUS Client* window, fill in the following information:
   a. *Friendly name*: `SecOnion`
   b. *Address*: `192.168.0.6`
   c. *Shared secret*: `password123`
   d. *Confirm shared secret*: `password123`



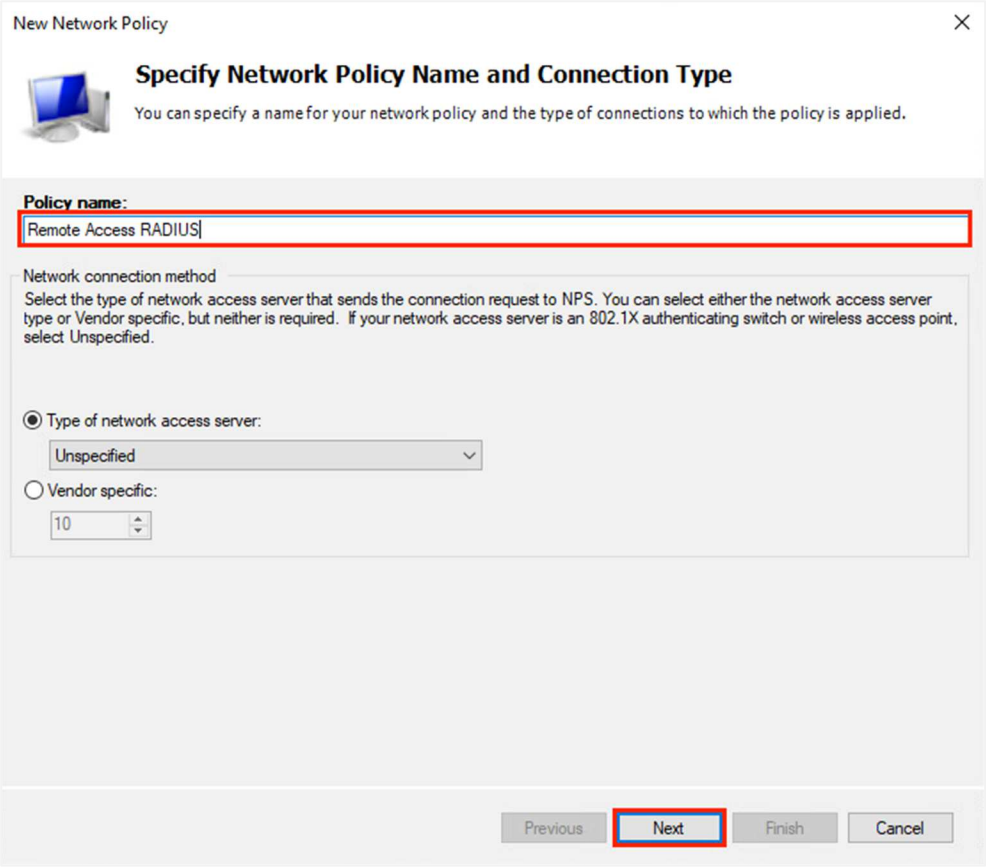
7. Once the configurations are made, click **OK.**
8. Select **RADIUS Clients** from the left pane and verify that the *SecOnion Client* appears in the right pane and that it is enabled.



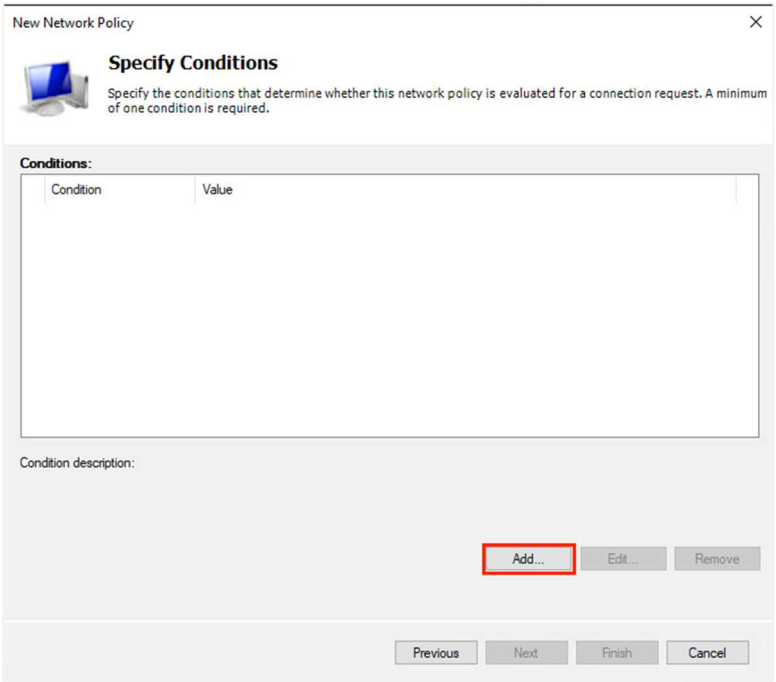
9. Now it is time to create a new network policy. In the left pane, expand **Policies**, right-click on **Network Policies** and select **New.**



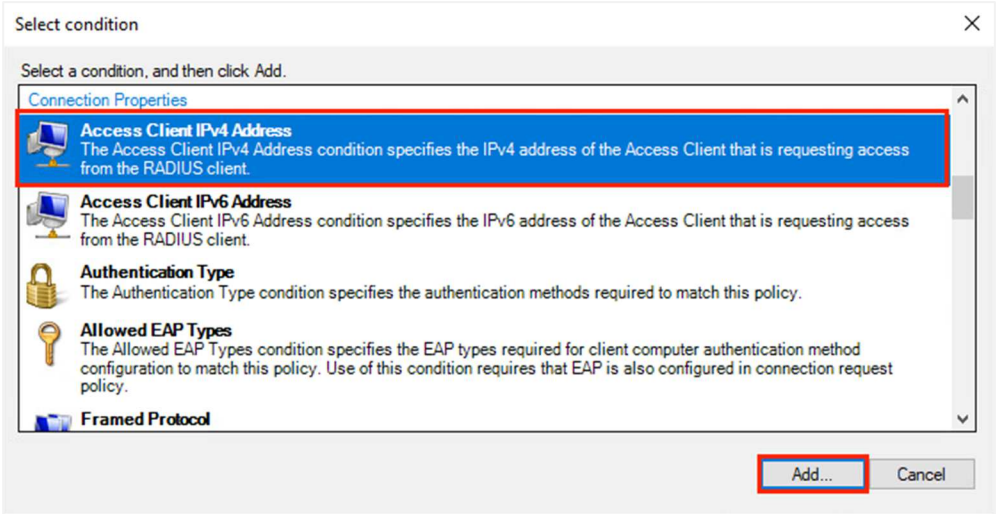
10. In the *New Network Policy* window, type `Remote Access RADIUS` in the *Policy name* text field and click **Next.**



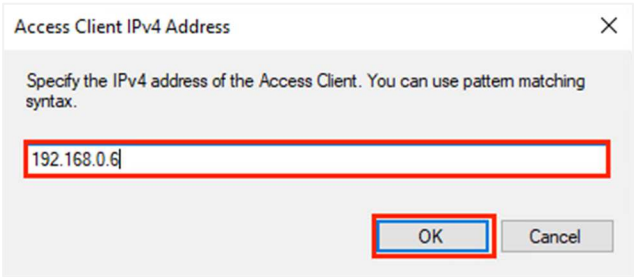
11. On the *Specify Conditions* step, click the **Add** button.



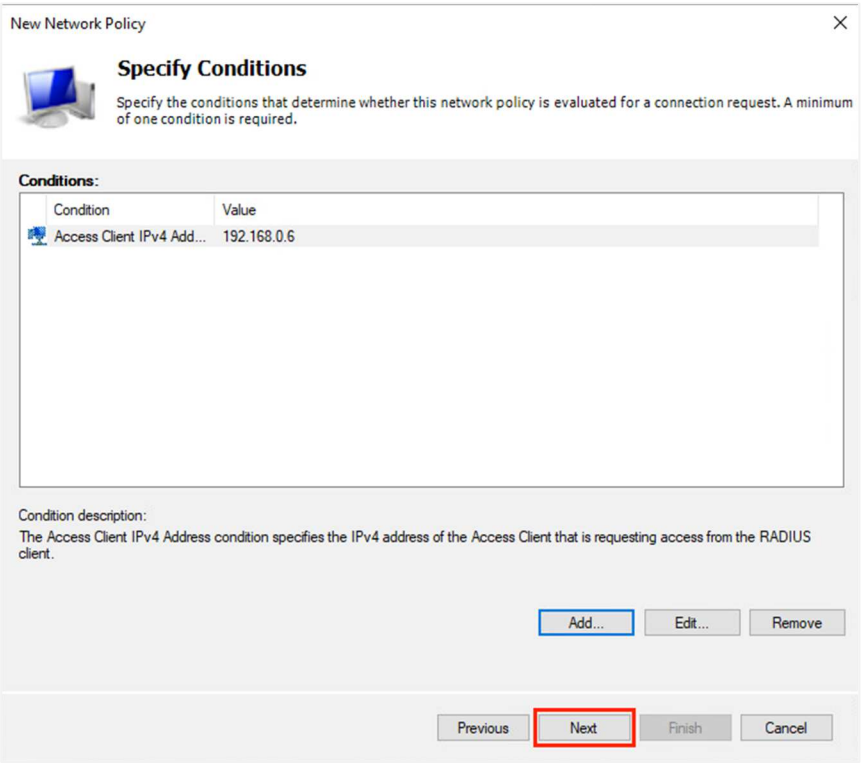
12. In the *Select condition* window, scroll down and select **Access Client IPv4 Address** and click **Add.**



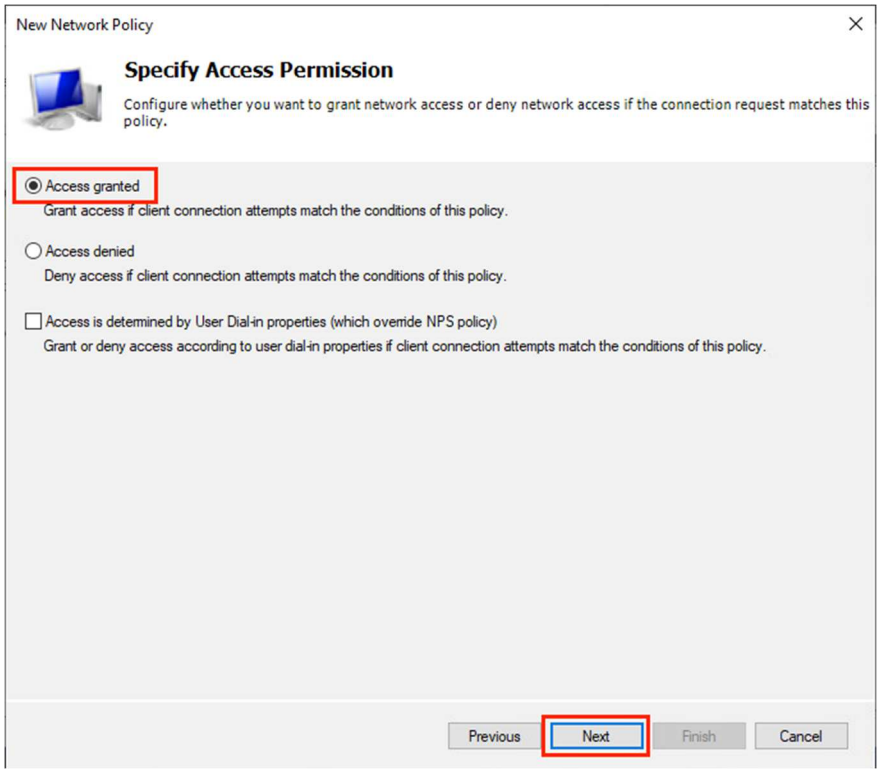
13. When prompted for an *IPv4* address, type `192.168.0.6` and click **OK.**



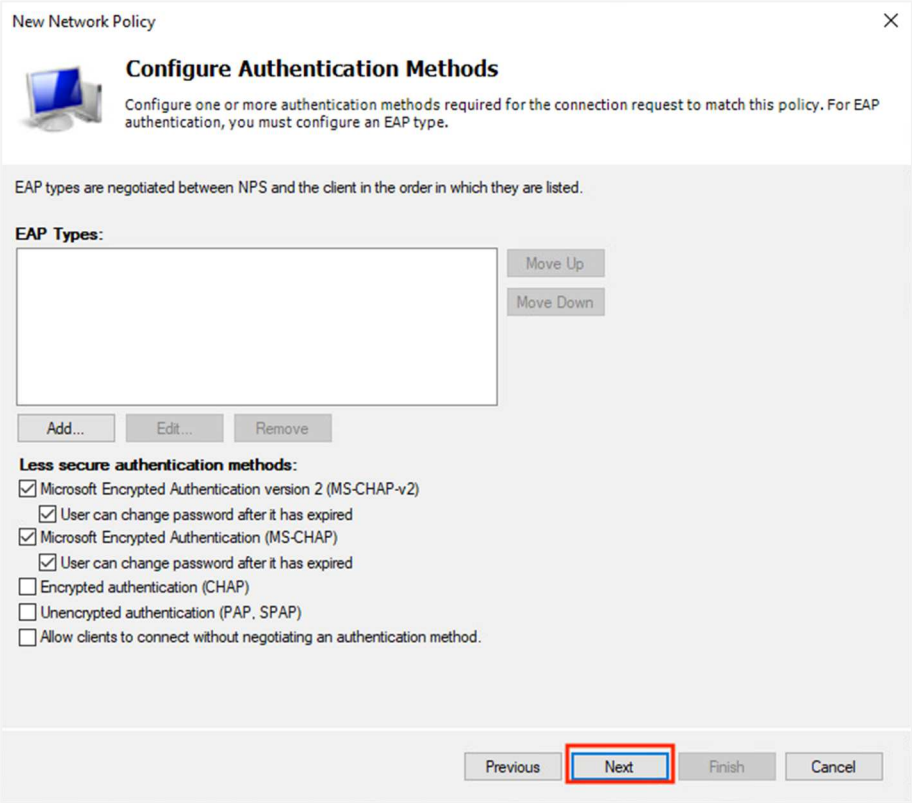
14. Back on the *Specify Conditions* step, ensure that the new condition is listed and click **Next.**



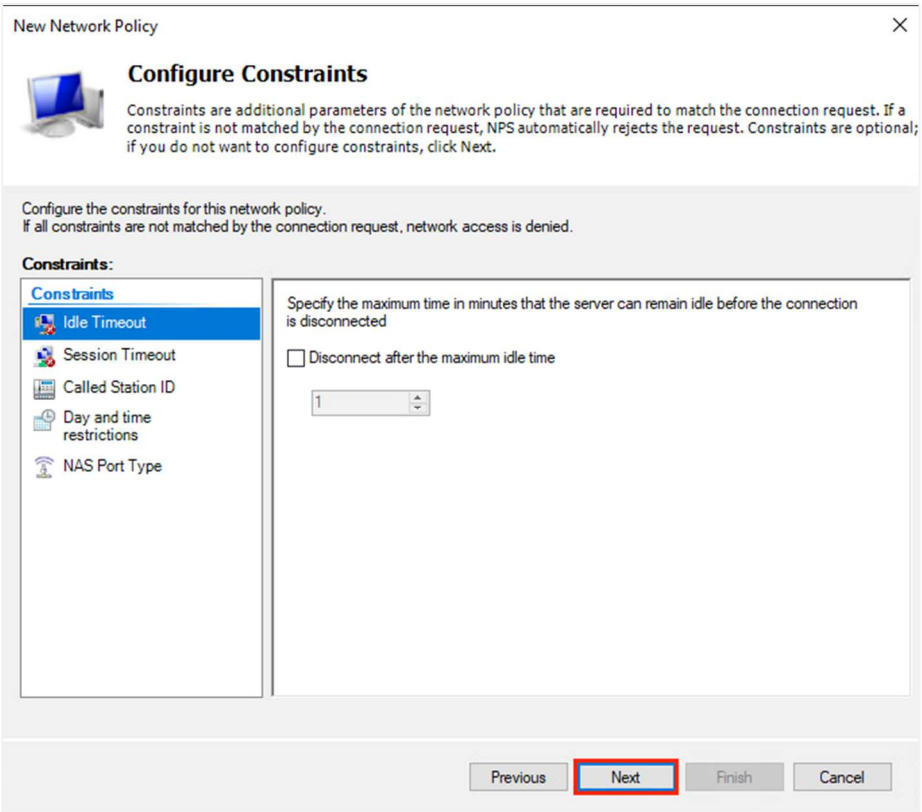
15. On the *Specify Access Permission* step, leave **Access granted** selected and click **Next.**



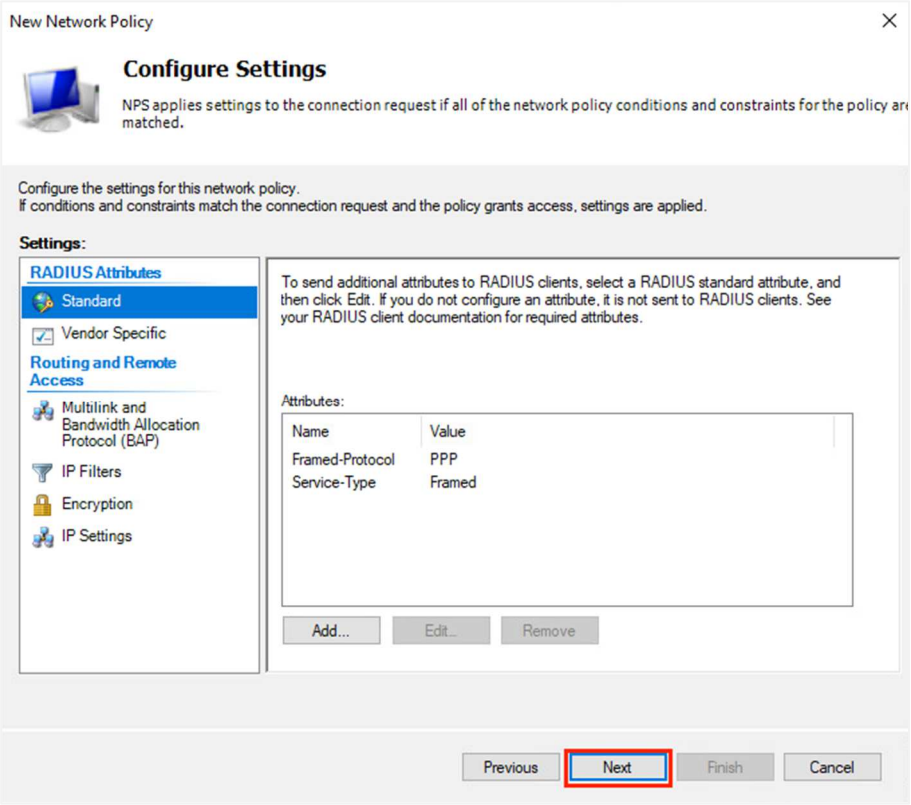
16. On the *Configure Authentication Methods* step, leave the default settings and click **Next.**



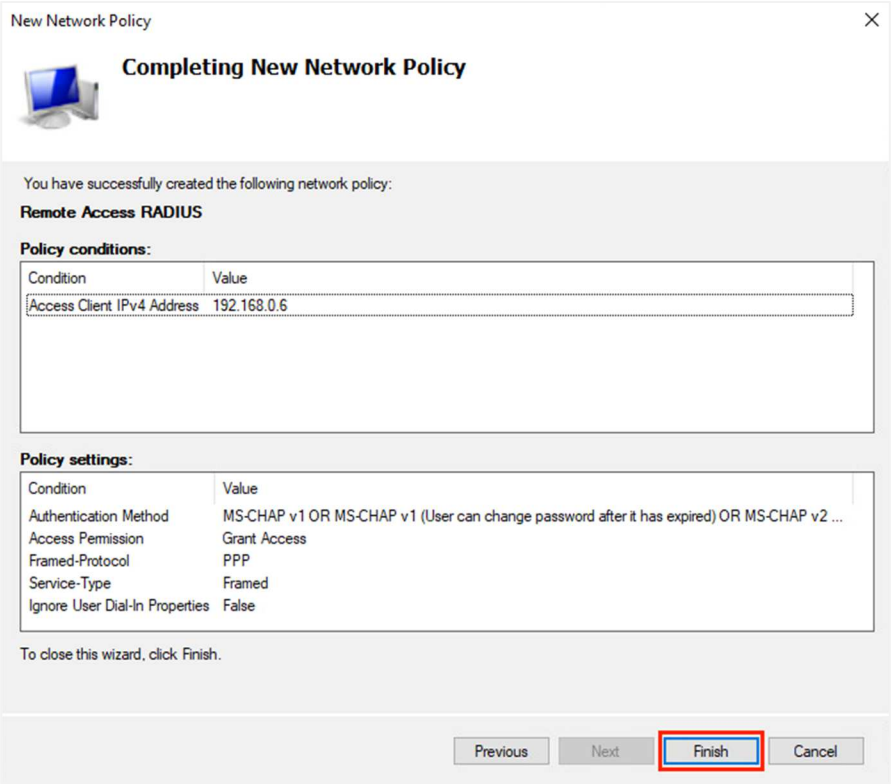
17. On the *Configure Constraints* step, leave the default settings and click **Next.**



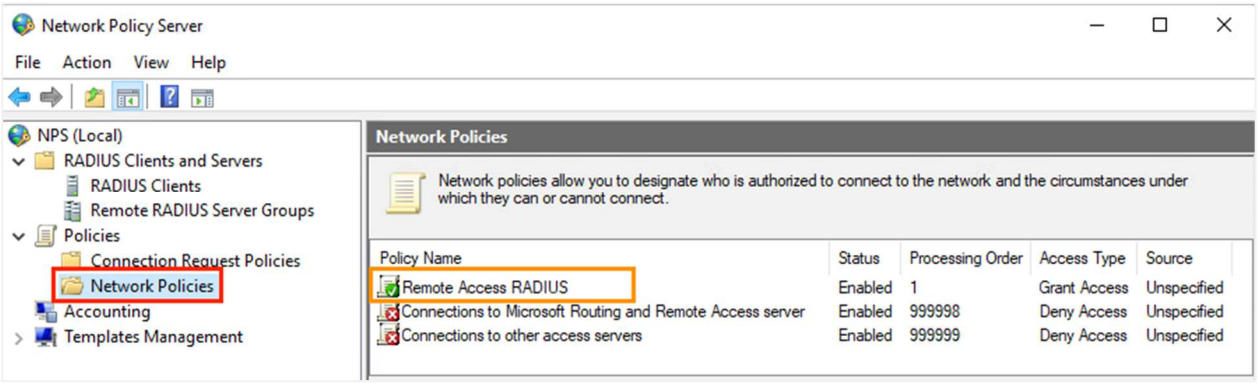
18. On the *Configure Settings* step, leave the defaults and click **Next.**



19. On the *Completing New Network Policy* step, leave the defaults and click **Finish.**



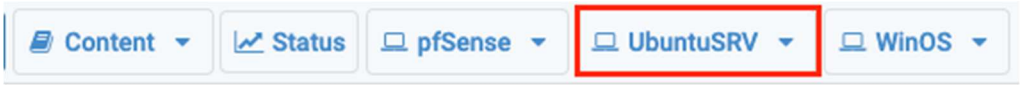
20. Back on the *Network Policy Server* window, in the left pane, click on **Network Policies** and confirm the new policy appears in the right pane.



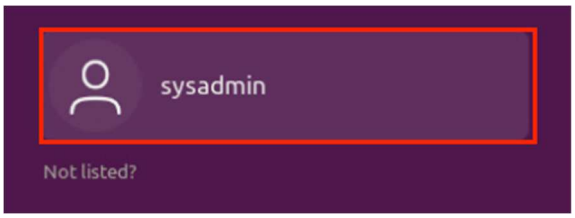
21. This section is now complete. Proceed to the next section to configure RADIUS on the Ubuntu system.

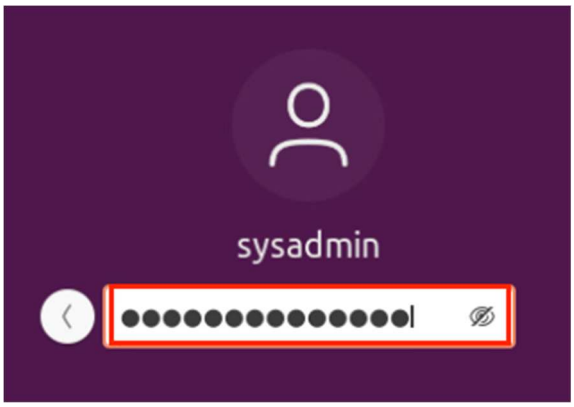
2 FreeRADIUS on Ubuntu

1. Launch the **UbuntuSRV** virtual machine to access the graphical login screen.



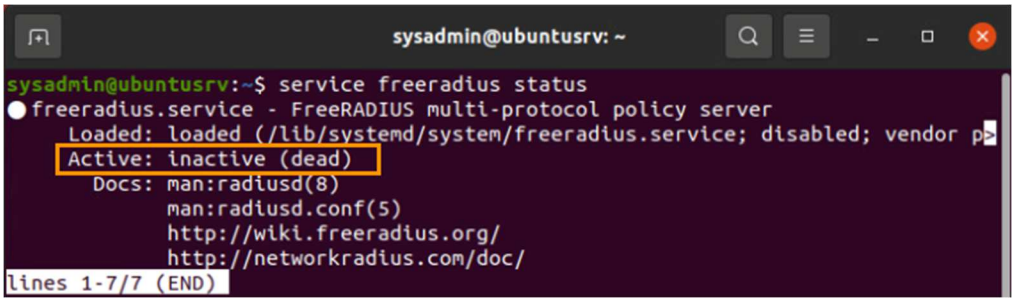
2. Log in as `sysadmin` using the password `NDGlabpass123!`.





3. Click the **Terminal** icon in the dock to start *Terminal***.**

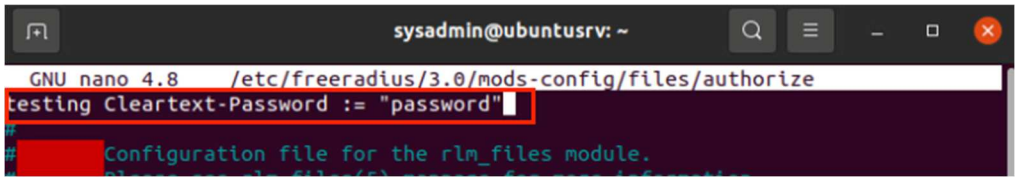
4. In the new *Terminal* window, type the command `service freeradius status` to check the status of the service. It should say *inactive*. Press **q** on the keyboard to get out of the status check.



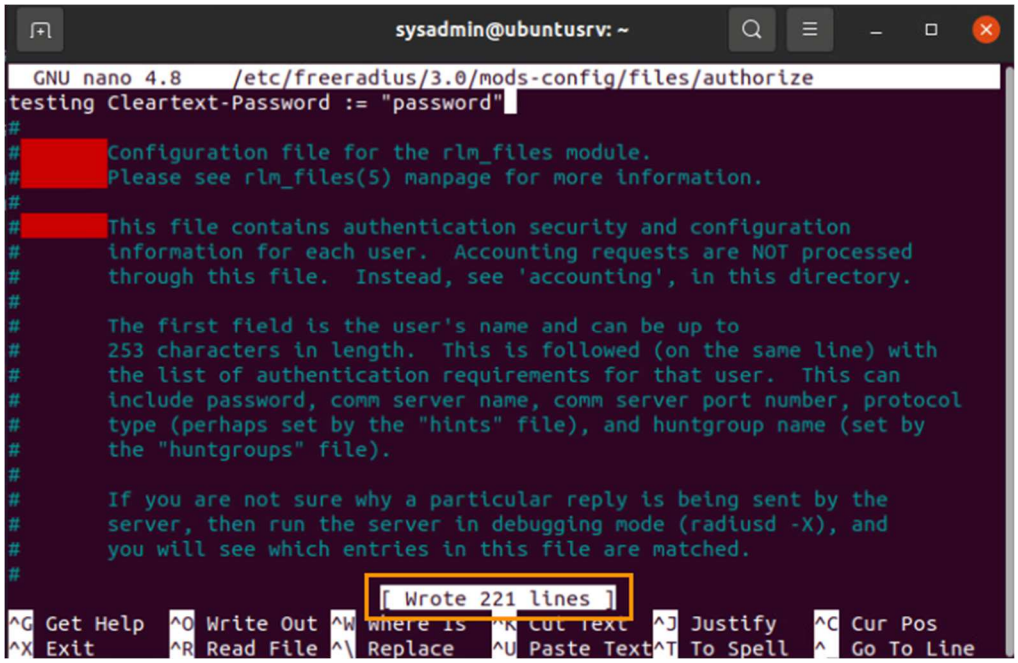
5. We are going to create a test account to locally test the RADIUS server. Type the command below in the *Terminal*. Enter `NDGlabpass123!` when prompted for a password.

```
sudo nano /etc/freeradius/3.0/mods-config/files/authorize
```

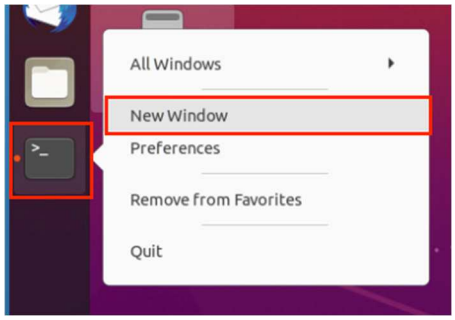
6. Once in the editor, press **Enter** to insert a newline at the top of the file. Then add the user info by typing: `testing Cleartext-Password := “password”`



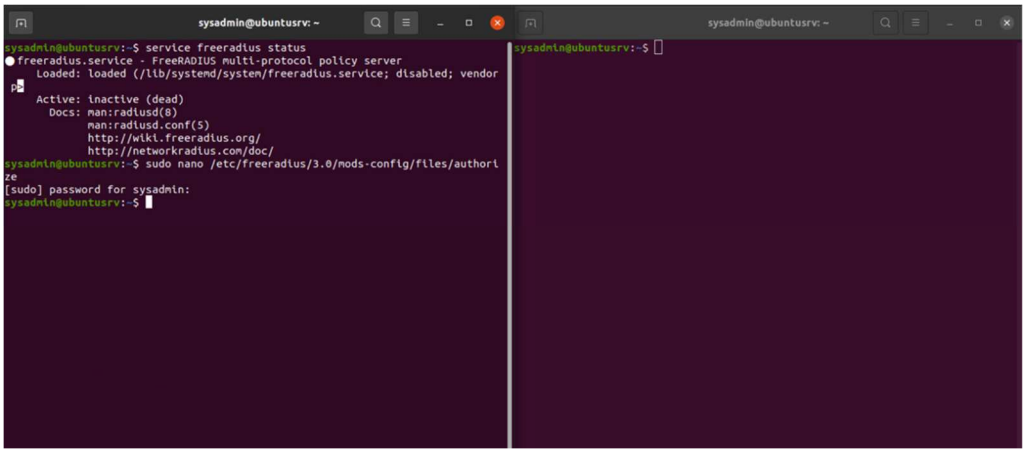
7. Then, press **Ctrl** + **S** to save the file and **Ctrl** + **X** to quit the editor.



8. Start a new *Terminal* by right-clicking the **Terminal** icon in the dock and selecting **New Window**.

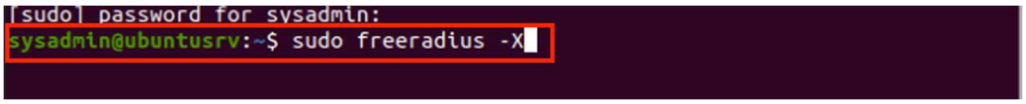


9. Rearrange the two windows to make them side-by-side.



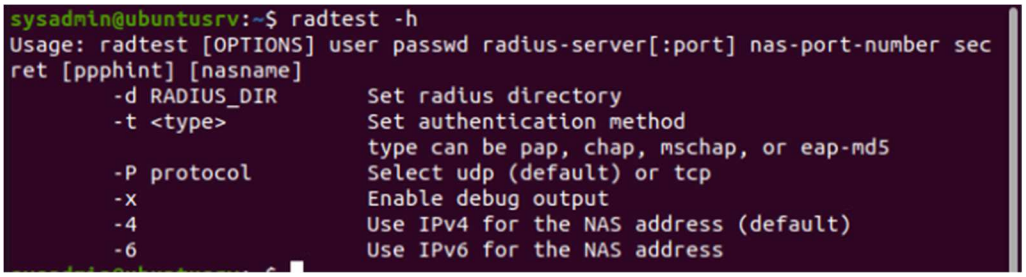
10. RADIUS is mostly used for *Business Wireless Access Point* authentication. It is an alternative method to Active Directory. In this lab, we are only going to test and learn the functionality of RADIUS rather than configure it.

    In the first *Terminal* window, type the command `sudo freeradius -X` and press **Enter** to run it. The -X option will run the *freeradius* service in debugging mode. Enter **NDGlabpass123!** when prompted for the password.



11. When you see the *Ready to process requests* prompt, it means the RADIUS server is up and running. And for local testing, it is bound to port **18120.**

12. Now switch to the other *Terminal* window. Enter the command `radtest -h` to bring up the help page.



13. Based on the help info, the command we will be using is:

```
radtest testing password 127.0.0.1 18120 testing123
```

14. If you see the result showing *Received Access-Accept*, it means the RADIUS server is working.

```
sysadmin@ubuntusrv:~$ radtest testing password 127.0.0.1 18120 testing123
Sent Access-Request Id 235 from 0.0.0.0:54668 to 127.0.0.1:1812 length 77
        User-Name = "testing"
        User-Password = "password"
        NAS-IP-Address = 172.16.1.10
        NAS-Port = 18120
        Message-Authenticator = 0x00
        Cleartext-Password = "password"
Received Access-Accept Id 235 from 127.0.0.1:1812 to 127.0.0.1:54668 length 2
0
sysadmin@ubuntusrv:~$
```

15. And you should see the response scrolling in the other window.

```
(0)     [pap] = ok
(0)   } # Auth-Type PAP = ok
(0) # Executing section post-auth from file /etc/freeradius/3.0/sites-enabled
/default
(0)   post-auth {
(0)     if (session-state:User-Name && reply:User-Name && request:User-Name &
& (reply:User-Name == request:User-Name)) {
(0)     if (session-state:User-Name && reply:User-Name && request:User-Name &
& (reply:User-Name == request:User-Name))  -> FALSE
(0)     update {
(0)       No attributes updated for RHS &session-state:
(0)     } # update = noop
(0)     [exec] = noop
(0)     policy remove_reply_message_if_eap {
(0)       if (&reply:EAP-Message && &reply:Reply-Message) {
(0)       if (&reply:EAP-Message && &reply:Reply-Message)  -> FALSE
(0)       else {
(0)         [noop] = noop
(0)       } # else = noop
(0)     } # policy remove_reply_message_if_eap = noop
(0)   } # post-auth = noop
(0) Sent Access-Accept Id 235 from 127.0.0.1:1812 to 127.0.0.1:54668 length 0
(0) Finished request
Waking up in 4.9 seconds.
(0) Cleaning up request packet ID 235 with timestamp +532
Ready to process requests
```

16. The lab is now complete; you may end your reservation.