Examine alerts, logs, and rules with Suricata

Scenario

In this scenario, you're a security analyst who must monitor traffic on your employer's network. You'll be required to configure Suricata and use it to trigger alerts.

Here's how you'll do this task: **First**, you'll explore custom rules in Suricata. **Second**, you'll run Suricata with a custom rule in order to trigger it, and examine the output logs in the `fast.log` file. **Finally**, you'll examine the additional output that Suricata generates in the standard `eve.json` log file.

For the purposes of the tests you'll run in this lab activity, you've been supplied with a `sample.pcap` file and a `custom.rules` file. These reside in your home folder.

Let's define the files you'll be working with in this lab activity:

- The `sample.pcap` file is a packet capture file that contains an example of network traffic data, which you'll use to test the Suricata rules. This will allow you to simulate and repeat the exercise of monitoring network traffic.

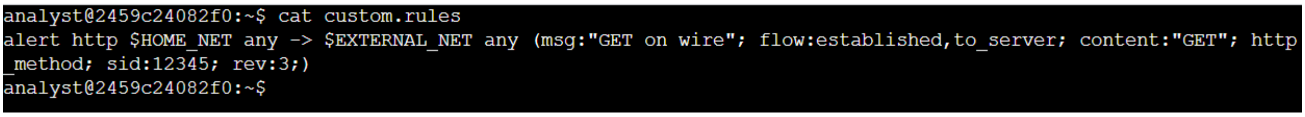
- The `custom.rules` file contains a custom rule when the lab activity starts. You'll add rules to this file and run them against the network traffic data in the `sample.pcap` file.

- The `fast.log` file will contain the alerts that Suricata generates. The `fast.log` file is empty when the lab starts. Each time you test a rule, or set of rules, against the sample network traffic data, Suricata adds a new alert line to the `fast.log` file when all the conditions in any of the rules are met. The `fast.log` file can be located in the `/var/log/suricata` directory after Suricata runs.The `fast.log` file is considered to be a depreciated format and is not recommended for incident response or threat hunting tasks but can be used to perform quick checks or tasks related to quality assurance.

- The `eve.json` file is the main, standard, and default log for events generated by Suricata. It contains detailed information about alerts triggered, as well as other network telemetry events, in JSON format. The `eve.json` file is generated when Suricate runs, and can also be located in the `/var/log/suricata` directory.

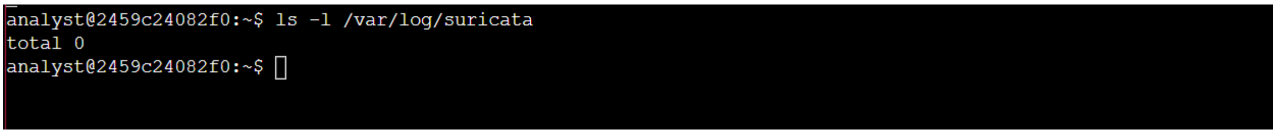
When you create a new rule, you'll need to test the rule to confirm whether or not it worked as expected. You can use the `fast.log` file to quickly compare the number of alerts generated each time you run Suricata to test a signature against the `sample.pcap` file.

It's time to get started.



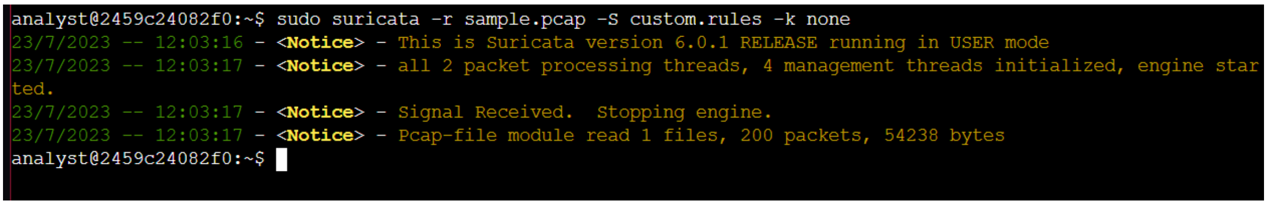
```
analyst@2459c24082f0:~$ cat custom.rules
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"GET on wire"; flow:established,to_server; content:"GET"; http
_method; sid:12345; rev:3;)
analyst@2459c24082f0:~$
```

Figure 1 Using the cat command to display the rule in the custom.rules file



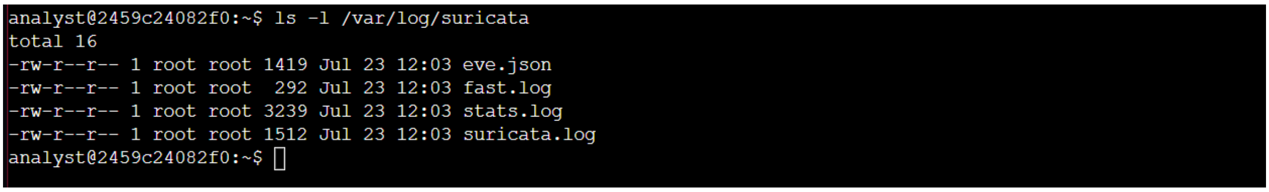
```
analyst@2459c24082f0:~$ ls -l /var/log/suricata
total 0
analyst@2459c24082f0:~$ 
```

Figure 2 Listing the files in the /var/log/suricata folder



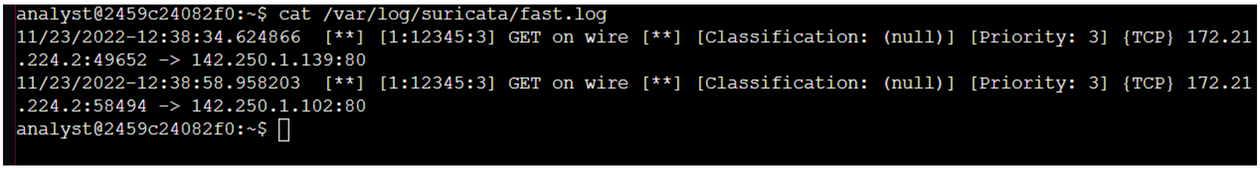
```
analyst@2459c24082f0:~$ sudo suricata -r sample.pcap -S custom.rules -k none
23/7/2023 -- 12:03:16 - <Notice> - This is Suricata version 6.0.1 RELEASE running in USER mode
23/7/2023 -- 12:03:17 - <Notice> - all 2 packet processing threads, 4 management threads initialized, engine star
ted.
23/7/2023 -- 12:03:17 - <Notice> - Signal Received.  Stopping engine.
23/7/2023 -- 12:03:17 - <Notice> - Pcap-file module read 1 files, 200 packets, 54238 bytes
analyst@2459c24082f0:~$ 
```

Figure 3 Runing suricata using the custom.rules and sample.pcap files



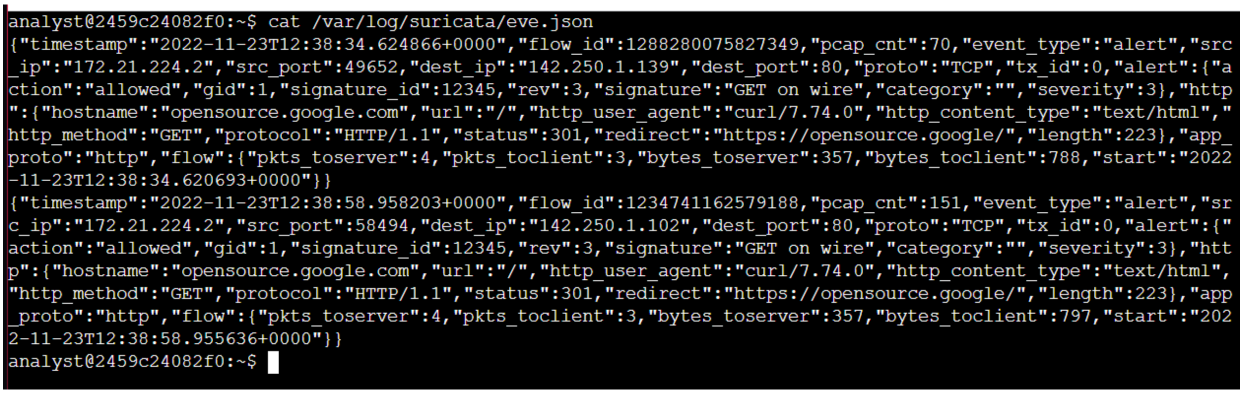
```
analyst@2459c24082f0:~$ ls -l /var/log/suricata
total 16
-rw-r--r-- 1 root root 1419 Jul 23 12:03 eve.json
-rw-r--r-- 1 root root  292 Jul 23 12:03 fast.log
-rw-r--r-- 1 root root 3239 Jul 23 12:03 stats.log
-rw-r--r-- 1 root root 1512 Jul 23 12:03 suricata.log
analyst@2459c24082f0:~$ 
```

Figure 4 Listing the files in the /var/log/suricata folder again



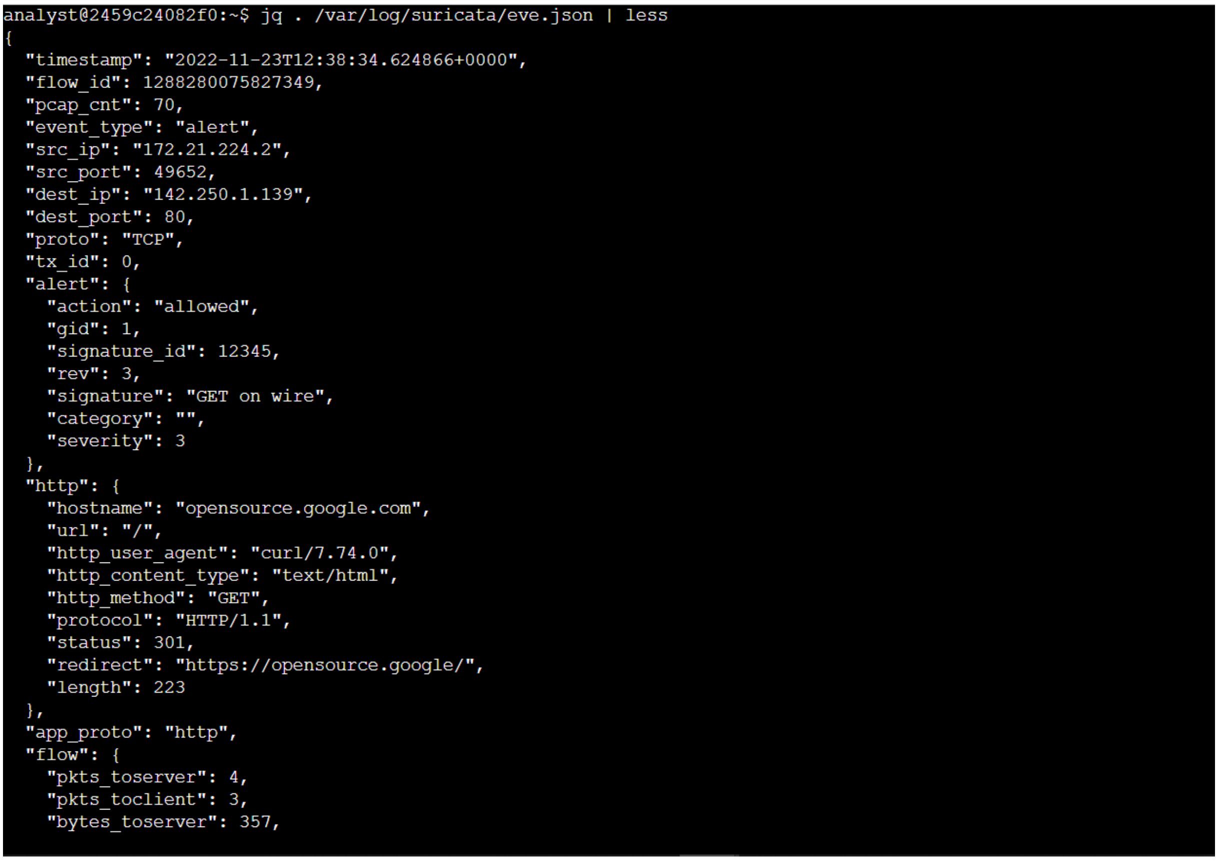
```
analyst@2459c24082f0:~$ cat /var/log/suricata/fast.log
11/23/2022-12:38:34.624866  [**] [1:12345:3] GET on wire [**] [Classification: (null)] [Priority: 3] {TCP} 172.21
.224.2:49652 -> 142.250.1.139:80
11/23/2022-12:38:58.958203  [**] [1:12345:3] GET on wire [**] [Classification: (null)] [Priority: 3] {TCP} 172.21
.224.2:58494 -> 142.250.1.102:80
analyst@2459c24082f0:~$ 
```

Figure 5 Using the cat command to display the fast.log file generated by Suricata



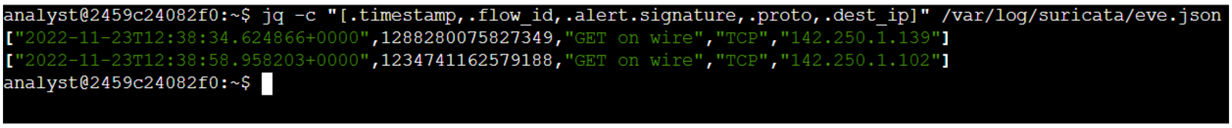
```
analyst@2459c24082f0:~$ cat /var/log/suricata/eve.json
{"timestamp":"2022-11-23T12:38:34.624866+0000","flow_id":1288280075827349,"pcap_cnt":70,"event_type":"alert","src
_ip":"172.21.224.2","src_port":49652,"dest_ip":"142.250.1.139","dest_port":80,"proto":"TCP","tx_id":0,"alert":{"a
ction":"allowed","gid":1,"signature_id":12345,"rev":3,"signature":"GET on wire","category":"","severity":3},"http
":{"hostname":"opensource.google.com","url":"/","http_user_agent":"curl/7.74.0","http_content_type":"text/html","
http_method":"GET","protocol":"HTTP/1.1","status":301,"redirect":"https://opensource.google/","length":223},"app_
proto":"http","flow":{"pkts_toserver":4,"pkts_toclient":3,"bytes_toserver":357,"bytes_toclient":788,"start":"2022
-11-23T12:38:34.620693+0000"}}
{"timestamp":"2022-11-23T12:38:58.958203+0000","flow_id":1234741162579188,"pcap_cnt":151,"event_type":"alert","sr
c_ip":"172.21.224.2","src_port":58494,"dest_ip":"142.250.1.102","dest_port":80,"proto":"TCP","tx_id":0,"alert":{"
action":"allowed","gid":1,"signature_id":12345,"rev":3,"signature":"GET on wire","category":"","severity":3},"htt
p":{"hostname":"opensource.google.com","url":"/","http_user_agent":"curl/7.74.0","http_content_type":"text/html",
"http_method":"GET","protocol":"HTTP/1.1","status":301,"redirect":"https://opensource.google/","length":223},"app
_proto":"http","flow":{"pkts_toserver":4,"pkts_toclient":3,"bytes_toserver":357,"bytes_toclient":797,"start":"202
2-11-23T12:38:58.955636+0000"}}
analyst@2459c24082f0:~$
```

Figure 6 Using the cat command to display the entries in the eve.json file



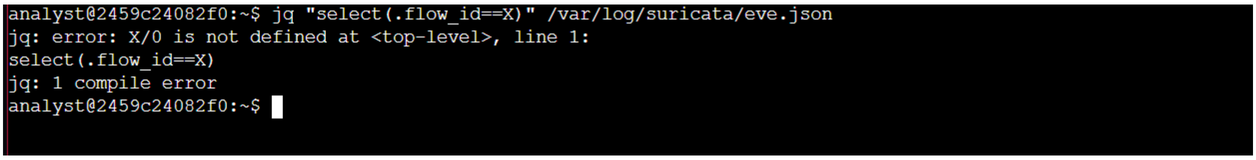
```
analyst@2459c24082f0:~$ jq . /var/log/suricata/eve.json | less
{
  "timestamp": "2022-11-23T12:38:34.624866+0000",
  "flow_id": 1288280075827349,
  "pcap_cnt": 70,
  "event_type": "alert",
  "src_ip": "172.21.224.2",
  "src_port": 49652,
  "dest_ip": "142.250.1.139",
  "dest_port": 80,
  "proto": "TCP",
  "tx_id": 0,
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 12345,
    "rev": 3,
    "signature": "GET on wire",
    "category": "",
    "severity": 3
  },
  "http": {
    "hostname": "opensource.google.com",
    "url": "/",
    "http_user_agent": "curl/7.74.0",
    "http_content_type": "text/html",
    "http_method": "GET",
    "protocol": "HTTP/1.1",
    "status": 301,
    "redirect": "https://opensource.google/",
    "length": 223
  },
  "app_proto": "http",
  "flow": {
    "pkts_toserver": 4,
    "pkts_toclient": 3,
    "bytes_toserver": 357,
```

Figure 7 Using the jq command to display the entries in an improved format



```
analyst@2459c24082f0:~$ jq -c "[.timestamp,.flow_id,.alert.signature,.proto,.dest_ip]" /var/log/suricata/eve.json
["2022-11-23T12:38:34.624866+0000",1288280075827349,"GET on wire","TCP","142.250.1.139"]
["2022-11-23T12:38:58.958203+0000",1234741162579188,"GET on wire","TCP","142.250.1.102"]
analyst@2459c24082f0:~$
```

Figure 8 Using the jq command to extract specific event data from the eve.json file



```
analyst@2459c24082f0:~$ jq "select(.flow_id==X)" /var/log/suricata/eve.json
jq: error: X/0 is not defined at <top-level>, line 1:
select(.flow_id==X)
jq: 1 compile error
analyst@2459c24082f0:~$
```

Figure 9 Using the jq command to display all event logs related to a specific flow_id from the eve.json file. The flow_id value is a 16-digit number and will vary for each of the log entries. Replace X with any of the flow_id values returned by the previous query