

### 3.3.11 Lab - Using Windows PowerShell



This lab has been updated for use on NETLAB+.  
[www.netdevgroup.com](http://www.netdevgroup.com)

#### Objectives

The objective of the lab is to explore some of the functions of PowerShell.

**Part 1: Access PowerShell console.**

**Part 2: Explore Command Prompt and PowerShell commands.**

**Part 3: Explore cmdlets.**

**Part 4: Explore the netstat command using PowerShell.**

**Part 5: Empty recycle bin using PowerShell.**

#### Background / Scenario

PowerShell is a powerful automation tool. It is both a command console and a scripting language. In this lab, you will use the console to execute some of the commands that are available in both the command prompt and PowerShell. PowerShell also has functions that can create scripts to automate tasks and work together with the Windows Operating System.

#### Instructions

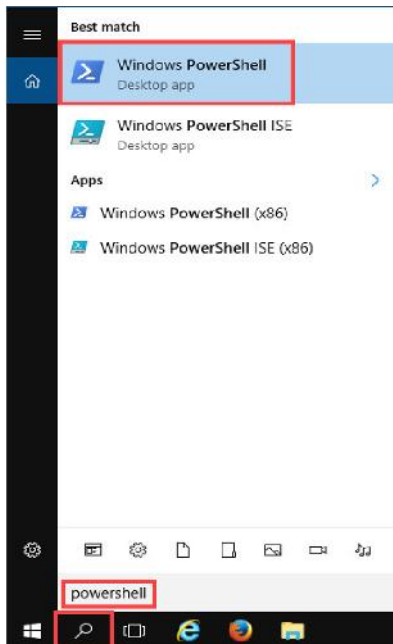
##### Part 1: Access PowerShell console.

- Access the **WinClient** machine. Unlock the machine by clicking on the drop-down arrow for that specific machine's tab and select **Send CTRL+ALT+DEL**.
- Login as the **Administrator** using **cyberops** as the password

### 3.3.11 Lab - Using Windows PowerShell

---

- c. Click on the **Search Windows** button. Type **powershell** and select **Windows PowerShell** to open.



- d. Click on the **Search Windows** button once more. Search and select **command prompt** to open.

## Part 2: Explore Command Prompt and PowerShell commands.

- a. Enter **dir** at the prompt in both the PowerShell and Command Prompt windows.  
What are the outputs to the dir command?
- 
- b. Try another command that you have used in the command prompt, such as **ping**, **cd**, and **ipconfig**.  
What are the results?
- 

## Part 3: Explore cmdlets.

- a. PowerShell commands, cmdlets, are constructed in the form of *verb-noun* string. To identify the PowerShell command to list the subdirectories and files in a directory, enter **Get-Alias dir** at the PowerShell prompt.

```
PS C:\Users\CyberOpsUser> Get-Alias dir
```

CommandType	Name	Version	Source
Alias	dir -> Get-ChildItem		

What is the PowerShell command for **dir**?

---

- b. For more detailed information about cmdlets, perform an internet search for **Microsoft PowerShell cmdlets** with an internet accessible machine.

- c. Close the Command Prompt window when done.

## Part 4: Explore the netstat command using PowerShell.

- a. At the PowerShell prompt, enter `netstat -h` to see the options available for the **netstat** command.

```
PS C:\Users\CyberOpsUser> netstat -h
```

Displays protocol statistics and current TCP/IP network connections.

```
NETSTAT [-a] [-b] [-e] [-f] [-n] [-o] [-p proto] [-r] [-s] [-x] [-t] [interval]
```

-a Displays all connections and listening ports.

-b Displays the executable involved in creating each connection or listening port. In some cases well-known executables host multiple independent components, and in these cases the sequence of components involved in creating the connection or listening port is displayed. In this case the executable name is in [] at the bottom, on top is the component it called, and so forth until TCP/IP was reached. Note that this option can be time-consuming and will fail unless you have sufficient permissions.

<some output omitted>

- b. To display the routing table with the active routes, enter `netstat -r` at the prompt.

```
PS C:\Users\CyberOpsUser> netstat -r
```

#### Interface List

```
4...00 50 56 82 da 48 .....vmxnet3 Ethernet Adapter
5...02 00 4c 4f 4f 50 .....Npcap Loopback Adapter
1.....Software Loopback Interface 1
4...00 00 00 00 00 00 00 e0 Microsoft ISATAP Adapter
5...00 00 00 00 00 00 00 e0 Microsoft ISATAP Adapter #3
```

#### IPv4 Route Table

##### Active Routes:

Network	Destination	Netmask	Gateway	Interface	Metric
	127.0.0.0	255.0.0.0	On-link	127.0.0.1	331
	127.0.0.1	255.255.255.255	On-link	127.0.0.1	331
	127.255.255.255	255.255.255.255	On-link	127.0.0.1	331
	169.254.0.0	255.255.0.0	On-link	169.254.12.163	281
	169.254.181.151	255.255.255.255	On-link	169.254.12.163	281
	169.254.255.255	255.255.255.255	On-link	169.254.12.163	281
	192.168.1.0	255.255.255.0	On-link	192.168.0.12	271
	192.168.1.5	255.255.255.255	On-link	192.168.0.12	271
	192.168.1.255	255.255.255.255	On-link	192.168.0.12	271
	224.0.0.0	240.0.0.0	On-link	127.0.0.1	331
	224.0.0.0	240.0.0.0	On-link	169.254.12.163	281
	224.0.0.0	240.0.0.0	On-link	192.168.0.12	271
	255.255.255.255	255.255.255.255	On-link	127.0.0.1	331
	255.255.255.255	255.255.255.255	On-link	169.254.12.163	281
	255.255.255.255	255.255.255.255	On-link	192.168.0.12	271

### 3.3.11 Lab - Using Windows PowerShell

Persistent Routes:

None

IPv6 Route Table

=====

Active Routes:

If	Metric	Network	Destination	Gateway
1	331	::1/128		On-link
3	281	fe80::/64		On-link
10	281	fe80::/64		On-link
10	281	fe80::408b:14a4:7b64:b597/128		On-link
3	281	fe80::dd67:9e98:9ce0:51e/128		On-link
1	331	ff00::/8		On-link
3	281	ff00::/8		On-link
10	281	ff00::/8		On-link

=====

Persistent Routes:

None

- c. The **netstat** command can also display the processes associated with active TCP connections. Enter the **netstat -abno** at the prompt.

```
PS C:\Windows\system32> netstat -abno
```

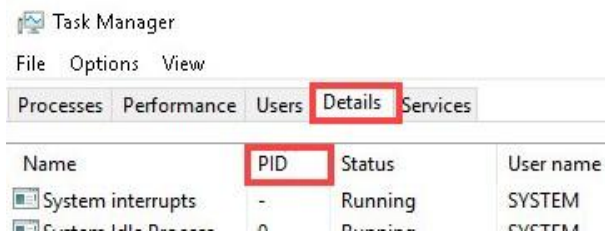
Active Connections

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	732
RpcSs				
[svchost.exe]				
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
Can not obtain ownership information				
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	444
Can not obtain ownership information				
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING	440
Schedule				
[svchost.exe]				
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING	304
EventLog				
[svchost.exe]				
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING	1856
[spoolsv.exe]				
TCP	0.0.0.0:49668	0.0.0.0:0	LISTENING	544

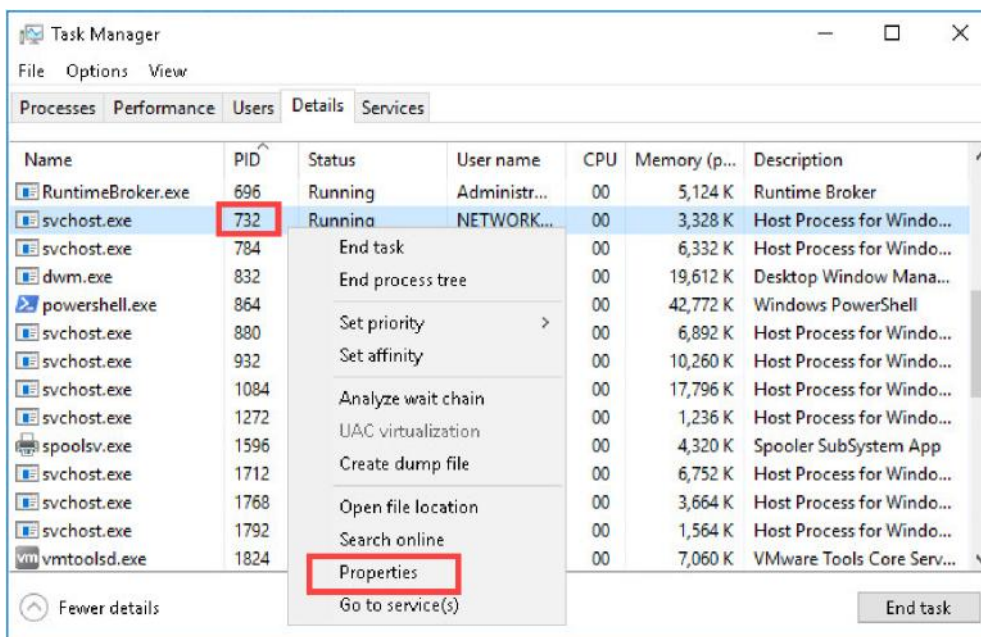
<some output omitted>

- d. Right-click on the taskbar and select **Task Manager**. Navigate to the **Details** tab. Click the **PID** heading so the PID are in order.

### 3.3.11 Lab - Using Windows PowerShell



- e. Select one of the PIDs from the results of `netstat -abno`. PID 756 is used in this example.
- f. Locate the selected PID in the Task Manager. Right-click the selected PID and select **Properties** for more information.



What information can you get from the Details tab and the Properties dialog box for your selected PID?

- g. Close the **Properties** window and **Task Manager**.

## Part 5: Empty recycle bin using PowerShell.

PowerShell commands can simplify management of a large computer network. For example, if you wanted to implement a new security solution on all servers in the network you could use a PowerShell command or script to implement and verify that the services are running. You can also run PowerShell commands to simplify actions that would take multiple steps to execute using Windows graphical desktop tools.

- a. Open the **Recycle Bin**. Verify that there are items that can be deleted permanently from your PC. If not, restore those files.
- b. If there are no files in the Recycle Bin, create a few files, such as text file using Notepad, and place them into the Recycle Bin.
- c. In the PowerShell console, enter `clear-recyclebin` at the prompt. When prompted, press "Y".

```
PS C:\Users\CyberOpsUser> clear-recyclebin
```

Confirm

### 3.3.11 Lab - Using Windows PowerShell

---

```
Are you sure you want to perform this action?  
Performing the operation "Clear-RecycleBin" on target "All of the contents of the  
Recycle Bin".  
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is  
"Y"): y
```

What happened to the files in the Recycle Bin?

---

### Reflection Question

PowerShell was developed for task automation and configuration management. Using the internet, research commands that you could use to simplify your tasks as a security analyst. Record your findings.

---

---

---