David Scott - Google Cybersecurity Course - Incident Handler’s Journal Portfolio Piece

Incident handler's journal

Scenario 1

A small U.S. health care clinic specializing in delivering primary-care services experienced a security incident on a Tuesday morning, at approximately 9:00 a.m. Several employees reported that they were unable to use their computers to access files like medical records. Business operations shut down because employees were unable to access the files and software needed to do their job.

Additionally, employees also reported that a ransom note was displayed on their computers. The ransom note stated that all the company's files were encrypted by an organized group of unethical hackers who are known to target organizations in healthcare and transportation industries. In exchange for restoring access to the encrypted files, the ransom note demanded a large sum of money in exchange for the decryption key.

The attackers were able to gain access into the company's network by using targeted phishing emails, which were sent to several employees of the company. The phishing emails contained a malicious attachment that installed malware on the employee's computer once it was downloaded.

Once the attackers gained access, they deployed their ransomware, which encrypted critical files. The company was unable to access critical patient data, causing major disruptions in their business operations. The company was forced to shut down their computer systems and contact several organizations to report the incident and receive technical assistance.

Date: December 5, 2023	**Entry:** 0001
Description	Medical facility suffering a ransomware attack.
Tool(s) used	None
	At 9:00 AM, employees discovered they were unable to access business files including customer information. The files were found to be encrypted along with a ransomware message from known threat actor Crystal Tornado (Microsoft designation), seeking a sum of money.

	The preceding day an employee had received a phishing email containing a malicious link. Clicking on the link allowed Crystal Tornado access to the systems and to install the encryption malware.
Additional notes	● Does the facility have pertinent back-ups which can be used to restore the systems? ● Does the facility use proper authorization techniques? ● Has the staff had cybersecurity training courses?

Scenario 2

You are a level one security operations center (SOC) analyst at a financial services company. You receive an alert about a suspicious file being downloaded on an employee's computer.

You investigate and discover the employee received an email containing an attachment. The attachment was a password-protected spreadsheet file. The spreadsheet's password was provided in the email. The employee downloaded the file, then entered the password to open the file. When the employee opened the file, a malicious payload was executed on their computer.

Date: December 7, 2023	**Entry:** 002
Description	Probably malicious email attachment
Tool(s) used	SHA256 hash, Virus Total
The 5 W's	– 1:11 p.m.: An employee received an email containing a spreadsheet attachment along with a password. – 1:13 p.m.: The employee successfully downloads and opens the file. – 1:15 p.m.: Multiple unauthorized executable files are created on the employee's computer. – 1:20 p.m.: An IDS detects the executable files and sends an alert to the SOC.

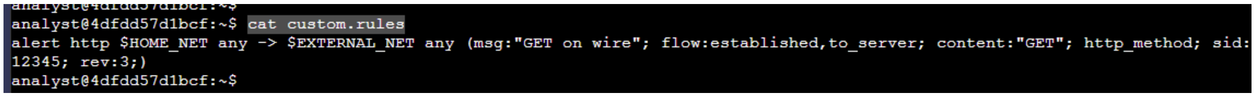
	I retrieved the file and created a SHA256 hash (54e6ea47eb04634d3e87fd7787e2136ccfbcc80ade34f246a12cf93bab527f6b). Virus Total investigation reveals the file to a backdoor trojan called Flagpro used by the threat actor BlackTech with a high confidence rate.
Additional notes	Block org.misecure.com and IP address 207.148.109.242

SURICATA RULES AND LOGS

Examine and Explain Suricata Custom Rules

The /home/analyst directory contains a custom.rules file that defines the network traffic rules and a sample.pcap file.

I use the cat custom.rules command to display the contents of the rules file (my commands highlighted in gray)



```
analyst@4dfdd57d1bcf:~$
analyst@4dfdd57d1bcf:~$ cat custom.rules
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"GET on wire"; flow:established,to_server; content:"GET"; http_method; sid:
12345; rev:3;)
analyst@4dfdd57d1bcf:~$
```

The returned file contents:

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:”GET on wire”; flow:established,to_server; content:”GET”; http_method; sid:12345; rev:3;)

Alert is the action which Suricata will take when a packet matches this rule.

Http is the first part of the header and the protocol which Suricata will apply this rule to.

$HOME_NET any is a variable defined in the */etc/suricata/suricata.yaml* definitions file to be a placeholder for a local network while *any* designates any port on the network. In this instance *$HOME_NET* is defined as *172.21.224.0/20*.

The *msg:* option provides the text to output to explain why the alert was triggered, in this case “GET on wire”.

The *flow:established,to_server* option determines packets from the client to the server should be matched.

The *content:"GET"* option tells Suricata to look for GET in the content of the http.method part of the packet.

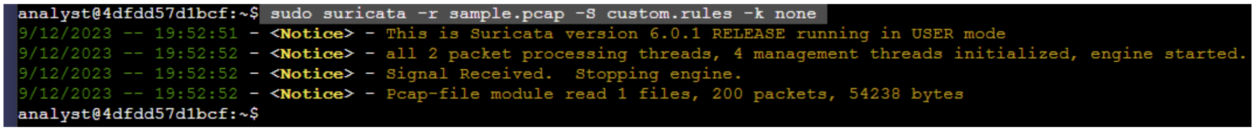
The *sid:12345* option is a unique number to identify the rule while *rev:3* shows this is the third revision of the rule.

Trigger a Suricata Rule

Run Suricata using the custom.rules and sample.pcap files to generate log files.

The command used:

sudo suricata -r sample.pcap -S custom.rules -k none

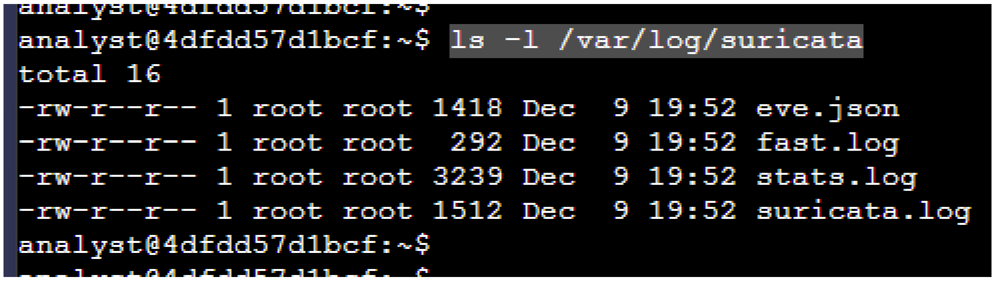


```
analyst@4dfdd57d1bcf:~$ sudo suricata -r sample.pcap -S custom.rules -k none
9/12/2023 -- 19:52:51 - <Notice> - This is Suricata version 6.0.1 RELEASE running in USER mode
9/12/2023 -- 19:52:52 - <Notice> - all 2 packet processing threads, 4 management threads initialized, engine started.
9/12/2023 -- 19:52:52 - <Notice> - Signal Received.  Stopping engine.
9/12/2023 -- 19:52:52 - <Notice> - Pcap-file module read 1 files, 200 packets, 54238 bytes
analyst@4dfdd57d1bcf:~$
```

The *-r* option specifies an input file (sample.pcap).
The *-S* instructs Suricata to use the rules defined in custom.rules.
The *-k none* option instructs Suricata to disable checksums.

Examine the logs

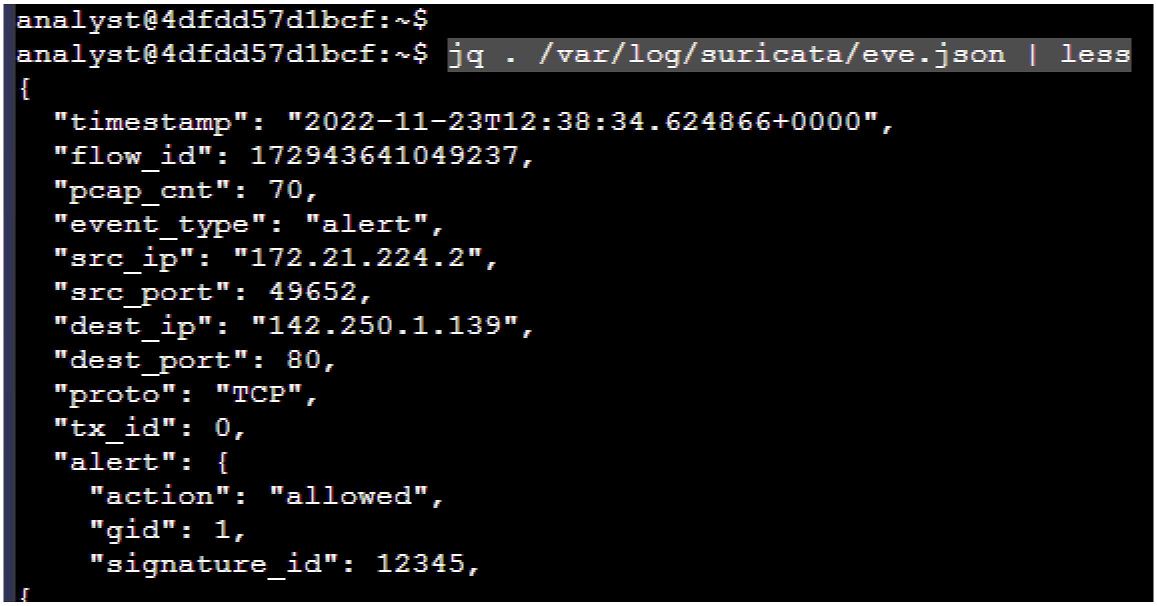
As can be seen below Suricata created four log files using the custom rules and sample.pcap.



```
analyst@4dfdd57d1bcf:~$
analyst@4dfdd57d1bcf:~$ ls -l /var/log/suricata
total 16
-rw-r--r-- 1 root root 1418 Dec  9 19:52 eve.json
-rw-r--r-- 1 root root  292 Dec  9 19:52 fast.log
-rw-r--r-- 1 root root 3239 Dec  9 19:52 stats.log
-rw-r--r-- 1 root root 1512 Dec  9 19:52 suricata.log
analyst@4dfdd57d1bcf:~$
```

The most useful of these four is usually the eve.json file as it contains the most detailed summary of events. It is a standard json output file and JQuery is used to examine it.

Output of the eve.json file (shortened for space)



```
analyst@4dfdd57d1bcf:~$
analyst@4dfdd57d1bcf:~$ jq . /var/log/suricata/eve.json | less
{
  "timestamp": "2022-11-23T12:38:34.624866+0000",
  "flow_id": 172943641049237,
  "pcap_cnt": 70,
  "event_type": "alert",
  "src_ip": "172.21.224.2",
  "src_port": 49652,
  "dest_ip": "142.250.1.139",
  "dest_port": 80,
  "proto": "TCP",
  "tx_id": 0,
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 12345,
```