

Challenge: [GitTheGate Lab](#)

Platform: CyberDefenders

Category: Threat Hunting

Difficulty: Medium

Tools Used: ELK

Summary: This lab involved investigating a brute-force and remote code execution (RCE) attack targeting an Elastic Stack deployment. The purpose of this lab is to test your knowledge on Elk, it requires you to run a bunch of different queries covering multiple operating systems and indexes. In all honesty, I didn't find it that enjoyable, however, it was great practice

Scenario: Overnight we've had an attack on our network, we have two devices in the cloud and it appears both have been compromised.

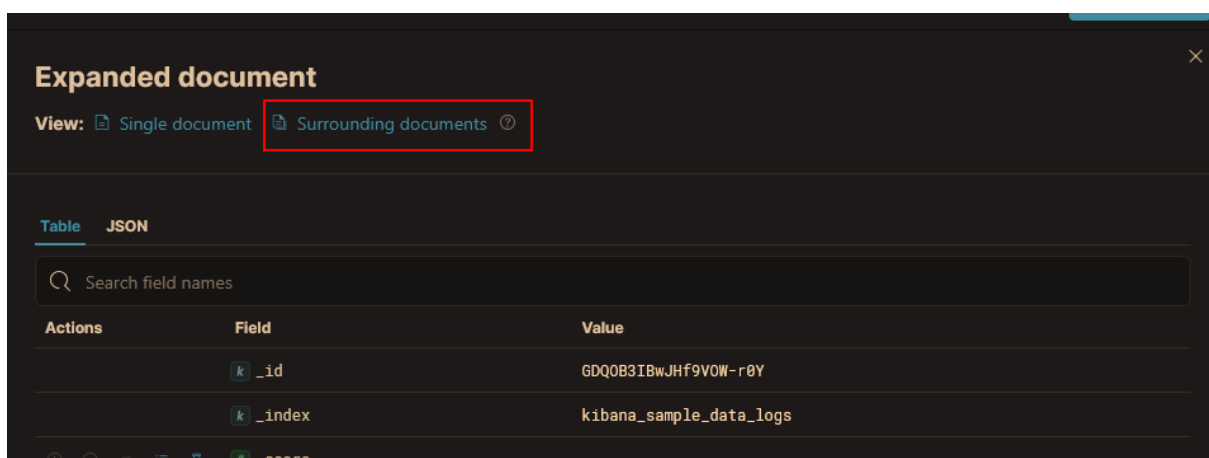
The attack appears to have taken place on the 25th of May between 9 am and 11:30 am. Our network is composed of one box that is front-facing with an SSH port open to the web and a second server behind it running an old Elastic Stack. As a soc analyst recover the information requested in these challenges so we can piece together what happened.

Using the "View Surrounding Documents" option, find the ID of the document that is 14 documents before (older) the id GDQOB3IBwJHf9VOW-r0Y?

For context, in Elastic, a document is a set of fields, which are key-value pairs that contain your data. Each document has a unique ID. To query for this ID, use the following query:


- `_id : "GDQOB3IBwJHf9VOW-r0Y"`

If you expand this document, we can click the "Surrounding documents" button:



Make sure to Load 14 older documents:

Load 14 older documents

Field	Value
 _id	tzQOB3IBwJHf9VOW-Lyd

Answer: tzQOB3IBwJHf9VOW-Lyd

Using the "View Surrounding Documents" option, find the IP of the document that is 16 documents after (newer) the id vDQOB3IBwJHf9VOW-Lyd?

Follow the same process as the previous question, but load 16 newer documents:

Load 16 newer documents

If you view this document, we can find the clientip:

 clientip	191.189.39.130
---	----------------

Answer: 191.189.39.130

How many requests have come from the IP address 2.49.53.218 between the 6th of May and the 13th of May? (time is in UTC)

Using the following query:

- `ip : 2.49.53.218 AND @timestamp >= "2020-05-06T00:00:00.000Z" and @timestamp <= "2020-05-13T23:59:59.999Z"`

We can hunt for all requests from 2.49.53.218 between the 6th of May and the 13th of May 2020.

7 hits

Answer: 7

What percentage of logs are from windows 8 machines on the 11th of May? (time is in UTC)

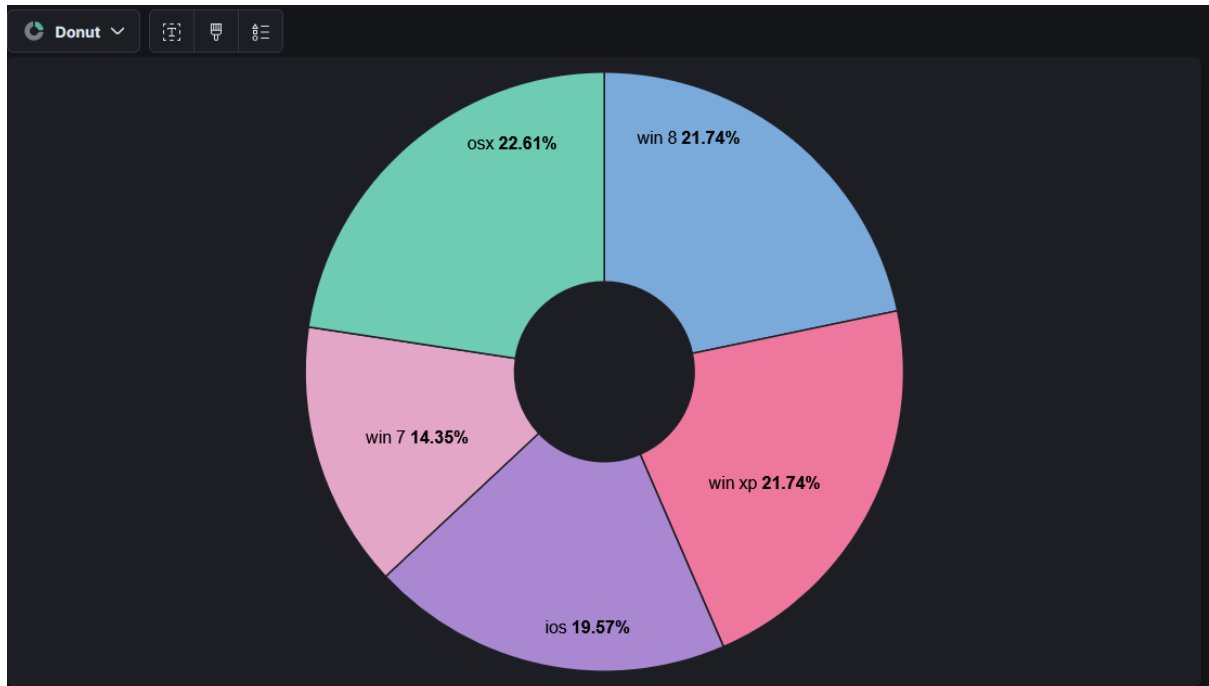
The following query hunts for all logs on the 11th of May:

- `@timestamp >= "2020-05-11T00:00:00.000Z" and @timestamp <= "2020-05-11T23:59:59.999Z"`

This results in 230 logs. Using the following query:

- `machine.os : "win 8" AND @timestamp >= "2020-05-11T00:00:00.000Z" and @timestamp <= "2020-05-11T23:59:59.999Z"`

We can see that Windows 8 machines generated 50 of these logs. You can calculate it from this. Alternatively, after filtering for all logs on the 11th of May, you can visualise the `machine.os.keyword` field to find the percentage of logs that are from Windows 8 machines:



Answer: 21.74%

How many 503 errors were there on the 8th of May? (time is in UTC)

The following query hunts for all 503 responses on the 8th of may:

- `@timestamp >= "2020-05-08T00:00:00.000Z" and @timestamp <= "2020-05-08T23:59:59.999Z" AND response : "503"`

8 hits

Answer: 8

How many connections to the host "www.elastic.co" were made on the 12th of May? (time is in UTC)

The following query hunts for all requests made to “www.elastic.co” on the 12th of May:

- `@timestamp >= "2020-05-12T00:00:00.000Z" and @timestamp <= "2020-05-12T23:59:59.999Z" AND host : "www.elastic.co"`

82 hits

Answer: 82

What is the second most common extension of files being accessed on the 12th of May? (time is in UTC)

The following query finds all events on the 12th of May:

- `@timestamp >= "2020-05-12T00:00:00.000Z" and @timestamp <= "2020-05-12T23:59:59.999Z"`

If you visualise the `extension.keyword` field, we can see what the second most common extension of files being accessed is:

Top 5 values of extension.keyword	Count of records
(empty)	89
css	40
gz	37

Answer: .gz

Find the first IP address to connect to the host `elastic-elastic-elastic.org` on the 12th of May. (time is in UTC)

Using the following query:

- `@timestamp >= "2020-05-12T00:00:00.000Z" AND @timestamp <= "2020-05-12T23:59:59.999Z" AND host : "elastic-elastic-elastic.org"`

We can hunt for all hosts that have connected to the host “elastic-elastic-elastic.org” on the 12th of May:

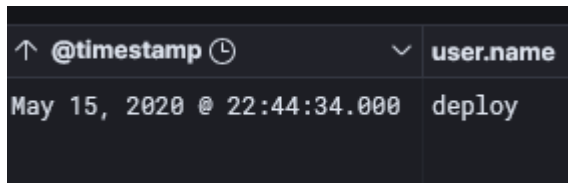
@timestamp	clientip
May 12, 2020 @ 09:21:13.279	114.246.225.218

Answer: 114.246.225.218

What was the username used that failed to log in on the 15th of May at 10:44 pm? (time is in UTC)

The following query hunts for failed authentication attempts on the 15th of May at 10:44 pm, make sure to be in the auditbeat index:

- `@timestamp >= "2020-05-15T22:44:00.000Z" and @timestamp <= "2020-05-15T22:44:59.999Z" AND event.category : "authentication" AND event.outcome : "failure"`

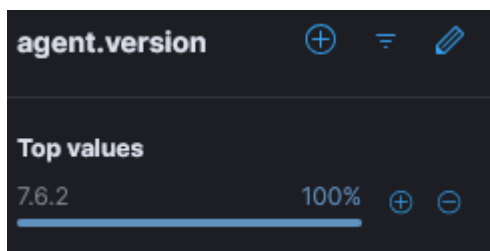


↑ @timestamp ⌚	user.name
May 15, 2020 @ 22:44:34.000	deploy

Answer: deploy

According to the logs, which vulnerable version of Kibana was identified as running in the stack?

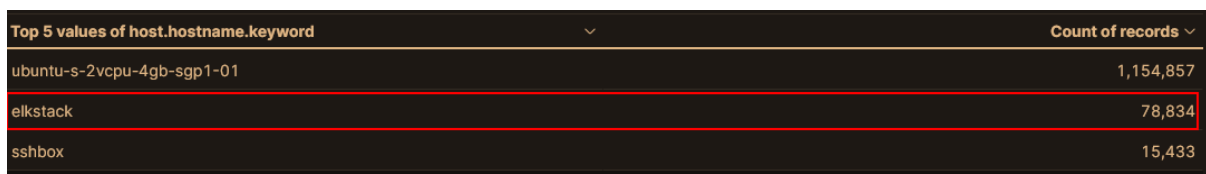
If you inspect any of the logs, you can find the agent.version number:



Answer: 7.6.2

Using current data in the auditbeat index, what is the name of the elasticsearch node? (one word)

If you visualise the host.hostname.keyword field, you can see a host called “elkstack”:



Top 5 values of host.hostname.keyword	Count of records
ubuntu-s-2vcpu-4gb-sgp1-01	1,154,857
elkstack	78,834
sshbox	15,433

Answer: elkstack

What is the name of the beat to collect windows logs? (one word)

Winlogbeat is responsible for shipping Windows event logs to Elasticsearch or Logstash.

Answer: winlogbeat

What is the name of the beat that sends network data? (one word)

Packetbeat is a network packet analyser that sends data from hosts and containers to Elasticsearch or Logstash.

Answer: packetbeat

How many fields are in the auditbeat-* index pattern?

If you navigate to Management > Kibana > Data Views > auditbeat-*, we can see there are 437 fields:

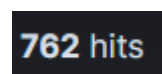


Answer: 437

On the 14th of May, how many failed authentication attempts did the host server receive? (time is in UTC)

The following query identified failed authentication attempts that occurred on the 14th of May:

- `@timestamp >= "2020-05-14T00:00:00.000Z" and @timestamp <= "2020-05-14T23:59:59.999Z" AND event.category : "authentication" AND event.outcome : "failure"`



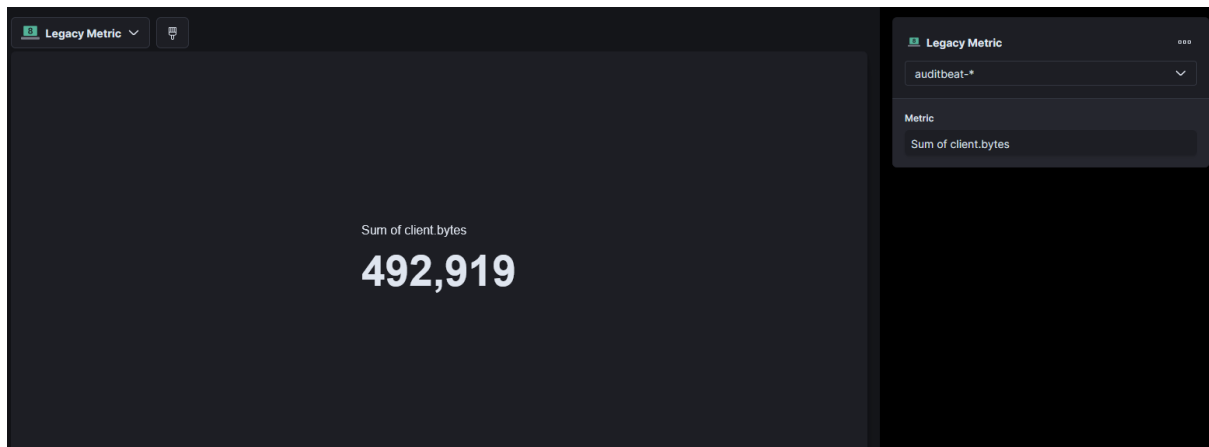
Answer: 762

On the 13th and 14th of May, how many bytes were received by the source IP 159.89.203.214 (time is in UTC)

Using the following query:

- `@timestamp >= "2020-05-13T00:00:00.000Z" and @timestamp <= "2020-05-14T23:59:59.999Z" AND client.ip : 159.89.203.214`

We can find all traffic associated with 159.89.203.214 on the 13th and 14th of May. If you visualise the client.bytes field, and sum it, you can find the total number of bytes received by 159.89.203.214:



Answer: 492,919

What username did they crack?

The following query looks for failed authentication attempts:

- `event.category : "authentication" AND event.outcome : "failure"`

If you visualise the user.name field, we can see a lot of failed authentication attempts to multiple users:

Top 100 values of user.name.keyword	Count of records
(invalid user)	2,510
(unknown user)	2,510
root	895
johnny	618
admin	417
huawei	254
user	207
support	102
test	84
guest	79

The last failed authentication occurred on the 25th of May, 2020 at 11:39:31 for user johnny by 134.122.125.130:

@timestamp	user.name	source.ip
May 25, 2020 @ 11:39:31.000	johnny	134.122.125.130

Using this query:

- `event.category : "authentication" AND event.outcome : "success" AND source.ip : 134.122.125.130`

We can see a successful authentication attempt occur at 11:50:13:

↓ @timestamp	user.name	source.ip
May 25, 2020 @ 11:50:13.111	johnny	134.122.125.130

If you investigate this IP further, we can see that it made 608 failed authentication attempts, all targeting the user “johnny”.

Answer: johnny

What host was attacked?

The brute force attack identified previously all targeted the sshbox user:



Answer: sshbox

How many were failed attempts made on the machine?

- `event.category : "authentication" AND event.outcome : "failure" AND host.hostname : "sshbox"`

Here we can find 12,523 failed authentication attempts targeting the sshbox host:

12,523 hits

Answer: 12523

What time was the last failed attempted login?

Using the same query as the previous question, if you filter the @timestamp field by New-old, you can find the last failed log attempt:

↓ @timestamp 🕒
May 25, 2020 @ 11:39:31.000

Answer: 11:39:31

What time did the attacker successfully login?

The last failed authentication event occurred at 11:39:31 on May 25th, 2020 from 134.122.125.130. Using the following filter:

- `event.category : "authentication" AND event.outcome : "success" AND host.hostname : "sshbox"`

We can see a successful authentication from 134.122.125.130 at 11:50:13:

↓ @timestamp 🕒	source.ip	user.name
May 25, 2020 @ 11:50:13.111	134.122.125.130	johnny

Answer: 11:50:13

What tool did the attacker use to get the exploit onto the machine?

I used the following query to look for process creation events:

- `host.hostname : "sshbox" AND process.name : *`

at 12:45:32, we can see Git being used to download a file called CVE-2019-7609.git:

↓ @timestamp 🕒	process.name	process.args
May 25, 2020 @ 12:45:15.886	git-remote-https	[/usr/lib/git-core/git-remote-https, origin, https://github.com/LandGrey/CVE-2019-7609.git]

CVE-2019-7609 is an arbitrary code execution vulnerability in Kibana.

Answer: git

Shortly after getting the exploit on the machine, the attacker used vim to create a file. What is the name of that file?

Using the following query:

- `host.hostname : "sshbox" AND process.name : "vim"`

We can see vim being used to create a file called ElasticCTFisFun!:

↓ @timestamp 🕒	process.name	process.args
May 25, 2020 @ 12:37:17.193	vim	[vim, ElasticCTFisFun!]
May 25, 2020 @ 12:37:07.193	vim	[vim, ElasticCTFisFun!]

Answer: ElasticCTFisFun!

What is the filename of the exploit that was run?

I used the following query to look for process creation events:

- `host.hostname : "sshbox" AND process.name : *`

You will eventually find python2 executing a script called CVE-2019-7609-kibana-rce.py:

May 25, 2020 @ 12:44:43.886	python2	[python2, CVE-2019-7609-kibana-rce.py, -u, http://10.116.0.3:5601, -host, ...]
May 25, 2020 @ 12:44:43.886	python2	[python2, CVE-2019-7609-kibana-rce.py, -u, http://10.116.0.3:5601, -host, ...]
May 25, 2020 @ 12:44:43.886	python2	[python2, CVE-2019-7609-kibana-rce.py, -u, http://10.116.0.3:5601, -host, ...]
May 25, 2020 @ 12:44:38.886	python2	[python2, CVE-2019-7609-kibana-rce.py, -u, http://10.116.0.3:5601, -host, ...]
May 25, 2020 @ 12:44:38.886	python2	[python2, CVE-2019-7609-kibana-rce.py, -u, http://10.116.0.3:5601, -host, ...]

Answer: CVE-2019-7609-kibana-rce.py

What is the first ID of the log that shows the exploit being run?

In the filebeat index, if you run the following query:

- `host.hostname : "sshbox" AND *CVE-2019-7609-kibana-rce.py*`

You can find the first ID of the log that shows the exploit being run:

Field	Value
<code>_id</code>	<code>_SHbS3IBCEolQs9lAD3z</code>

Answer: `_SHbS3IBCEolQs9lAD3z`

What parameter turned the script from testing to exploiting?

If you go to the GitHub repo for this exploit, we can see that the `--shell` option turns the script from testing to exploiting:

```
# python2 CVE-2019-7609-kibana-rce.py -h

usage: CVE-2019-7609-kibana-rce.py [-h] [-u URL] [-host REMOTE_HOST]
                                   [-port REMOTE_PORT] [--shell]

optional arguments:
  -h, --help            show this help message and exit
  -u URL                such as: http://127.0.0.1:5601
  -host REMOTE_HOST     reverse shell remote host: such as: 1.1.1.1
  -port REMOTE_PORT     reverse shell remote port: such as: 8888
  --shell               reverse shell after verify
```

Answer: `--shell`

Determining the destination IP is key to tracing the attacker's actions. What is the destination IP address where the malicious shell was sent?

Following execution of the exploit, we can see netcat being used to create a listener on port 8888. If you view these logs, we can see the server IP is 10.116.0.2:

May 25, 2020 @ 12:45:15.886	nc	[nc, -lvp, 8888]	10.116.0.2
-----------------------------	----	------------------	------------

Answer: 10.116.0.2

Identifying new users is vital to uncovering unauthorized access. What was the name of the user they created?

Using the following query, we can hunt for user creation events:

- `process.name : "useradd"`

Here we can find a new user called "Thanks4Playing" being created on the elkstack machine:

@timestamp	process.name	process.args	agent.hostname
May 25, 2020 @ 13:07:26.794	useradd	[useradd, Thanks4Playing]	elkstack

Answer: Thanks4Playing