## Listeners and Stagers

* *Uselistener http* : to select a basic windows/meterpreter/reverse\_http type payload listener.
* *DefaultDelay* attempts to simulate more legitimate HTTP traffic by setting the wait interval callback time from the compromised host to the listener.
* *DefaultJitter* makes the traffic seem less programmatically generated by setting DefaultDelay to a random offset.
* *KillDate* will self-terminate the listeners on all compromised hosts on the specified date. This is especially useful when performing cleanup after a penetration test.
* Once the options are set, we can start the listener with the execute command
* return to the main listener menu with back
* we can list all available stagers with usestager followed by Space and double Tab.
* usestager windows/launcher\_bat can be used to set a stager.
* Then set the listener to which you want the stager to connect to by “set Listener http”
* Excute. Payload will be generated.

## The Empire Agent

* agents command can be used to list all agents.
* Interact <agent-name> communicate to the agent.
* Note that the command does not return immediately. This delay is caused by the DefaultDelay parameter, which is currently set to the default value of five seconds
* The help command (Listing 832) shows all available commands, such as upload, download, and screenshot, which are self-explanatory. In addition, we can use shell to execute a command and spawn to create an additional agent on the same host.

## PowerShell Modules

* The power of Empire agents lies in the various modules offered by the framework. We can list all available modules by running usemodule followed by a Space and double Tab.

### Situational Awareness

* While there are many methods and commands for performing network enumeration, the primary focus of this category is on local client and Active Directory enumeration.
* The Active Directory enumeration modules are found in the network subcategory with a prefix of PowerView. This is a reference to @harmj0y’s original Veil-PowerView724 project.
* If the NeedsAdmin field is set to “True”, the script requires local Administrator permissions.
* If the OpsecSafe field is set to “True”, the script will avoid leaving behind indicators of compromise, such as temporary disk files or new user accounts.
* This stealth-driven approach has a greater likelihood of evading endpoint protection mechanisms.
* The MinLanguageVersion field describes the minimum version of PowerShell required to execute the script.
* This is especially relevant when working with Windows 7 or Windows Server 2008 R2 targets as they ship with PowerShell version 2.
* Background tells us if the module executes in the background without visibility for the victim, while OutputExtension tells us the output format if the module returns output to a file.
* The Bloodhound module is especially noteworthy. It automates much of PowerView’s functionality, collecting all computers, users, and groups in the domain as well as all currently logged-in users.
* The output is stored in CSV files suitable for use with the backend BloodHound application,725 which uses graph theory726 to highlight often-overlooked and highly complex attack paths in an Active Directory environment.

#### Credentials and Privilege Escalation

* The privesc category contains privilege escalation modules. One of the more interesting modules in this group is powerup/allchecks.727 It uses several techniques based on misconfigurations such as unquoted service paths, improper permissions on service executables, and much more
* “usemodule powershell/privesc/powerup/allchecks”
* The bypassuac\_fodhelper module is quite useful if we have access to a local administrator account. Depending on the local Windows version, this module can bypass UAC and launch a high-integrity PowerShell Empire agent.
* Once we have a high-integrity session, we can perform actions that require local administrator or SYSTEM rights, such as executing mimikatz to dump cached credentials.

### Lateral Movement