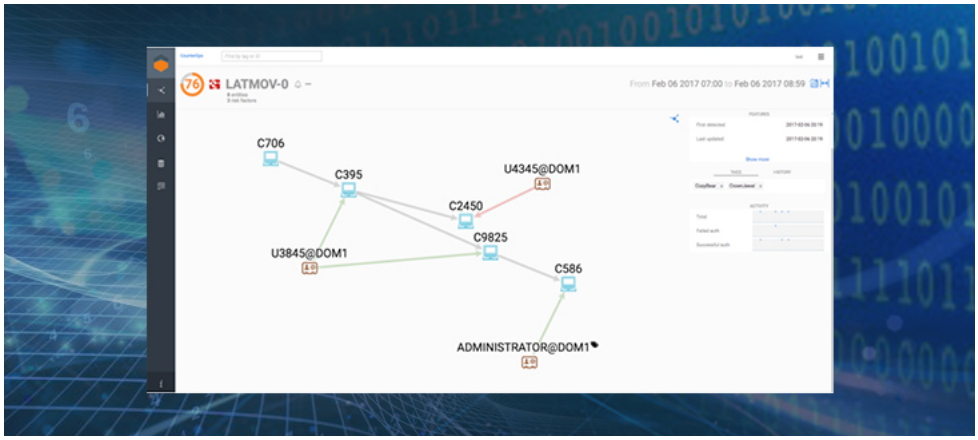
FINDING EVIL WHEN HUNTING FOR LATERAL MOVEMENT

(/blog/)

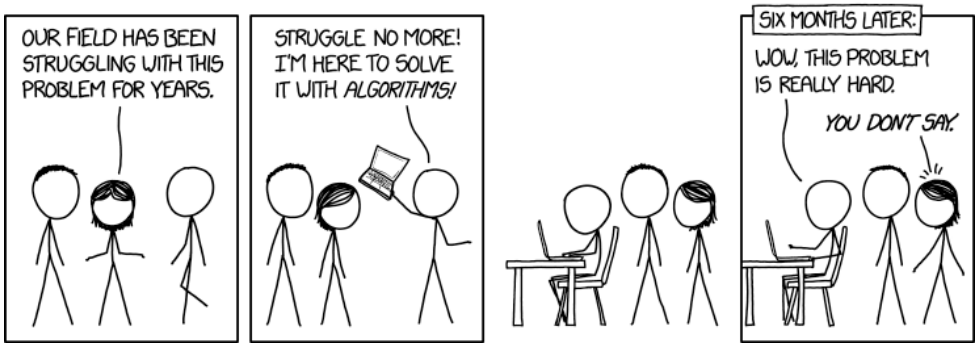


(/finding-evil-when-hunting-for-lateral-movement/)

August 16, 2017 by Ryan Nolette ()

FINDING EVIL WHEN HUNTING FOR LATERAL MOVEMENT

Lateral movement is a critical step that attackers use when targeting your network. In the last Hunter's Den (https://sqrrl.com/threat-hunting-lateral-movement-pt-2-infection/) post we covered how attackers lay the groundwork for lateral movement. Now that we know what tactics to look for, let's get to hunting.



(https://sqrrl.com/media/here_to_help.png)

Image source: Randall Munroe, XKCD (https://xkcd.com/1831/), 2017 #PleaseDontSueMe

Browse by Topic

Threat Hunting

Subscribe to Blog

Email Address

SUBSCRIBE

Indicator Searches

The type of dataset that you use to hunt for lateral movement depends on what you are hunting for and, by extension, what your hypothesis is.

For identifying use of remote access protocols, you will want to focus primarily on network session metadata, including:

- Netflow ("flow" data in general)
- Firewall logs (should log allowed / accepted packets)
- Bro Conn log

For identifying User Access Control (UAC) events, you will want to focus on authentication logs, including:

- Active Directory logs/Windows Security Event logs
  - EventID
    - 528 or 4624 is indicative of a successful logon
    - 529 or 4625 is a failed logon
    - 552 and 4648 are indicative of an attacker attempting to use the runas command or authenticate against a remote host as an alternative user, #privilegeEscalation.
    - 602 and 4698 are indicative of a scheduled task creation
    - 601 and 4697 are indicative of a service creation
  - Account Features
    - Service account
    - Interactive login
- Linux Security Event logs
- OSX Security Event logs
- Multi-Factor Authentication (MFA) logs
- Additional UAC applications if exists

Techniques to Use:

After having developed the right hypotheses and chosen the necessary datasets, a hunter must still know what techniques to use to investigate a hypothesis. Here we will survey 3 types of techniques that you can use to investigate the above.

Indicator Search

Featured Posts

The Nuts and Bolts of Detecting DNS Tunneling
By Sqrrl Team
(/the-nuts-and-bolts-of-detecting-dns-tunneling/)

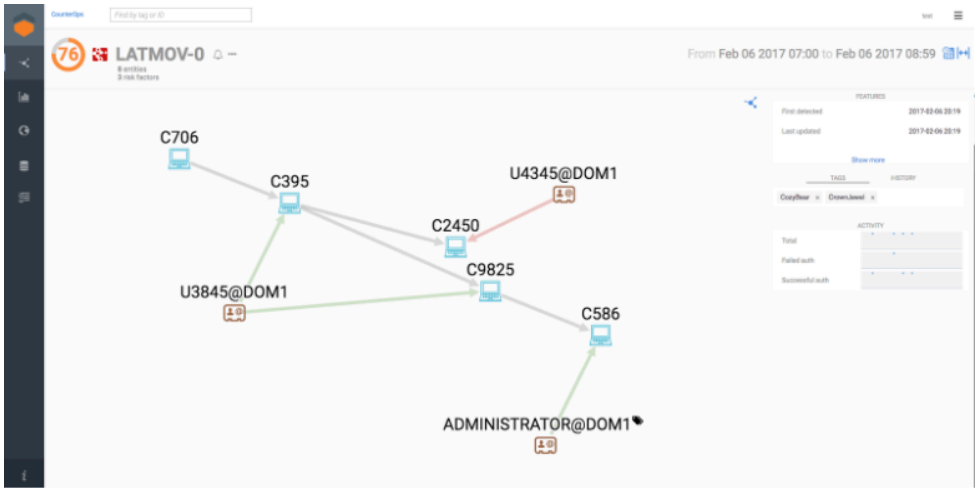
Scoping Attacks By Following Attacker Breadcrumbs
By Chris Sanders
(/scoping-attacks-by-following-attacker-breadcrumbs/)

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/)



(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-3.09.06-PM.png)

As in all cases of using indicators in hunting, the value of this approach will be impacted by the value of the indicator. Locally sourced indicators will generally provide a high value because they tend to be timely and relevant to the network or systems you might be trying to protect. These types of indicators can be gathered from previous incidents or by internal threat intelligence teams.

It's also important to remember that it is relatively easy for attackers to change the remote infrastructure that they use to conduct attacks; if you are using indicators to hunt, then you should be aware that the indicators may no longer be relevant to a particular attacker or attack tool.

Some common network session indicators to search for include:

- IP address
- Port
- Top Level Domain (TLD)
- URI
- Unique strings in the connection

Some common host based indicators to search for include:

- File hashes
- Filenames
- Registry modification
- Process injection

Virustotal

Resources

Whitepaper
The Who, What, Where, When, Why and How of Effective Threat Hunting
(http://info.sqrrl.com/sqrrl-sans-hunting-white-paper)

Whitepaper
Technical Product Guide: Nuts and Bolts of Sqrrl's Threat Hunting Platform
(http://info.sqrrl.com/sqrrl-product-paper-0)

Webinar
IBM QRadar Integration: Proactive Incident Detection and Investigations
(http://info.sqrrl.com/sqrrl-ibm-threat-hunting-for-qradar-users)

Webinar
HPE ArcSight Integration: Finding Incidents with Hunting Techniques
(http://info.sqrrl.com/sqrrl-hpe-threat-hunting-for-arcsight-users)

Webinar
Carbon Black Integration: Threat Hunting from the Network to Endpoint
(http://info.sqrrl.com/july-2016-sqrrl-carbon-black-webinar)

Report
The Hunter Strikes Back: The SANS 2017 Threat Hunting Survey
(http://info.sqrrl.com/sans-2017-threat-hunting-report)

You can submit a file hash or URL to Virustotal to either have it scanned or return existing results if the file/URL has been submitted before.

For example: My file built in Metasploit has previously been built and submitted to VT. The link to the results is here (https://www.virustotal.com/en/file/f264700a36a21f96851bddd1a488da66b30c1693544cba5ed2c83 497886(

And comes with the additional high level information of:

Key	Value	Notes
SHA256	f264700a36a21f96851bddd1a488da66b30c1693544cba5ed2c83562ba62c212	This is the hash value that identifies the file uploaded.
File name	badguy3.exe	This is the name of the file that was uploaded. In addition to this file name there will also be a list of other name the same hash has been known by. (located in the additional information under file names)
Detection ratio	48 / 61	This is the ratio of AV engines that detect the file in comparison to all AV engines that scanned it.
Analysis date	2017-06-19 15:27:17 UTC (1 minute ago)	This is the timestamp of the last analyzed date of the file.

(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-3.10.25-PM.png)

VT also contains a bunch of interesting information about the file but you can just use the link above to explore that additional information.

Detecting PsExec Activity with Snort

Snort can be used to detect malicious SMB activity. Snort is an open-source intrusion detection and and prevention system, and designed to detect attacks via a pattern-matching signature.

Below is an example of a Snort rule designed to detect the use of PsExec (Emerging Threats, 2011 http://doc.emergingthreats.net/2010781 (http://doc.emergingthreats.net/2010781)):

alert tcp any any -> $HOME_NET [139,445] (msg:"ET POLICY PsExec? service

created"; flow:to_server, established; content:"|5c 00 50 00 53 00 45 00 58 00 45 00 53

00 56 00 43 00 2e 00 45 00 58 00 45|"; reference:url, xinn.org/Snort-psexec.html;

reference:url, doc.emergingthreats.net/2010781; classtype:suspicious-filename-detect;

sid:201781; rev:2;)

Detecting PsExec Activity Using Bro

PsExec is a Windows administration tool used connect to different systems on a network via SMB, using administrative credentials. SMB is legitimately used to provide file sharing

Watch Overview

(https://www.youtube.com/watch?v=Vl_zLBc4KQM&t&width=640&heigh

functionality, however; misconfigurations can allow malware to propagate throughout a network. Combine PsExec with the password theft abilities of mimikatz and you have an equation for lateral movement.

One technique to detect PsExec activity with Bro is by using custom Bro scripts looking for PsExec's use of the C$, ADMIN$, and/or IPC$ shares. These shares added notice messages of "Potentially Malicious Use of an Administrative Share" in the Bro Notice log. The use of PsExec creates an executable named PSEXESVC.exe on the target system.

Modified code for my usage. Code is originally from https://www.sans.org/reading-room/whitepapers/detection/detecting-malicious-smb-activity-bro-37472 (https://www.sans.org/reading-room/whitepapers/detection/detecting-malicious-smb-activity-bro-37472)

```
@load base/frameworks/files

@load base/frameworks/notice

@load policy/protocols/smb


export {  redef enum Notice::Type += {  Match  };

	global isTrusted = T;

	global trustedIPs: set[addr] = {192.168.1.1,192.168.1.10} &redef;

	function hostAdminCheck(sourceip : addr) : bool

	{

		if (sourceip !in trustedIPs)

		{

			return F;

		}

		else

		{

			return T;

		}

	}
```

```
event smb2_tree_connect_request(c : connection, hdr : SMB2::Header, path : string)
{
    isTrusted = hostAdminCheck(c$id$orig_h);
    if (isTrusted == F) {
        if (“IPC$” in path || “ADMIN$” in path || “C$” in path)
        {
            NOTICE([$note=Match, $msg=fmt(“Potentially Malicious Use of an Administrative Share”), $sub=fmt(“%s”,path), $conn=c]);
        }
    }
}
event smb1_tree_connect_andx_request(c : connection, hdr : SMB1::Header, path : string, service : string)
{
    isTrusted = hostAdminCheck(c$id$orig_h);
    if (isTrusted ==F) {
        if (“IPC$” in path || “ADMIN$” in path || “C$” in path)
        {
            NOTICE([$note=Match, $msg=fmt(“Potentially Malicious Use of an Administrative Share”), $sub=fmt (“%s”,path), $conn=c]);
        }
    }
}
```

Why does this detect the usage of PsExec? Metasploit contains a modified version of PsExec.exe that doesn't have a source file hash as it is run from the attacker system via meterpreter. To detect this PsExec usage, we depend on the Bro notice log, which verifies detection of the Metasploit PsExec module's use of C$, ADMIN$, and IPC$ shares.

Detection of PsExec traffic via a Bro network sensor

(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-3.12.31-PM.png)

In addition to remove control via SMB by PsExec, attackers will upload other binaries to the victim system or use more meterpreter modules. For example, the tool Mimikatz, which is used to dump passwords from memory, can be uploaded to a remote system via the C$, ADMIN$, and IPC$ shares. Bro has the ability to detect Mimikatz getting transferred over SMB and the ability to check its hash against VirusTotal.

If you don't dump your bro logs into a SIEM or other log aggregation platform, I suggest a simple grep command to search for PsExec usage traffic, "grep -iE "C$|ADMIN$|IPC$"".

Stacking

CounterOps Find by tag or ID

```
1 SELECT EventID AS EventID, count(*) AS counter FROM Sqrrl_WindowsEvents WHERE EventID IS NOT
  NULL GROUP BY EventID ORDER BY counter DESC
```

GO

SELECT EventID AS EventID, count(*) AS counter FROM Sqrrl_WindowsEvents WHERE EventID IS NOT NULL GROUP BY EventID ORDE

Filter

EventID	counter
4624	317
4634	317
4672	302
5320	159
4017	105
5017	104
5327	52
4326	27
5315	26
4006	26
5313	26
5326	26
5314	26
5310	26
5312	26
5311	26
5308	26

(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-3.15.07-PM.png)

Query Used: SELECT EventID AS EventID, count(*) AS counter FROM Sqrrl_WindowsEvents WHERE EventID IS NOT NULL GROUP BY EventID ORDER BY counter DESC

Stacking is a technique commonly used in many different kinds of hunts. In the case of hunting for command and control activity, a hunter will want to stack for anomalous instances of inbound or outbound traffic. The same metadata types from indicator search above can be used for stacking, including:

- EventID
- UserName
- Account Type
- Hostname

To find lateral movement, you will want to focus on either bidirectional or external to inbound connection flows. The effectiveness of using stacking is dependent on having a finely tuned input. Too little won't reveal enough and too much will flood your ability to tease out meaningful deviations. If a given result set is too large, then consider further filtering of the input data set (e.g., isolate your focus to specific internal subnets). Alternatively, change the metadata that is being stacked (e.g., change from stacking hostnames to stacking EventID and username).

Machine Learning

A more advanced technique involves using machine learning to isolate lateral movement activity. Supervised machine learning uses labeled training data to make predictions about unlabeled data. Given a set of known good and known bad examples, you can create a binary classifier capable of taking in new transactions and deciding if they look more similar to the good training set or the bad training set. After the classifier is trained, assuming you have done a good job, you can feed your network and UAC data through it and get back much smaller set of records that require analyst attention.

For more information on using machine learning for hunting, we highly recommend watching the presentation 'Practical Cyborgism: Getting Started with Machine Learning for Incident Detection' from David Bianco and Chris McCubbin (presented at BSides DC 2016) (https://youtu.be/2FvP7nwb2UE).

EXAMPLE HUNT

The example below illustrates how to use the hypotheses laid out above with the data and techniques enumerated.

Lateral Movement (Windows Environment)

(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-3.16.26-PM.png)

1. What are you looking for? (Hypothesis)	**Hypothesis:** Attackers may be attempting to move laterally in my Windows environment by leveraging PsExec. **Look for:** • Anomalies in host to host traffic leveraging the PsExec binary, service, and/or network traffic. • "C$\|ADMIN$|IPC$" shares being used in network traffic.
2. Investigation (Data)	**Datasets:** For identifying use of PsExec, you will want to focus primarily on application protocol metadata, including: • Netflow ("flow" data in general) • Active Directory logs • Windows Security Event logs • Multi-Factor Authentication (MFA) logs (if windows hosts leverage MFA)

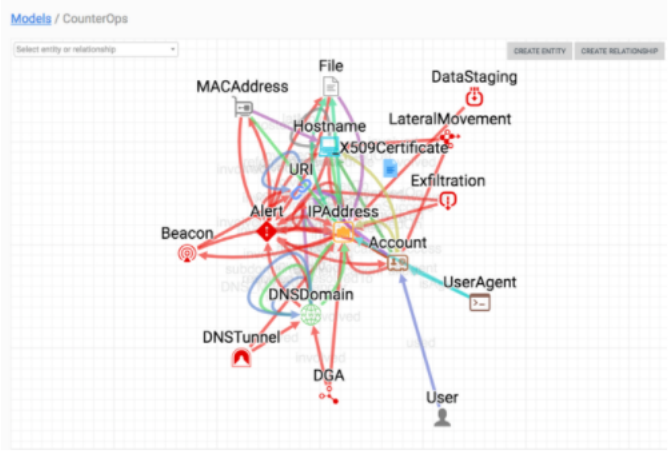
	• Additional UAC applications logs (if exists) • EDR tool logs (if exists)
3. Uncover Patterns and IOCs (Techniques)	1. Use a search (https://sqrrl.com/threat-hunting-reference-guide/#searching) to identify "Potentially Malicious Use of an Administrative Share" messages in your bro_notice log. 2. Take the output of step 1 and remove hosts as you confirm they are legitimately connecting to a destination over SMB. This should leave only unexplained SMB connections that need further analysis. 3. Take the results of step 2 and stack the data for what is useful to investigating your hypothesis 1. For example: destination IP, port used, connection duration/length, etc.
4. Inform and Enrich Analytics (Takeaways)	The destination IP addresses, path, and ports involved in the Lateral Movement activity you have discovered can be taken as IOCs and added to an indicator database in order to expand automated detection systems. You can also create packet-level signatures to trigger alerts for cases where the admin share connections you have discovered may appear again.

Always keep in mind that for each instance of a hunt, there will always be multiple different paths that a hunter can take to address a given hypothesis.

So what does this activity actually look like in a threat hunting platform like sqrrl?

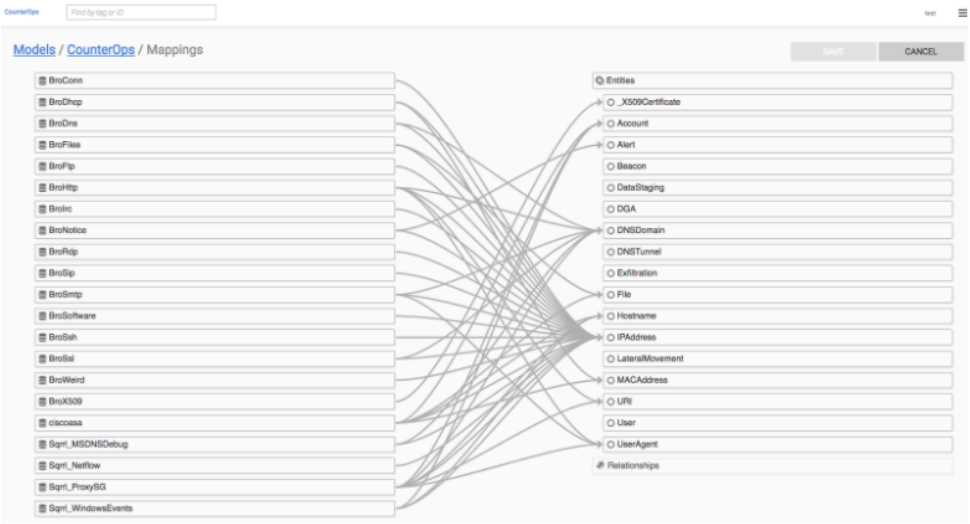
First I need to define relationships between different data sources. For example, my network data and authentication event data both contain the IP address field. This means I can create a relationship between disparate data sets as shown below.

Creating relationship between disparate data sets



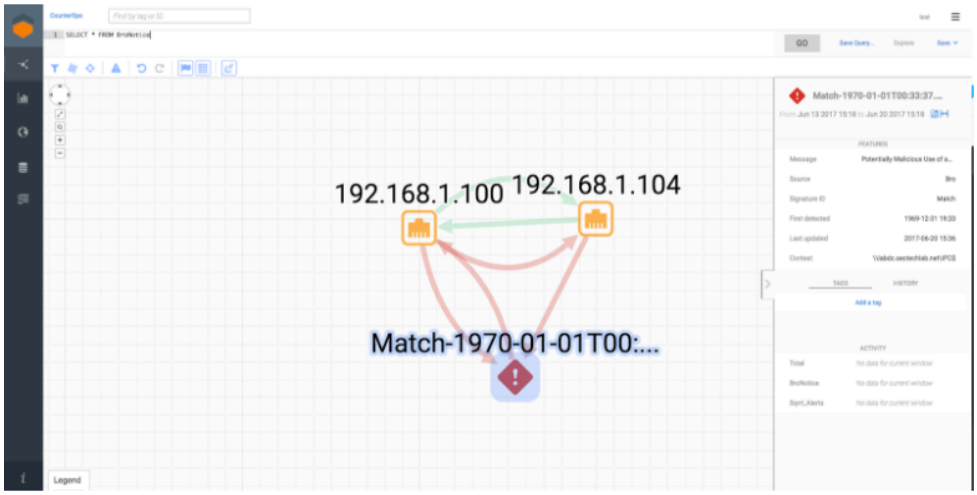
(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-3.19.23-PM.png)

The above image displays all of the relationships I have created between different entities within my data. This is a high level visualization of the relationships.



(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-5.26.51-PM.png)

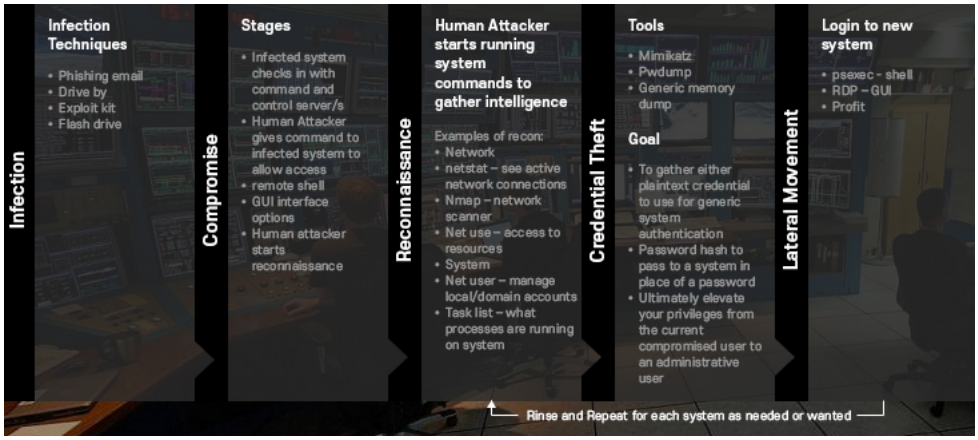
The above visualization expands on the high level visualization and illustrates all of the low level field mappings between each data source being ingested and each entity defined. This is where the actual mapping of the IP address field between authentication data and network data.



(https://sqrrl.com/media/Screen-Shot-2017-08-15-at-5.28.32-PM.png)

Finally, with all the hard work done, we are able to visualize the event of psexec being used to move from 192.168.1.100 to 192.168.1.104 as well as incorporate a bro alert for psexec to add further validation. While this image only shows the exact activity I am describing for ease of reading, this is what I expect an end result of a hunt to look like. All additional data and possible connections have been investigated and excluded from the original large data set until you are only left with the anomalous/suspicious/malicious event. I consider this a successful hunt. I would also have considered it a success if I had found nothing at all because the point of a hunt isn't to a true positive malicious event every time, but instead it is to validate a hypothesis, to answer a question with a definitive yes or no. Good luck to all and happy hunting.

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



(https://sqrrl.com/threat-hunting-lateral-movement-pt-2-infection/)

August 8, 2017 by Ryan Nolette ()

HOW ATTACKERS LAY THE GROUNDWORK FOR LATERAL MOVEMENT (HTTPS://SQRRL.COM/THREAT-HUNTING-LATERAL-MOVEMENT-PT-2-INFECTION/)

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.

READ MORE
(https://sqrrl.com/threat-hunting-lateral-movement-pt-2-infection/)

Next Post

SQRRL NEWSLETTER

Subscribe to our mailing list

Email

SUBMIT

(https://twitter.com/SqrrlData)
(https://www.facebook.com/SqrrlData)
(https://plus.google.com/116795302724746825954/posts)
(https://www.linkedin.com/company/sqrrl)

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)

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

PRODUCT (/PRODUCT/SQRRL-ENTERPRISE)

Sqrrl Enterprise (Https://Sqrrl.com/Product/Sqrrl-Enterprise/)

Technology (Https://Sqrrl.com/Product/Architecture/)

Architecture (Https://Sqrrl.com/Product/Architecture/)

Behavior Graph (Https://Sqrrl.com/Product/Behavior-Graph/)

User And Entity Behavior Analytics (Https://Sqrrl.com/Product/User-And-Entity-Behavior-Analytics-Ueba/)

Test Drive VM (Http://Info.sqrrl.com/Trial-Software-Vm-1)

SOLUTIONS (/SOLUTIONS/USE-CASES/)

Use Cases (Https://Sqrrl.com/Solutions/Use-Cases/)

Cyber Threat Hunting (Https://Sqrrl.com/Solutions/Cyber-Threat-Hunting/)

Cyber Incident Investigation (Https://Sqrrl.com/Solutions/Cyber-Incident-Response-And-Investigation/)

PARTNERS (HTTPS://SQRRL.COM/THE-SQRRL-PARTNER-PROGRAM/)

Threat Hunting Ecosystem (Https://Sqrrl.com/The-Sqrrl-Partner-Program/)

Technology (/Partners/Technology)

Sales (/Partners/Sales)

SERVICES (/SERVICES/SQRRL-ENTERPRISE-SUPPORT/)

Sqrrl Enterprise Support (Https://Sqrrl.com/Services/Sqrrl-Enterprise-Support/)

RESOURCES (/RESOURCES/)

Datasheets (/Resources/#Datasheet)

EBooks (/Resources/#Ebook)

Quick Reads (/Resources/#Quick-Read)

Reports (/Resources/#Report)

Videos (/Resources/#Video)

Webinars (/Resources/#Webinar)

Whitepapers (/Resources/#Whitepaper)

COMPANY (/COMPANY/OVERVIEW/)

Overview (Https://Sqrrl.com/Company/Overview/)

Team (/Company/Team/Management)

Advisors (Https://Sqrrl.com/Company/Team/Advisors/)

Blog (Http://Blog.sqrrl.com)

News Room (Https://Sqrrl.com/Company/News/)

Careers (Https://Sqrrl.com/Company/Careers/)

Contact Us (Https://Sqrrl.com/Company/Contact-Us/)