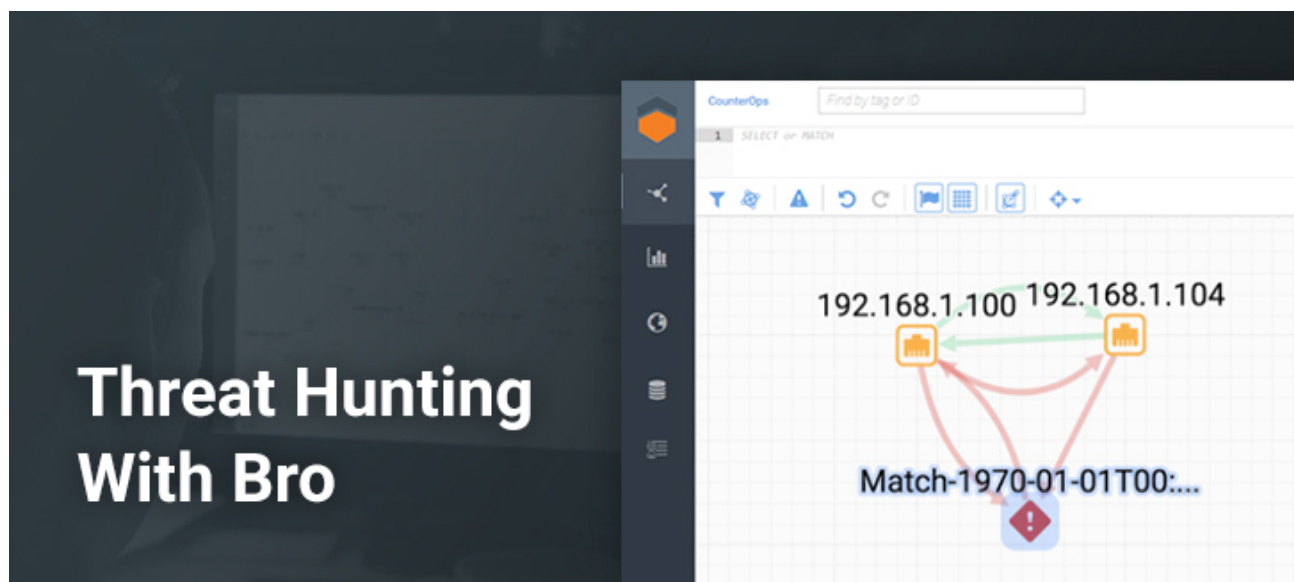


[BLOG \(HTTPS://SQRRL.COM/COMPANY/BLOG/\)](https://sqrrl.com/company/blog/)[TEST DRIVE VM \(HTTP://INFO.SQRRL.COM/TRIAL-SOFTWARE-VM-1\)](http://info.sqrrl.com/trial-software-vm-1)[SUPPORT PORTAL \(HTTPS://PORTAL.SQRRL.COM\)](https://portal.sqrrl.com)[PARTNER PORTAL \(HTTP://PARTNERS.SQRRL.COM/\)](http://partners.sqrrl.com/)[CONTACT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

THREAT HUNTING WITH BRO

(/blog/)



(/threat-hunting-bro/)

January 4, 2018 by Ryan Nolette (<https://sqrrl.com/author/ryan/>)

THREAT HUNTING WITH BRO

This blog is a quick overview of how I use Bro IDS for threat hunting.

Specifically:

- Example queries I run when I start a hunt by specific data set.
- Examples of Risk Trigger templates customized for my organization's environment
- Example of a Threat Hunt I performed

- Illustrate how the Threat Hunt I performed maps to the Threat Hunting Framework
- A few ideas of other Hunts you can do

What is Bro?

“Bro is an open source Unix based network monitoring framework. Often compared to a network intrusion detection system (NIDS), Bro can be used to build a NIDS but is much more. Bro can also be used for collecting network measurements, conducting forensic investigations, traffic baselining and more. Bro has been compared to tcpdump, Snort, netflow, and Perl (or any other scripting language) all in one. It is released under the BSD license.” – Wikipedia
([https://en.wikipedia.org/wiki/Bro_\(software\)](https://en.wikipedia.org/wiki/Bro_(software)))

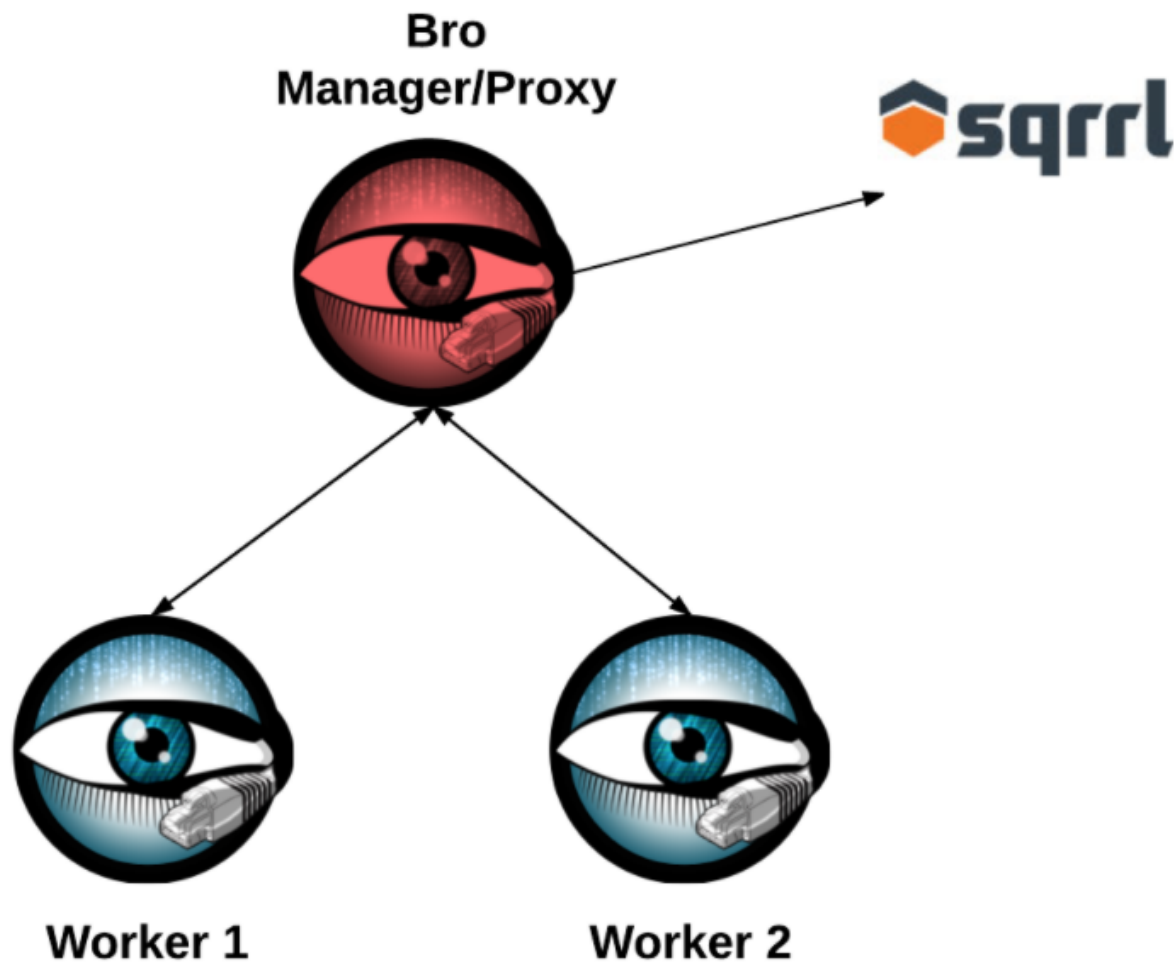
Basically, Bro is a protocol analyzer. It will accept network events from a PCAP file or a live traffic feed, watch it, and parse out individual protocols such as RDP, FTP, HTTP, and many more into individual log files. One of Bro's largest strengths is its ability to turn network events into actionable/useful metadata. And that metadata helps to provide us with context which is the key to finding potential threats quickly.

Bro log equivalent to normalized data source:

Standard datasource	Bro Equivalent
Firewall	Bro Conn (https://www.bro.org/sphinx/scripts/base/protocols/conn/main.bro.html#type-Conn::Info)
NetFlow	Bro Conn (https://www.bro.org/sphinx/scripts/base/protocols/conn/main.bro.html#type-Conn::Info)
Proxy	Bro HTTP (https://www.bro.org/sphinx/scripts/base/protocols/http/main.bro.html#type-HTTP::Info)
MS IIS	Bro HTTP (https://www.bro.org/sphinx/scripts/base/protocols/http/main.bro.html#type-HTTP::Info)
MS DNS Debug	Bro DNS (https://www.bro.org/sphinx/scripts/base/protocols/dns/main.bro.html#type-DNS::Info)

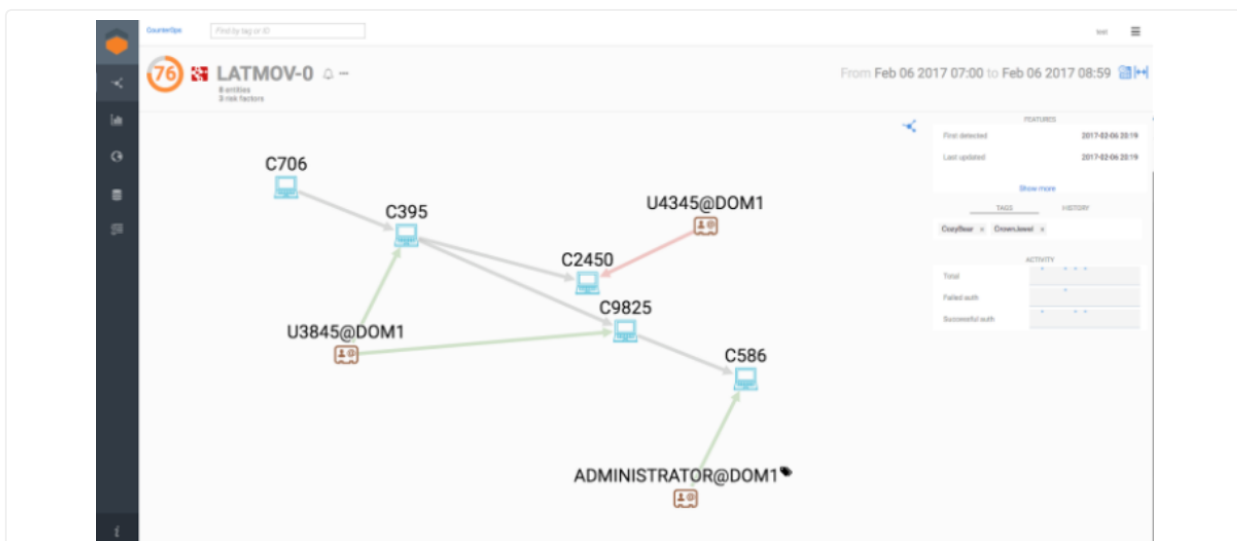
MS DNS Analytics	Bro DNS (https://www.bro.org/sphinx/scripts/base/protocols/dns/main.bro.html#type-DNS::Info)
MS DHCP	Bro DHCP (https://www.bro.org/sphinx/scripts/base/protocols/dhcp/main.bro.html#type-DHCP::Info)
SSHD	Bro SSH (https://www.bro.org/sphinx/scripts/base/protocols/ssh/main.bro.html#type-SSH::Info)
MySQL Server	Bro MySQL (https://www.bro.org/sphinx/scripts/base/protocols/mysql/main.bro.html#type-MySQL::Info)
MS Message Tracking	Bro SMTP (https://www.bro.org/sphinx/scripts/base/protocols/smtp/main.bro.html#type-SMTP::Info)

Logical Topology



(<https://sqrri.com/media/1-17.png>)

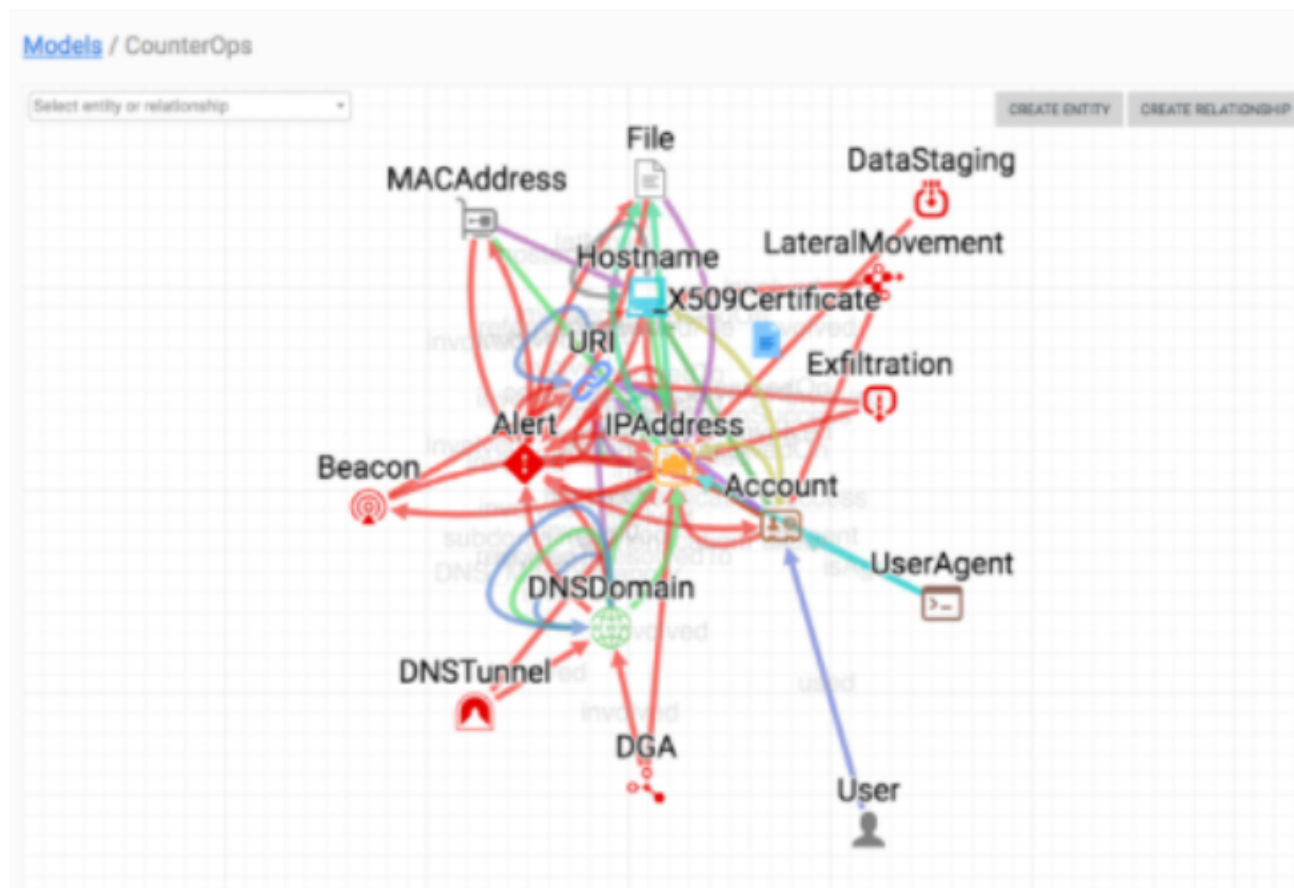
Threat Hunting With Bro



(<https://sqrri.com/media/2-17.png>)

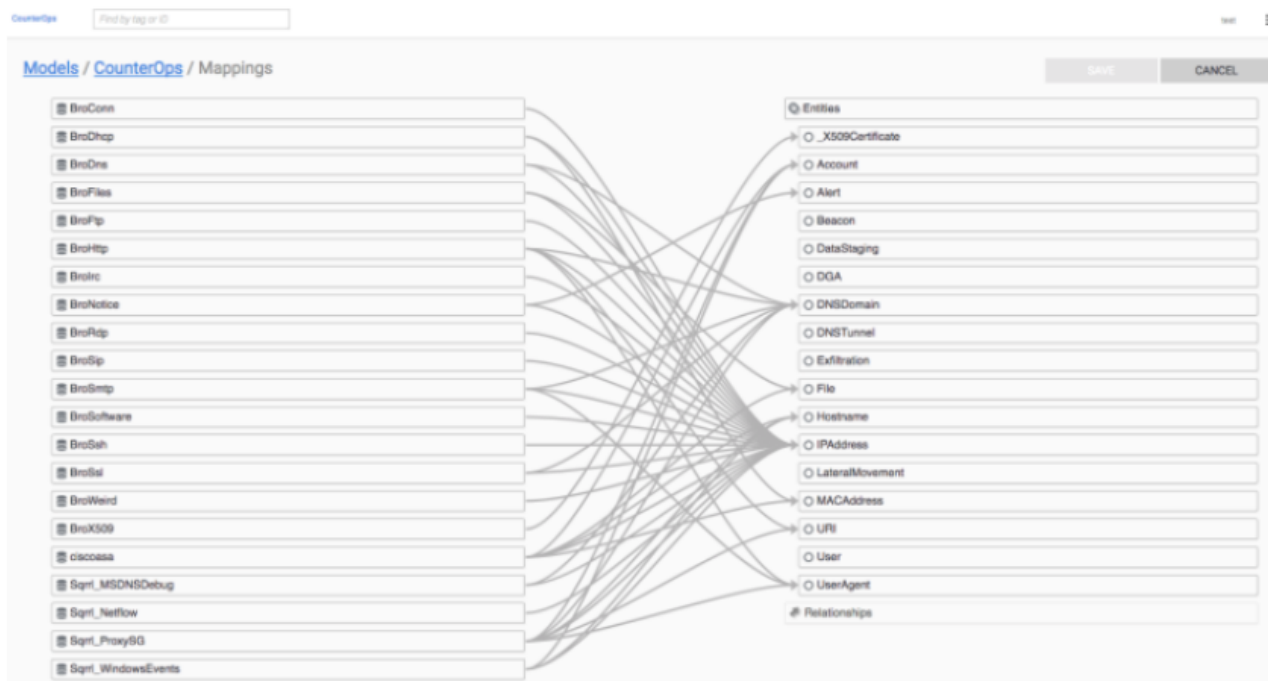
This image shows a completed Threat Hunt for Lateral Movement using netflow and Windows authentication logs.

Creating relationship between disparate data sets



(<https://sqrrl.com/media/3-13.png>)

The above image displays a high level visualization of all the relationships I have created between different entities within my data. This high level view allows me to quickly and easily see how my different types of data are connected and allows me to setup a game plan for a linked analysis hunt.



(<https://sqrrl.com/media/4-10.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.

Example hunt

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

Lateral Movement (Windows Environment)

What are you looking for? (Hypothesis)	Hypothesis: Attackers may be attempting to move laterally in my Windows environment by leveraging PsExec. Look for: <ul style="list-style-type: none">• Anomalies in host to host traffic leveraging the PsExec binary, service, and/or network traffic.• "C\$ ADMIN\$ IPC\$" shares being used in network traffic.
Investigation (Data)	Datasets: For identifying use of PsExec, you will want to focus primarily on application protocol metadata, including: <ul style="list-style-type: none">• 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)

Uncover Patterns and IOCs (Techniques)	<ol style="list-style-type: none"> 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 <ol style="list-style-type: none"> 1. For example: destination IP, port used, connection duration/length, etc.
Inform and Enrich Analytics (Takeaways)	<p>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.</p> <p>You can also create packet-level signatures to trigger alerts for cases where the admin share connections you have discovered may appear again.</p>

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.

Alert or Analytics Driven hunt

One technique to detect and alert on 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.

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 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.

Detecting PsExec Activity Using Bro

Modified code for my usage. Code is originally from here

(<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]);
  }
}
}
}
}

```

Detection of PsExec traffic via a Bro network sensor

```

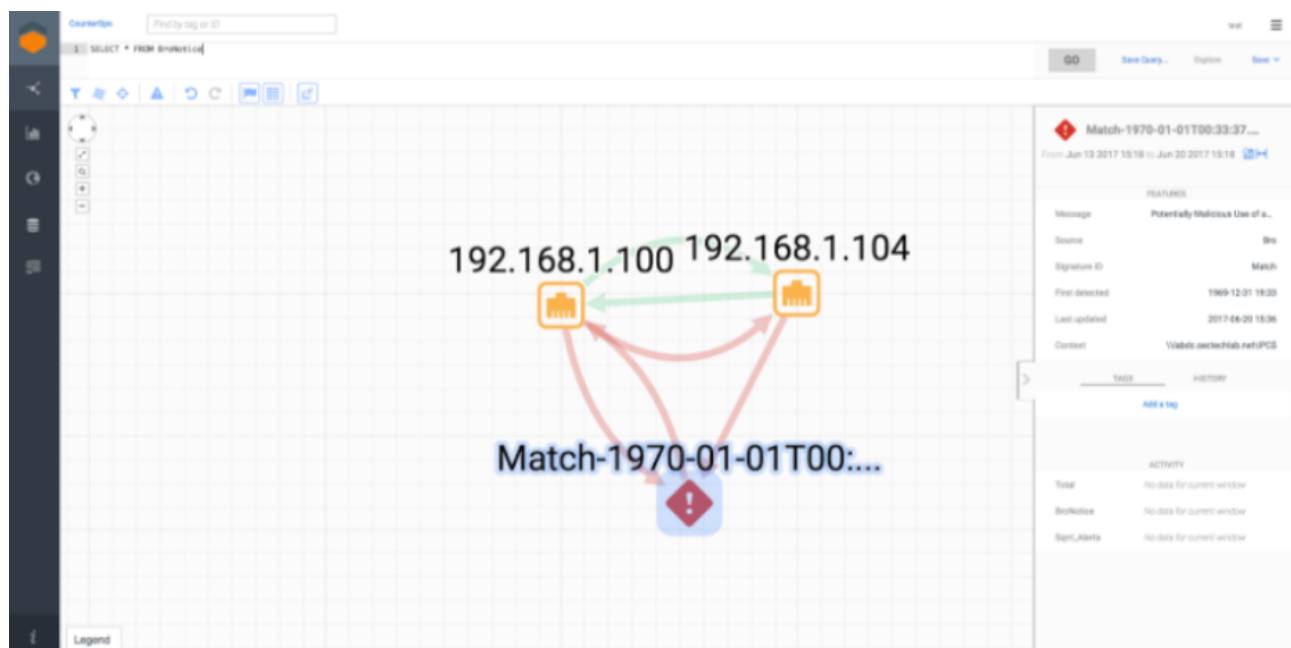
[root@brobro scripts]# grep -i "Malicious" /opt/bro/logs/current/*
/opt/bro/logs/current/loaded_scripts.log: ("name": " /opt/bro/spool/installed-scripts-do-not-touch/site/scripts/maliciousFile9PB.bro")
/opt/bro/logs/current/notice.log: ("ts": "2017-06-20T15:47:59.935586Z", "uid": "C08JDR3jetDhIba14", "id.orig_h": "192.168.1.106", "id.orig_p": 35623, "id.resp_h": "192.168.1.104", "id.resp_p": 445, "proto": "tcp", "note": "Match", "msg": "Potentially Malicious Use of an Administrative Share", "sub": "\u005c\u005c192.168.1.104\u005cIPC$", "src": "192.168.1.106", "dst": "192.168.1.104", "p": 445, "peer_descr": "bro", "actions": [{"Notice": "ACTION_LOG"}, {"suppress_for": 3600.0, "dropped": false}])
/opt/bro/logs/current/notice.log: ("ts": "2017-06-20T15:47:59.937829Z", "uid": "C08JDR3jetDhIba14", "id.orig_h": "192.168.1.106", "id.orig_p": 35623, "id.resp_h": "192.168.1.104", "id.resp_p": 445, "proto": "tcp", "note": "Match", "msg": "Potentially Malicious Use of an Administrative Share", "sub": "\u005c\u005c192.168.1.104\u005cADMIN$", "src": "192.168.1.106", "dst": "192.168.1.104", "p": 445, "peer_descr": "bro", "actions": [{"Notice": "ACTION_LOG"}, {"suppress_for": 3600.0, "dropped": false}])
/opt/bro/logs/current/notice.log: ("ts": "2017-06-20T15:47:59.954678Z", "uid": "C08JDR3jetDhIba14", "id.orig_h": "192.168.1.106", "id.orig_p": 35623, "id.resp_h": "192.168.1.104", "id.resp_p": 445, "proto": "tcp", "note": "Match", "msg": "Potentially Malicious Use of an Administrative Share", "sub": "\u005c\u005c192.168.1.104\u005cIPC$", "src": "192.168.1.106", "dst": "192.168.1.104", "p": 445, "peer_descr": "bro", "actions": [{"Notice": "ACTION_LOG"}, {"suppress_for": 3600.0, "dropped": false}])

```

(<https://sqrrl.com/media/5-6.png>)

In addition to remote 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\$""



(<https://sqrrl.com/media/6-2.png>)

Finally, with all the hard work done by that bro script, 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 of the relationship between the hosts. 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.

Intelligence/TTP driven hunting

Intelligence driven hunts are created from threat intelligence reports, threat intelligence feeds, malware analysis, vulnerability scans, and other trusted sources.

For this example, we are going to do a Hunt on http user-agents.

HTTP User-Agent Analysis

Background and Purpose

I want to identify malware by analyzing the User-Agent strings they leverage.

User-Agent (UA) strings are used to identify applications or services that perform HTTP requests. Similar to legitimate applications, HTTP-based malware may use distinct UA strings to identify itself to a command and control (C2) server; malware may also use common UA strings (e.g., UA strings used by legitimate web browsers) in order to blend in with normal web traffic.

The process described herein may also be used to analyze other HTTP headers and values, but User-Agents are among the most commonly used for hunting and detective measures.

Hypothesis

HTTP-based malware may use distinct UA strings during the C2 phase. If we analyze UA strings seen in our network and look for outliers, then we may find malware.

The assumption made in this type of analysis is that the activity in question will not be “normal” or overly prevalent on the active network. Ideally, this will lead to identification of malicious or otherwise prohibited activity that was missed via other detection mechanisms, which the UAs can then be used to detect in the future.

Data Required

- HTTP proxy data
- list of known-bad UAs (either external threat intelligence or internal threat intelligence)
- HTTP requests
 - This hunt requires metadata that contains HTTP requests.
 - This data should include
 - the UA string used in the HTTP requests
 - the source (endpoint, user, or IP address) of the HTTP request
 - the URI requested in the HTTP request.

Analysis Techniques

- Stack counting
- String matching
- Tokenization
- Outlier detection

Define Your Data Set

For this hunt, the data set is a set of UA strings used in outbound HTTP requests. Identification of these UA strings may vary from network to network; however, it is recommended to start with a larger set of data (e.g., all UA strings or a specific type of UA string) and reduce the size of the set as required by the results of the hunt.

Any UA string seen in an HTTP request can be considered for inclusion in the data set. However, you may want to consider defining the activity group based upon known legitimate UA strings. By filtering out these strings, outliers will be more noticeable. (However, keep in mind that filtering out legitimate UA strings will not help you identify attackers who are maliciously using legitimate UA strings!) There are several resources online for identifying common UA strings, here is one.

A query like this can be used to identify all UA strings:

```
SELECT user_agent, count(*) AS count FROM BroHttp WHERE user_agent IS NOT NULL GROUP BY user_agent ORDER BY count DESC
```

A query like this can be used to filter out legitimate UA strings from the results, would be one method for reducing your data set.

```
SELECT user_agent, count(*) AS count FROM BroHttp WHERE user_agent IS NOT NULL AND user_agent NOT IN ('<UA string 1>', '<UA string 2>', '<UA string 3>', ...) GROUP BY user_agent ORDER BY
```

If you've reached the point where you're routinely running hunts in order to iteratively determine what might be anomalous on the network, you'll likely find that adjusting timeframes for queries is of use. For instance, if the UA report is run weekly, you may

want to only pull one week of data for your report. It is also be worthwhile to compare those results with the results of the prior weeks, over time.

Identify Candidates

Depending on your data set, create a query that returns the UA strings you are interested in. This may be necessary in cases where there are too many results from a data-type wide search. One method of isolating UA strings is to look for uncommonly short or long strings. There is a relatively standard format for most To do this, we first have to identify a common UA string length. This query can do that:

```
SELECT avg(char_length(user_agent)) FROM BroHttp
```

Next, we need to isolate the results using the average length data from the previous query. A simple way to do this is to look for any UA string that is, for example, shorter than the average length, like this:

```
SELECT user_agent, char_length(user_agent) AS ua_len FROM BroHttp WHERE  
char_length(user_agent) < 66 GROUP BY user_agent, ua_len ORDER BY ua_len ASC  
LIMIT 20
```

OR

```
select * from BroHttp where length(user_agent) < 10 limit 20
```

OR

```
MATCH UserAgent AS entity FROM CounterOps WHERE len(entity.instance_id()) < 10  
limit 20
```

We limit the results to 20 so that we can quickly examine the results to make sure we're getting the right kinds of results.

At this point, we need to expand the LIMIT in the queries above and review the results to identify investigation candidates.

```
SELECT user_agent, char_length(user_agent) AS ua_len FROM BroHttp WHERE  
char_length(user_agent) < <average UA length> GROUP BY user_agent, ua_len ORDER  
BY ua_len ASC LIMIT 1000
```

With these results, start at the top of the list (which will be the shortest UA strings) and review the character length for each string– with the average UA string length and the content of the UA string in mind, identify outliers that you feel may be worth looking into further. An example of the results of this query is shown below:

user_agent	count
	0
Packer	6
Aether	6
curl/7.47.0	11
curl/7.30.0	11
curl/7.50.1	11

(<https://sqrri.com/media/7-3.png>)

There are some other data stacking slices that may be useful for us to examine in order to identify interesting candidates. In some cases, these may also help us learn about our network and the functions that take place across systems.

Next, we can try to identify which UAs were observed on only a few hosts. This could be of interest in the the case that an attacker is present or malware was deployed to only a few systems.

Lastly, we may also be interested in which UAs are observed infrequently in volume across the network as a whole. In certain situations, there's a possibility that the results from this query could differ from the prior one, which may introduce

additional candidates.

Description

- Stack the entire UA string and look for rare occurrences.
 - There may be a LOT of these, though. Every web plugin changes the UA string a bit, but that doesn't mean there's anything evil.
- Consider more detailed analysis, including
 - tokenizing the string and focusing on strings with the lowest number of tokens, most unique tokens, or some combination
 - Looking for abnormally short or long strings
- Look for list of known-bad UAs

Additional Sqrri Uses

The queries identified in the prior section can all be used as starting points for exploration in the behavior graph. However, one of the goals of hunting is to reduce the amount of repetition and to re-apply what was learned on a hunt. As such, there are two additional mechanisms within Sqrri that we can use to monitor occurrences of User-Agents.

Hunt Reports

We can turn each of our example candidate investigation queries into a hunt report in order to get quick snapshots to aid in identification of additional candidates on a regular basis. Hunt Reports are particularly well suited for data stacking.

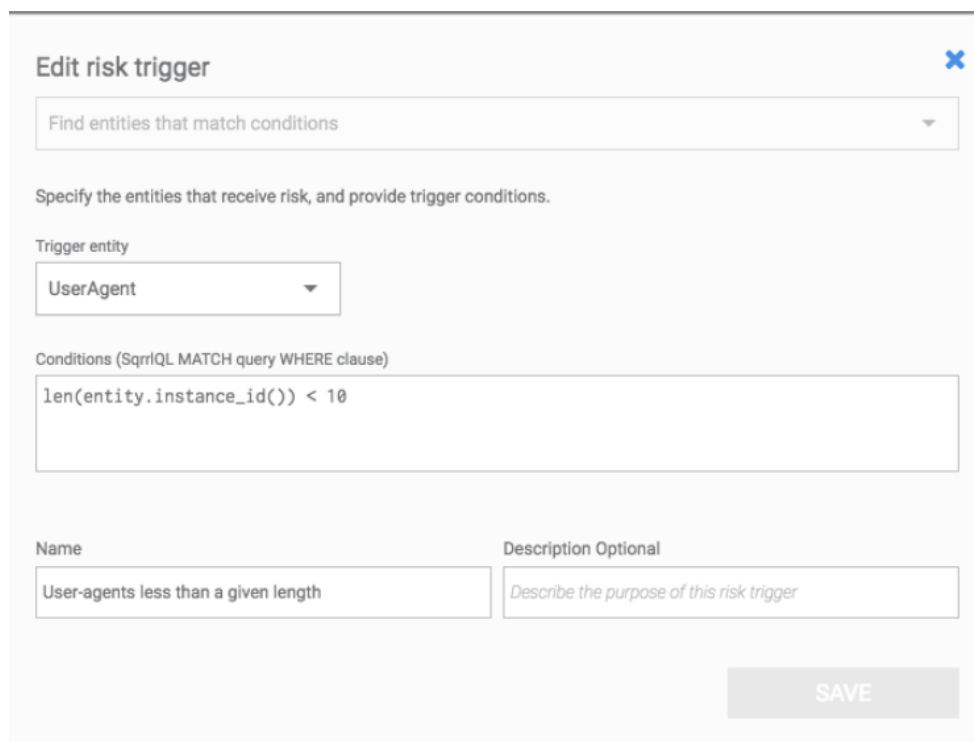
Hunt Reports	Bro Reports	CounterOps	Protocol Reports	HTTP Reports
UserAgents with less than 40 Characters, Past 7 Days				
TeXShop/3.88 Sparkle/1.18.0	2017-11-09 00:00:06	2017-11-09 00:00:06		1
Mac OS X/10.10.5 (14F2411)	2017-11-14 21:26:39	2017-11-14 21:26:39		1
DXCoreServiceToken	2017-11-14 15:12:24	2017-11-14 15:12:24		1
CaptiveNetworkSupport-346 wispr	2017-11-13 13:54:15	2017-11-13 13:54:15		1
Spotify/842600793 (36; 2; 3)	2017-11-13 23:16:18	2017-11-13 23:16:18		1
Setup/1.0 CFNetwork/889.9 Darwin/17.2.0	2017-11-10 18:21:07	2017-11-10 18:21:07		1
WicaAgent	2017-11-10 12:08:27	2017-11-10 12:08:27		1
AppleTV3,2/8.4.2 (12H606)	2017-11-10 18:55:47	2017-11-10 18:55:47		1

(<https://sqrri.com/media/8-3.png>)

Risk Triggers

While Hunt Reports assist with making data from certain hunts readily available, we can make use of Risk Triggers to identify observations of interest with respect to UAs and automatically use those to help bubble up entities of interest.

In one of our above examples, we determined that short UAs may be suspicious relative to others. As a result, I may want to create a Risk Trigger for entities observed using UAs shorter than a determined threshold.



Edit risk trigger

Find entities that match conditions

Specify the entities that receive risk, and provide trigger conditions.

Trigger entity

UserAgent

Conditions (Sqrrl MATCH query WHERE clause)

len(entity.instance_id()) < 10

Name

User-agents less than a given length

Description Optional

Describe the purpose of this risk trigger

SAVE

(<https://sqrrl.com/media/9-3.png>)

Triggers also provide a good area to make use threat intelligence data that may be collected from any number of sources. Perhaps a relevant threat was recently observed using malware that made use of a slightly malformed UA that falls within normal length bounds. We can implement a trigger to help raise the scores of internal entities with observed pattern matches.

Conclusion

Bro is powerful. Bro is free. Bro is a kind and benevolent ruler.

Bro offers something that many threat hunting tools don't, context. Using Bro as a protocol analyzer to identify traffic and its metadata are extremely valuable tools. Its ability to turn network events into actionable/useful metadata make it a must have in my security stack. This metadata helps to provide me with context which is the key to finding potential threats quickly.

Bro includes a scripting language that makes it possible to do indicator, packet-level, and "heuristic" network detection. Knowing how to convert a hunt conducted in Bro into an automated function in its framework adds tremendous value to a security operations team. This aspect of hunting is where Sqrrl Risk Triggers shine.

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

For more threat hunting insight:

- Read Ryan's blog post (<https://sqrri.com/finding-evil-when-hunting-for-lateral-movement/>) on threat hunting for lateral movement
- Watch our podcast (<https://www.youtube.com/watch?v=N7JvE9nBCJs>) on endpoint and network threat hunting
- Check out our training session (<http://info.sqrri.com/building-a-threat-hunting-team-david-bianco>) on building a threat hunting team
- View our webinar (<http://info.sqrri.com/threat-hunting-lateral-movement>) with Carbon Black featuring Ryan on how to threat hunt for lateral movement

(<https://sqrri.com/media/demo-1.png>)

(<http://info.sqrri.com/demo>)



(<https://sqrri.com/setting-threat-hunting-calendar-2018/>)

January 2, 2018 by Kristina Sisk ()

SETTING YOUR THREAT HUNTING CALENDAR FOR 2018 ([HTTPS://SQRRL.COM/SETTING-THREAT-HUNTING-CALENDAR-2018/](https://sqrri.com/setting-threat-hunting-calendar-2018/))

What is your team hunting for in 2018? If you don't know, how can you be sure you are positioned to safeguard your organization?


In the days of old, threat hunting was regarded as an ad hoc service for an organization. It is now an intrinsic part of an organization's defensive posture and provides the organization the ability to be nimble and seek out threat actors in their environment based on the most recent attacker TTPs. Threat hunting has undeniable return on investment for an organization, but with threat actor dwell times still averaging in the hundreds of days, the investment matters more.

READ MORE


(<https://sqrri.com/setting-threat-hunting-calendar-2018/>)

Threat Hunting

Buy, Build, Beg or Borrow



HOST
Paul Bartruff,
Engineer at Sqrrl



GUEST
Taylor Lehmann,
CISO at Wellforce

(<https://sqrrl.com/threat-hunting-buy-build-beg-borrow/>)

December 28, 2017 by Sqrrl Team (<https://sqrrl.com/author/george/>)

THREAT HUNTING: BUY, BUILD, BEG OR BORROW ([HTTPS://SQRRL.COM/THREAT-HUNTING-BUY-BUILD-BEG-BORROW/](https://sqrrl.com/threat-hunting-buy-build-beg-borrow/))

What goes into running a top-notch SOC? Recently, we sat down with (<https://www.youtube.com/watch?v=0Y-0wxqv7Ls>) Taylor Lehmann, the CISO of Wellforce, to get his takes on managing breaches, leveraging data, and adapting new hunting techniques.

READ MORE

(<https://sqrrl.com/threat-hunting-buy-build-beg-borrow/>)

Next Post

Browse by Topic

Featured Defenders ▴ ▾

Subscribe to Blog

Email Address

SUBSCRIBE

Featured Posts

Top #InfoSec Twitter Accounts (From A Threat Hunter's Perspective)

By Danny Akacki
(/top-infosec-twitter-accounts/)

Is Threat Hunting-As-A- Service (THaaS) for you?

By Luis Maldonado
(/threat-hunting-service-
thaas/)

Threat Hunting for Uncategorized Proxy Events

By Chris Sanders
(/cyber-threat-hunting-sqrri-
uncategorized-proxy-
events/)

Threat Hunting for Lateral Movement

Resources

Webinar

How to Perform "Friendly Reconnaissance" on your Network

([http://info.sqrri.com/friendly-
reconnaissance](http://info.sqrri.com/friendly-reconnaissance))

eBook

Hunt Evil: Your Practical Guide to Threat Hunting

([http://info.sqrri.com/practical-
guide-to-threat-hunting-
ebook](http://info.sqrri.com/practical-guide-to-threat-hunting-ebook))

Whitepaper

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

by Brandon Baxter
(/threat-hunting-lateral-
movement-identifying-pivot-
points/)

([http://info.sqrri.com/sqrri-
sans-hunting-white-paper](http://info.sqrri.com/sqrri-sans-hunting-white-paper))

Whitepaper

Threat Hunting for Evidence of Eavesdropping

By Matthew Hosburgh
(/hunting-evidence-
eavesdropping/)

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

([http://info.sqrri.com/sqrri-
product-paper-0](http://info.sqrri.com/sqrri-product-paper-0))

Threat Hunting Starting Points: Web Shells

By James Bower
(/3-threat-hunting-starting-
points-web-shells-edition/)

Watch Overview



([https://www.youtube.com/watch?
v=VI_zLBc4KQM&t&width=640&height=480](https://www.youtube.com/watch?v=VI_zLBc4KQM&t&width=640&height=480))

(<https://twitter.com/SQRRLNEW>)
(<https://www.facebook.com/SqrrlData>)
(<https://plus.google.com/116795302724746825954/posts>)
(<https://www.linkedin.com/company/sqrrl>)
(<http://www.youtube.com/user/sqrrldata>)

PRODUCT (/PRODUCT/SQRRL-ENTERPRISE) Sqrrl Enterprise (https://Sqrrl.Com/Product/Sqrrl-Enterprise/) Technology (https://Sqrrl.Com/Product/Technology/) Architecture (https://Sqrrl.Com/Product/Architecture/) Security Behavior Graph (https://Sqrrl.Com/Product/Security-Behavior-Graph/) User And Entity Behavior Analytics (https://Sqrrl.Com/Product/User-And-Entity-Behavior-Analytics-Ueba/) Test Drive VM (http://Info.Sqrrl.Com/Products/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/ENTERPRISE-SUPPORT/) Sqrrl-Partner-Program (https://Sqrrl.Com/Partners/Cyber-Technology/) Sales (https://Sqrrl.Com/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/)	GET A DEMO (http://INFO.SQRRL.COM/SQRRL-ENTERPRISE-DEMO-REQUEST)	BLOG (/BLOG/) (https://SQRRL.COM/SQRRL-ENTERPRISE-SUPPORT/)	CONTACT US (https://SQRRL.COM/COMPANY/CONTACT-US/)
---	---	--	---	--	--	--	---	--