

Nowhere to hide: Hunting adversaries on the **Network**

\$ whoami



Josef Gustafsson, *Systems Engineer*
Corelight Nordics / Baltics

Göteborg, Sweden





Security is hard
Attackers are getting better
Tools are getting better



Security is hard
Defenders are getting better
Tools are getting better

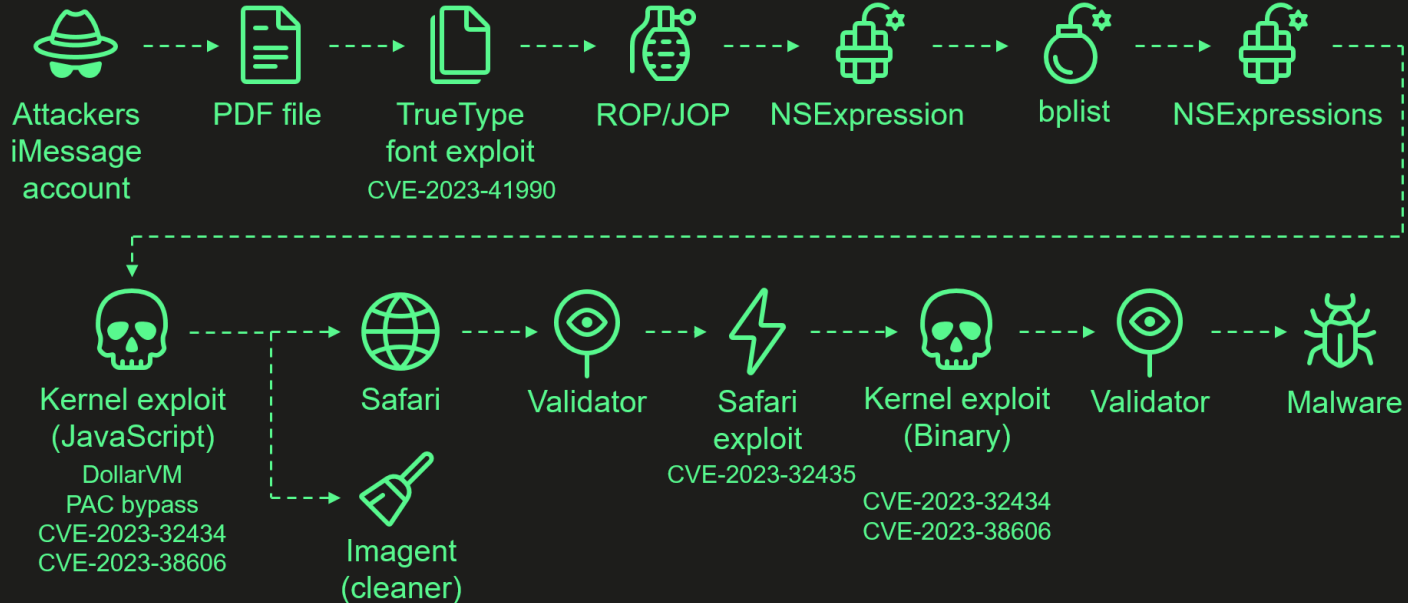


Attackers are getting better

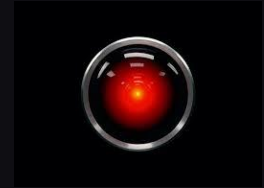
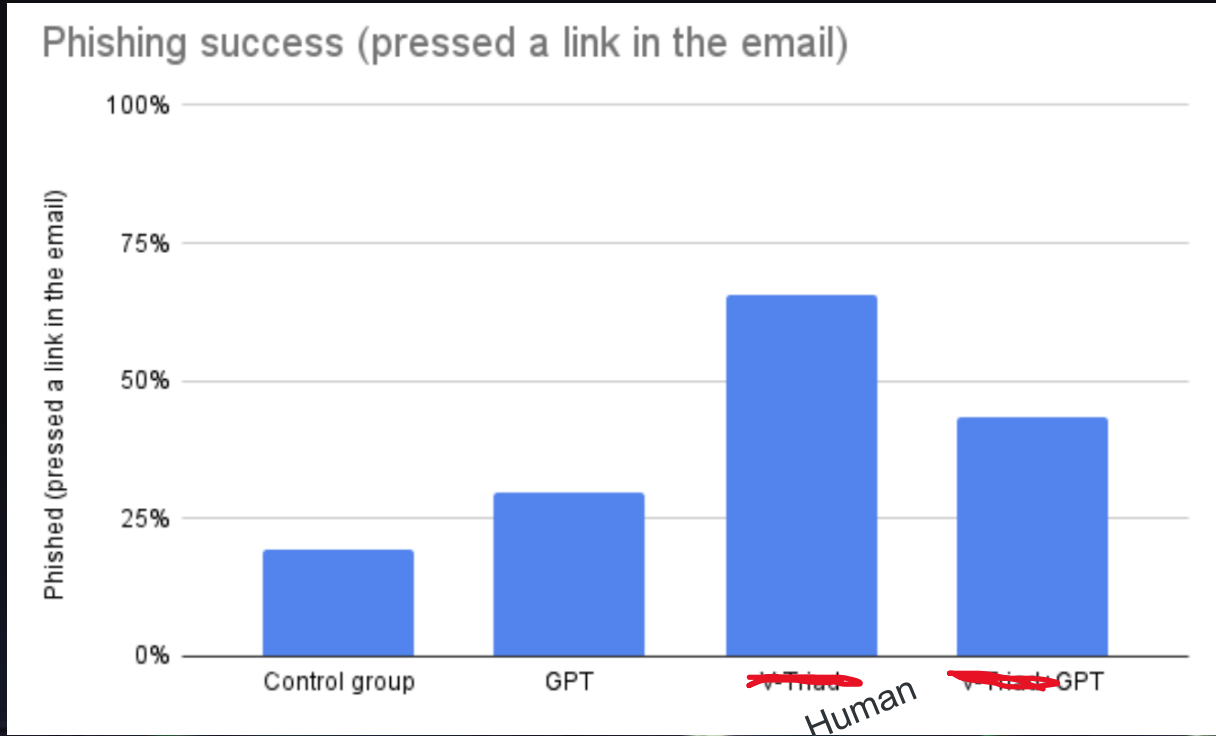


Operation Triangulation (Kaspersky, at 37C3)

Attack chain



Will I be replaced by AI?



Devising and Detecting Phishing: Large Language Models vs. Smaller Human Models
Fredrik Heiding, Bruce Schneier, Arun Vishwanath, Jeremy Bernstein

Ancient History (*pre-pandemic*)
Following **Sunburst Command & Control Traffic**



Threat hunting for Sunburst

- ...do I even have Solarwinds?
- Have I been compromised?
- What did the attackers do?



Threat hunting for Sunburst

- Attack used the domain "avsvmcloud.com" for C2
- Do we have any hosts reaching out?



Platform

Solutions

Intelligence

Services

Resources

Company

The DGA generates subdomains with the following DNS suffixes to create the FQDN:

- .appsync-api.eu-west-1[.]avsvmcloud[.]com
- .appsync-api.us-west-2[.]avsvmcloud[.]com
- .appsync-api.us-east-1[.]avsvmcloud[.]com
- .appsync-api.us-east-2[.]avsvmcloud[.]com

Threat hunting for Sunburst: DNS request

New Search

id.orig_h

10.220.30.132

10.220.30.133

10.230.13.10

10.230.13.11

10.230.7.211

10.230.7.212

10.230.7.26

8.0.0.11

8.0.0.15

8.0.0.9

8.0.1.10

```
index=corelight-sunburst path=dns avsvmcloud.com  
| stats count by id.orig_h
```

count

59

46

19

20

544

585

565

302

33

278

307

Threat hunting for Sunburst: Hosts reaching out

id.orig_h

10.220.30.132

10.220.30.133

10.230.13.10

10.230.13.11

10.230.7.211

10.230.7.212

10.230.7.26

8.0.0.11

8.0.0.15

Threat hunting for Sunburst

- Confirmed that we are impacted by Sunburst.
- Confirmed that several hosts are reaching out to attacker C2.
- Is it establishing contact?
- What commands are we receiving from the attacker?



Threat hunting for Sunburst: Requests getting through

New Search

```
index=corelight-sunburst path=dns avsvmcloud.com  
| stats count by id.orig_h,id.resp_h,query,rcode_name,answers{}
```

id.orig_h	id.resp_h	query	rcode_name	answers
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com	NOERROR	131.228.14.204
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com	NOERROR	87.238.80.34
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com	NOERROR	87.238.84.168
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-2.avsvmcloud.com	NOERROR	20.141.204.111
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-2.avsvmcloud.com	NOERROR	8.18.145.129
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-west-2.avsvmcloud.com	NOERROR	71.152.53.128
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-west-2.avsvmcloud.com	NOERROR	96.31.172.208
10.220.30.132	34.239.173.174	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com	NOERROR	131.228.15.253
10.220.30.132	34.239.173.174	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com	NOERROR	87.238.80.138
10.220.30.132	34.239.173.174	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com	NOERROR	87.238.81.8

Threat hunting for Sunburst: DNS Replies = C2

The malware starts in “Passive” mode, and only transitions to “Active” if it receives a DNS a record within the following blocks:

41.84.159.0/24 74.114.24.0/21 154.118.140.0/24 217.163.7.0/24

If it does not receive a reply in these ranges, it remains dormant and does not download a payload for the next stage.



Threat hunting for Sunburst: DNS Replies = C2

New Search

```
index=corelight-sunburst path=dns avsvmcloud.com  
| stats count by id.orig_h, id.resp_h, query, rcode_name, a
```

id.orig_h	id.resp_h	query
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-2.avsvmcloud.com
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-2.avsvmcloud.com
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-west-2.avsvmcloud.com
10.220.30.132	13.56.226.124	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-west-2.avsvmcloud.com
10.220.30.132	34.239.173.174	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com
10.220.30.132	34.239.173.174	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com
10.220.30.132	34.239.173.174	vvv8q9e0oh5q6u3aen60eudo1uv2f0c.appsync-api.us-east-1.avsvmcloud.com

answers() ⇅

131.228.14.204

87.238.80.34

87.238.84.168

20.141.204.111

8.18.145.129

71.152.53.128

96.31.172.208

131.228.15.253

87.238.80.138

87.238.81.8

Threat hunting for Sunburst: Conclusion

- Confirmed that we are impacted by Sunburst.
- Confirmed that 10.0.2.103 is reaching out to attacker C2.
- Confirmed contact and open C2 channel.
- All commands mapped to dormant behavior.
- Now we just need to clean up a bit, patch and install block rules 😊



Data is Key.



Pre-Historic to Today
Gozi (*Banking Malware Family*)



Gozi Banking Malware

One of the most widely spread and longest-standing Banking Trojans

2006: First seen

Source code leaked

2023: Mihai Ionut Paunescu, a Romanian hacker, sentenced to three years in U.S. prison



Gozi Banking Malware

GOZI/ISFB C2 TRAFFIC:

- 151.248.117.244 port 80 - diwdjndsfnj.ru - GET `/uploaded/[long base64 string with backslashes and underscores].pct`
- 151.248.117.244 port 80 - diwdjndsfnj.ru - POST `/uploaded/[long base64 string with backslashes and underscores].dib`
- 151.248.117.244 port 80 - diwdjndsfnj.ru - GET `/uploaded/[long base64 string with backslashes and underscores].pmg`
- 151.248.117.244 port 80 - iwqdndomdn.su - GET `/uploaded/[long base64 string with backslashes and underscores].pmg`
- 151.248.117.244 port 80 - iwqdndomdn.su - POST `/uploaded/[long base64 string with backslashes and underscores].dib`

GOZI/ISFB MODULES (ENCRYPTED DATA BINARIES):

- 91.199.147.95 port 80 - 91.199.147.95 - GET `/vnc32.rar`
- 91.199.147.95 port 80 - 91.199.147.95 - GET `/vnc64.rar`
- 91.199.147.95 port 80 - 91.199.147.95 - GET `/stilak32.rar`
- 91.199.147.95 port 80 - 91.199.147.95 - GET `/stilak64.rar`
- 91.199.147.95 port 80 - 91.199.147.95 - GET `/cook32.rar`
- 91.199.147.95 port 80 - 91.199.147.95 - GET `/cook64.rar`

Gozi Banking Malware

SIMPLE DETECTION:



```
$ cat http.log | zeek-cut host uri | gawk -F '\t' '$2 ~ /\(stilak|cook|vnc)(32|64)\.rar$/ || ($2 ~ /\^\w+\/([a-zA-Z0-9\|_?2V?F|_?2V?B|_?0V?A|_?0V?D){200,}\.[a-zA-Z0-9]+$/ && gsub(/\|/, "/", $2) > 10)'
```

<https://corelight.com/blog/gozi-banking-malware>

Gozi Banking Malware

```
151.248.117.244      80 GET
/uploaded/DjeJ0blPQ/_2BkfrDEoFQgD04w02F7/0jqqobto35jEVZ1IQyU/G7zu4_2BFUfhIMJcKkibbg/
fjjRaEElICvmR/
e5DoJnsG/vsx3T8e0iuXp0AlWknwttvf/A_2FrNprrb/bHnMsv4916Q0BUf_2/B3XIECBmUK_2/FW3G5XPXaPV/
ySf6P_2BIXQ
e7C/q0IvZNIhZt2c8lCjnMGY/BP81zPwMMzAU3VS/Y_2BCg7CLJsM0vz/Ml0Z0Th38yNZ0adE6L/qwrs9PKza/
13Lw9jqWnb
yh08rIXwcG/mMM0HdBwcj6NPi6_2FH/5qnQe2GM1T/ZuHEvxYT/j.pmg
1689201032.792223   CtPZjs20MLrsMU0Ji2   10.7.12.121           49801
91.199.147.95      80      GET      /vnc32.rar
1689201033.823106   CtPZjs20MLrsMU0Ji2   10.7.12.121           49801
91.199.147.95      80      GET      /vnc64.rar
```

AsyncRAT

(How not to do) encrypted C2



AsyncRAT

AsyncRAT is a **Remote Access Tool (RAT)** designed to remotely monitor and control other computers through a secure encrypted connection.

AsyncRAT can be delivered via **various methods** such as spear-phishing, malvertising, exploit kit and other techniques.

First seen in 2019

Whole family of malware with **many variants**



AsyncRAT

Popular in spring 2024 (#4 most seen)

Remote Access Trojan

Uses HTTPS Command & Control

Uses its real name in x509 TLS certificate...

x509.log | X.509 certificate info

FIELD	TYPE	DESCRIPTION
ts	time	Current timestamp
id	string	File ID of certificate
certificate	record X509::Certificate	Basic information about certificate
san	record X509::SubjectAlternativeName	Subject alternative name extension of certificate
basic_constraints	record X509::BasicConstraints	Basic constraints extension of certificate

AsyncRAT

CN=AsyncRAT Server

<https://corelight.com/blog/newsroom/news/hunt-of-the-month-detecting-async-rat-malware>

AsyncRAT

#path=x509 and (

certificate.issuer = /(((async|dc)rat|SXN) Server)|(DcRat By)/i or

certificate.subject = /(((async|dc)rat|SXN) Server)|(DcRat By)/i)

AsyncRAT

```
## AsyncRAT SURICATA RULE  
#  
alert tls $EXTERNAL_NET any -> $HOME_NET any (msg:"CORELIGHT AsyncRAT  
Style TLS Certificate"; flow:established,to_client; tls.cert_issuer;  
content:"RAT Server"; nocase; endswith; tls.cert_subject; content:"Rat";  
nocase; reference:url,any.run/malware-trends/asyncrat;  
classtype:trojan-activity; sid:3000065; rev:1;)
```

Defenders are getting better!
Nowhere to hide



The case for NDR: Network Detection and Response



It is everywhere:

- IoT/OT
- BYOD
- Shadow IT

When you need it the most,
you can still trust it

Invisible to threat actors,
no chance for adaptation

The power of network evidence



Adversaries leave footprints on the network

The network provides evidence

Evidence drives knowledge

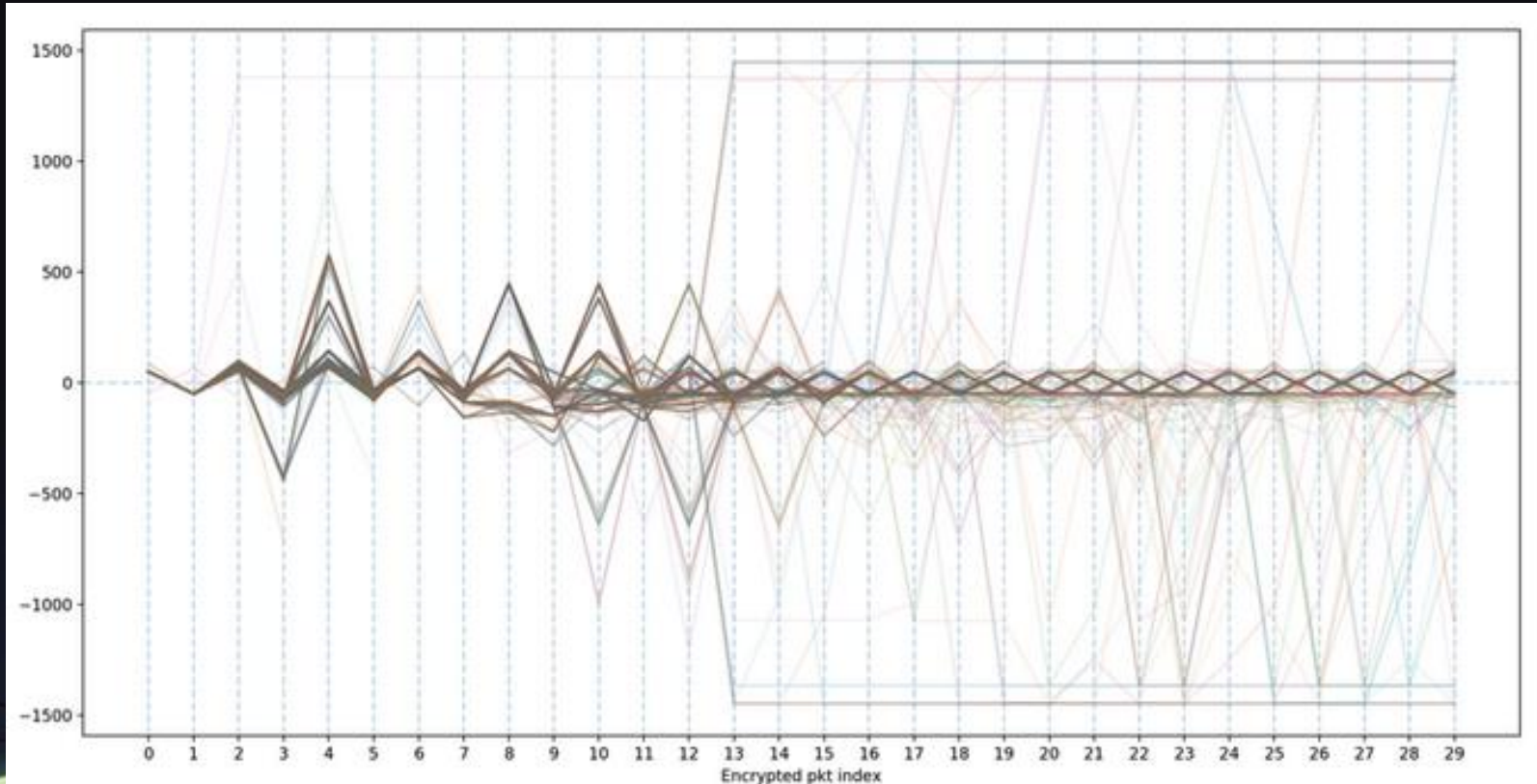
Knowledge fuels disruptive defense



Defenders are getting better!
Inferences in Encrypted Traffic

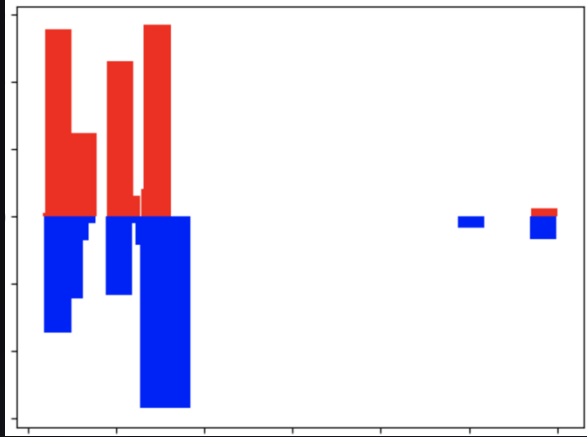


Analyzing encrypted traffic: SSH

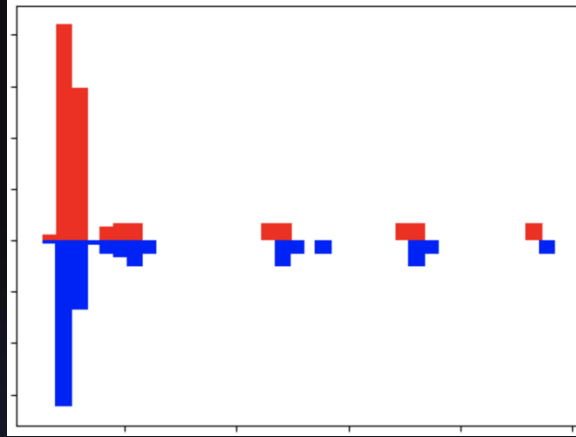


Analyzing encrypted traffic

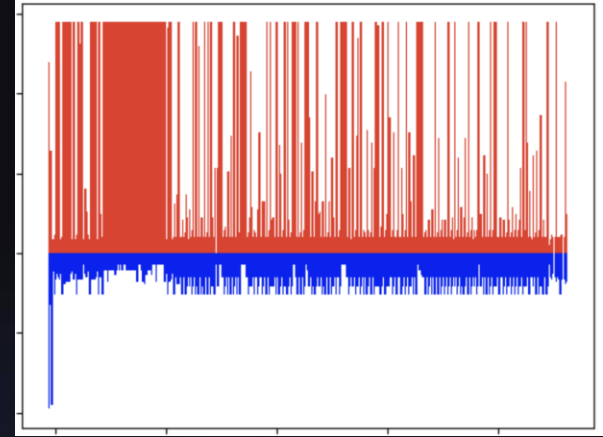
SSH traffic analysis



Normal login



Password guessing



Exporting data



Defenders are getting better!

AI & Large Language Models

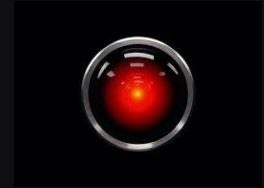


Generative AI use for a SOC analyst



Josef

```
alert smb any any -> $HOME_NET any (msg:"ET EXPLOIT Possible ETERNALBLUE MS17-010 Heap Spray"; flow:to_server,established; content:"|ff|SMB|33 00 00 00 00 18 07 c0 00 00 00 00 00 00 00 00 00 08 ff fe 00 08|"; offset:4; depth:30; fast_pattern; content:"|00 09 00 00 00 10|"; distance:1; within:6; content:"|00 00 00 00 00 00 10|"; within:8; content:"|00 00 00 10|"; distance:4; within:4; pcre:"/^[a-zA-Z0-9+/{1000,}/R"; metadata:attack_target SMB_Server, created_at 2017_04_17, deployment Internal, former_category EXPLOIT, signature_severity Critical, updated_at 2022_03_17; classtype:trojan-activity; threshold: type both, track by_src, count 3, seconds 30; gid:1; sid:2024217; rev:4;)
```



LLM

The Suricata rule is an alert rule that detects a possible exploitation attempt using the ETERNALBLUE MS17-010 vulnerability. The rule is triggered when there is a flow from any source SMB (Server Message Block) to any destination in the HOME_NET (defined network). The rule examines the packet contents and looks for specific patterns. It checks for the presence of specific byte sequences that indicate a heap spray attack. It also checks for the occurrence of specific byte sequences at certain offsets and distances within the packet. Additionally, the rule includes a Perl-Compatible Regular Expression (PCRE) pattern matching condition that looks for a base64-encoded string with a length of 1000 or more characters. The rule provides metadata about the attack target (SMB_Server), creation date (2017_04_17), deployment type (Internal), former category (EXPLOIT), severity (Critical), and update date (2022_03_17). It is classified as trojan-activity and has a threshold configured to trigger an alert if three instances of this rule are matched within a 30-second window. The rule is assigned a unique identifier (SID:2024217) and revision number (rev:4), and it belongs to the group with ID 1 (gid:1).

Generative AI use for a SOC analyst



Josef



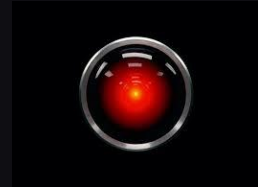
What might this alert mean?

This Suricata alert indicates a possible exploitation attempt using the EternalBlue vulnerability (MS17-010). EternalBlue is a known exploit that targets the Server Message Block (SMB) protocol. The rule in this alert specifically looks for a heap spray attack related to EternalBlue.

The alert's rule inspects network traffic flowing from any source IP to your network (HOME_NET) on any port. It examines the flow direction and checks for specific byte sequences within the packets. The rule searches for a series of bytes indicating an SMB protocol message, followed by specific patterns that align with the heap spray technique used in the EternalBlue exploit.

The combination of these byte sequences, offsets, depths, and content distances helps identify potential instances of the attack. Additionally, the rule includes a regular expression (pcre) to detect a long base64-encoded string, which could indicate malicious payload.

The alert is classified as "trojan-activity" and has a critical severity level, as it suggests an attempt to exploit a significant vulnerability. The alert's metadata provides additional information such as the attack target, deployment (internal network), and creation/update dates. The threshold indicates that if this alert is triggered three times within a 30-second timeframe by the same source, it will be counted and tracked. The rule's unique identifier is sid:2024217, and it is currently in revision 4.



LLM

Generative AI use for a SOC analyst



Josef



What are some possible next steps to investigate this alert?

To investigate the Suricata alert for the possible ETERNALBLUE MS17-010 Heap Spray, a Cyber Security Analyst would typically use the following Zeek logs:

1. **conn.log:** This log captures connection-level information such as source and destination IP addresses, ports, timestamps, and duration of the connections. It can help identify the involved hosts and potential communication patterns.
2. **files.log:** This log records metadata about files transferred over the network. It includes details like file names, sizes, hashes, and MIME types. It can assist in identifying any files associated with the exploit attempt.
3. **http.log:** This log provides information related to HTTP transactions. It includes details about HTTP requests and responses, such as URLs, user agents, and response codes. It can be useful for tracing any HTTP-based activities related to the exploit.
4. **smb_files.log:** This log specifically focuses on SMB file transfers. It can provide insights into any SMB-related interactions and potentially identify files involved in the exploit attempt.

To investigate the Suricata alert, the analyst should follow these steps:

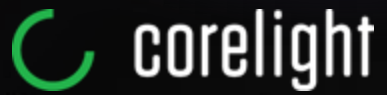
1. Identify the source and destination IP addresses from the alert. Cross-reference them with the **conn.log** to gather additional context about the communication.
2. Examine the **files.log** to check if any files were transferred during the suspicious connection. Pay attention to the file names, sizes, and hashes. Look for any files that align with the ETERNALBLUE MS17-010 exploit.
3. Analyze the **http.log** to determine if any HTTP transactions occurred during the connection in question. Look for any URLs, user agents, or response codes that may provide further insights into the exploit



LLM

Data is Key.





THANK YOU!

NETWORK DETECTION AND RESPONSE | ON PREMISE AND IN THE CLOUD