



# Cybersecurity

## Module 15 Challenge Submission File

### Testing Web Applications for Vulnerabilities

Make a copy of this document to work in, and then respond to each question below the prompt. Save and submit this completed file as your Challenge deliverable.

#### **Web Application 1: Your Wish is My Command Injection**

Provide a screenshot confirming that you successfully completed this exploit:

Vulnerability: Command Injection

Ping a device

Enter an IP address:  Submit

```
PING 127.0.0.1 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.040 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.047 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.062 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.063 ms
--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.040/0.053/0.063/0.000 ms
root:x:0:root:/root:/bin/bash
daemon:x:1:daemon:/usr/sbin/nologin
bin:x:2:bin:/bin:/sbin/nologin
sys:x:3:sys:/dev:/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/bin/false
mysqld:x:101:101:MySQL Server,,,:/nonexistent:/bin/false
```

Vulnerability: Command Injection

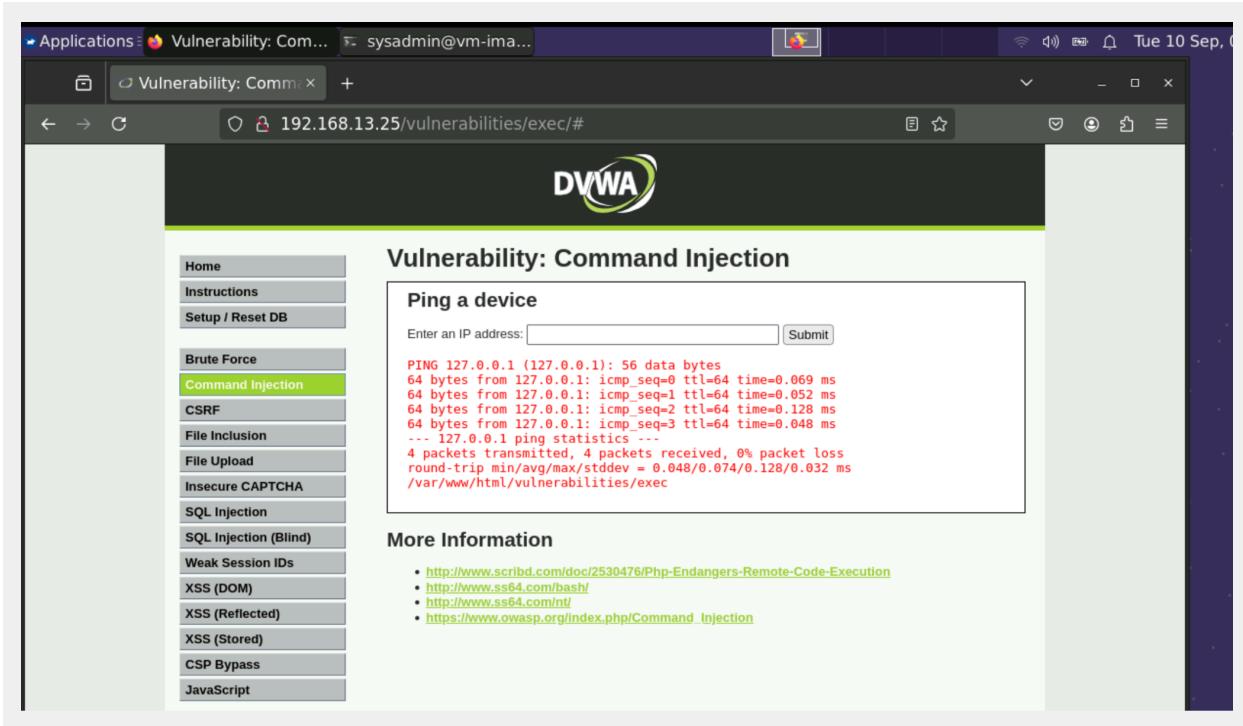
Ping a device

Enter an IP address:  Submit

```
PING 127.0.0.1 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.044 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.058 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.044 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.057 ms
--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.044/0.051/0.058/0.000 ms
127.0.0.1 ::1
::1::1 localhost ip6-localhost ip6-loopback
ff00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
192.168.13.25 82dbedb6d23f
```

More Information

- <http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution>
- <http://www.ss64.com/bash/>
- <http://www.ss64.com/m/>
- [https://www.owasp.org/index.php/Command\\_Injection](https://www.owasp.org/index.php/Command_Injection)



Write two or three sentences outlining mitigation strategies for this vulnerability:

1. Server-side validation that does not allow selection of unintended files.
2. Segregation of confidential files from the web server and accessible directories.
3. Permissions to restrict web server account accessibility.

## Web Application 2: A Brute Force to Be Reckoned With

Provide a screenshot confirming that you successfully completed this exploit:

Applications Burp Suite Commu... bWAPP - Broken Au... Terminal Thu 12 Sep, 22:18 sysadr

Burp Suite Community Edition v2023.3.3 - Temporary Project

Burp Project Intruder Repeater Window Help

Dashboard Target **Proxy** Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Extensions Learn

1 x 2 x +

Positions **Payloads** Resource pool Settings

**Payload sets**

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set: 1 Payload count: 10

Payload type: Simple list Request count: 0

**Start attack**

**Payload settings [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

Paste Load ... Remove Clear Deduplicate

1. superman  
2. loislane  
3. spiderman  
4. jennylane  
5. Clark  
6. timtom  
7. peterparker  
8. clarkkent  
9. michaelshith  
10. henryhacker

Add Enter a new item Add from list ... [Pro version only]

**Payload processing**

Login

192.168.13.35

ow @MME\_IT on Twitter

Burp Suite Community Edition v2023.3.3 - Temporary Project

Burp Project Intruder Repeater Window Help

Dashboard Target **Proxy** **Intruder** Repeater Collaborator Sequencer Decoder Comparer Logger Extensions Learn

1 x 2 x +

Positions **Payloads** Resource pool Settings

**Payload sets**

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set: 2 Payload count: 10

Payload type: Simple list Request count: 100

**Start attack**

**Payload settings [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

Paste Load ... Remove Clear Deduplicate

Up, up and away!  
Avengers Assemble!  
Cowabunga!  
Here I come to Save the Day  
With great power comes great responsibility  
You wouldn't like me when I'm angry  
Courage is immortal  
I am Iron Man  
His Past. Our future  
Change is coming

Add Enter a new item Add from list ... [Pro version only]

**Payload processing**

You can define rules to perform various processing tasks on each payload before it is used.

Add ... Rule

The screenshot displays two windows from the Burp Suite interface. The top window, titled '2. Intruder attack of http://192.168.13.35 - Temporary attack - Not saved to project file', shows the 'Intruder' tab selected. It lists various payloads (Payload 1 and Payload 2) for requests 70 through 84. Payload 75 ('tonystark') is highlighted. The bottom window, also titled '2. Intruder attack of http://192.168.13.35 - Temporary attack - Not saved to project file', shows the captured response. The response body contains HTML code indicating a successful login for Iron Man.

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	Comment
70	henryhacker	Courage is immortal	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
71	superman	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
72	loislane	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
73	spiderman	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
74	jennyjones	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
75	tonystark	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11827	
76	tintinom	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
77	peterparker	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
78	clarkkent	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
79	michaelsmith	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
80	henryhacker	I am Iron Man	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
81	superman	His Past. Our future	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
82	loislane	His Past. Our future	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
83	spiderman	His Past. Our future	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	
84	jennyjones	His Past. Our future	200	<input type="checkbox"/>	<input type="checkbox"/>	11801	

Request Response

```

Pretty Raw Hex Render
Pretty
</form>
<br>
<font color="green">
Successful login! You really are Iron Man :(
</font>
</div>
<div id="side">

```

Raw

0 matches

Request Response

```

Pretty Raw Hex Render
Pretty
Bugs Change Password Create User Set Security Level Reset Credits Blog Logout Welcome Bee

```

Raw

0 matches

Request Response

```

Pretty Raw Hex Render
Pretty
/ Broken Auth. - Insecure Login Forms /

```

Raw

0 matches

Request Response

```

Pretty Raw Hex Render
Pretty
Enter your credentials.
Login: _____
Password: _____
Login

```

Raw

0 matches

Request Response

```

Pretty Raw Hex Render
Pretty
Successful login! You really are Iron Man :(

```

Raw

0 matches

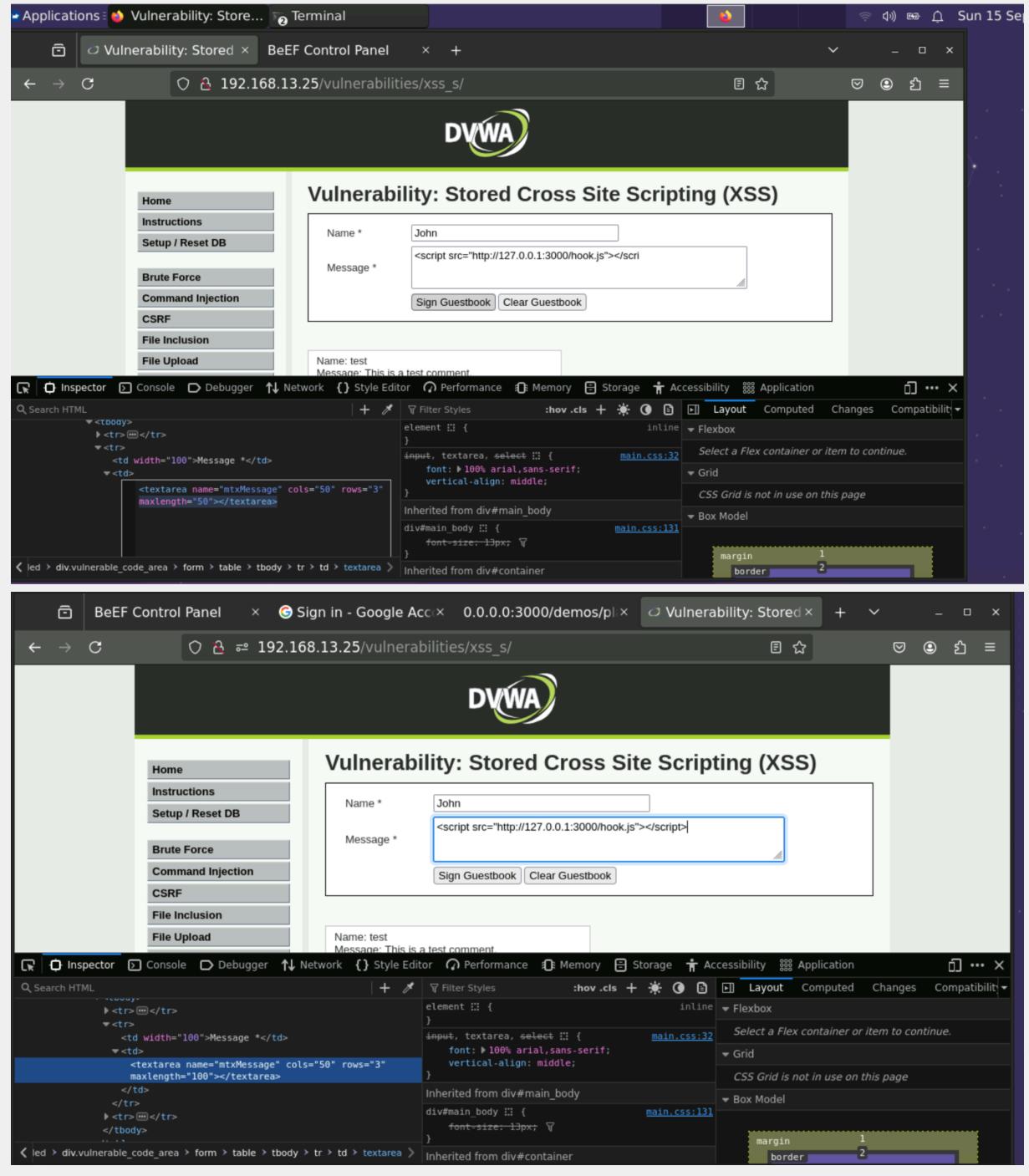
Write two or three sentences outlining mitigation strategies for this vulnerability:

Mitigations can include requiring complex usernames and passwords, using multi-factored authentication and enabling a lockout after a certain amount of failed login attempts.

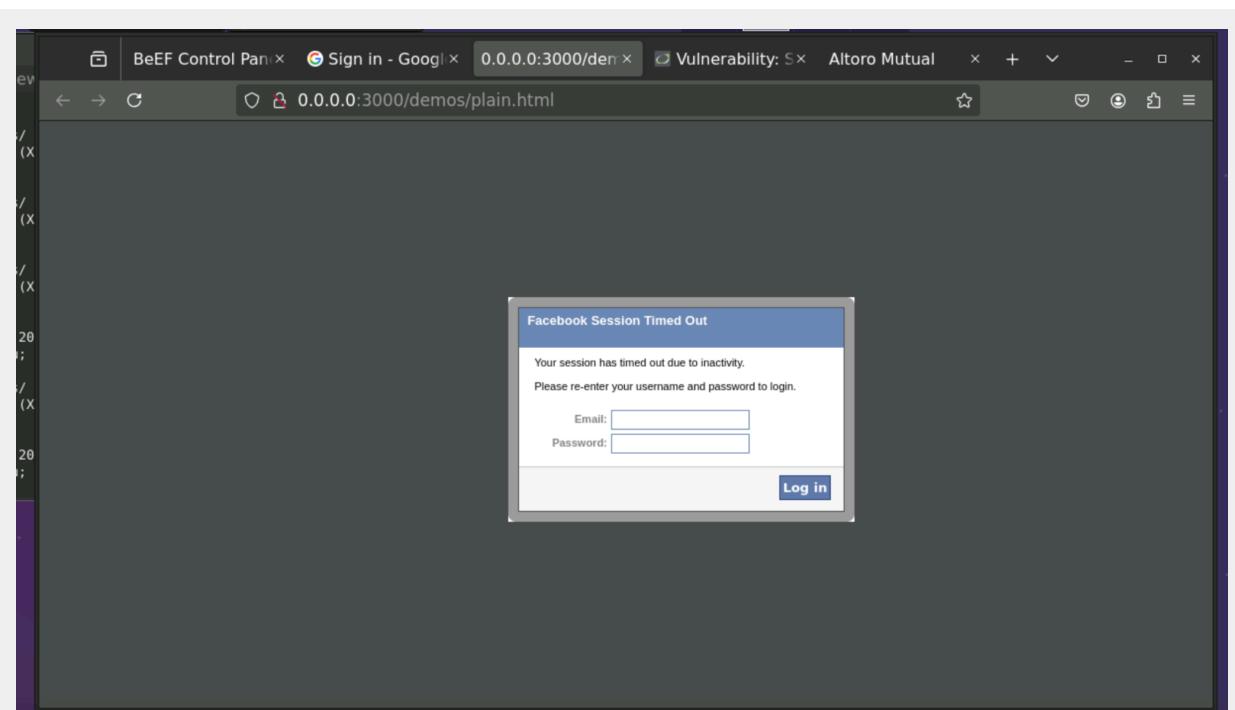
## Web Application 3: Where's the BeEF?

Provide a screenshot confirming that you successfully completed this exploit:

Firstly, I found that in the message box file, there only has maxlen=50 and I fixed it as "100" and I could type the message.



The image shows two screenshots of a web browser interface. The top screenshot displays the DVWA (Damn Vulnerable Web Application) 'Vulnerability: Stored Cross Site Scripting (XSS)' page. A user has entered 'John' in the 'Name' field and '



The screenshot shows the BeEF Control Panel interface. On the left, there is a sidebar titled 'Hooked Browsers' listing 'Online Browsers' and 'Offline Browsers' both connected to IP address 192.168.13.1. The main area has tabs for 'Getting Started', 'Logs', 'Zombies', 'Commands' (which is selected), 'Proxy', 'XssRays', and 'Network'. The 'Commands' tab displays a 'Module Tree' on the left with categories like Metasploit, Misc, Network, Persistence, Phonegap, and Social Engineering, each containing various exploit modules. To the right, there is a 'Module Results History' table and a 'Command results' table. The 'Command results' table shows two entries:

id	date	label
1	Sun Sep 15 2024 17:35:11 GMT+0000 (Coordinated Universal Time)	data: result=Notification has been displayed
2	Sun Sep 15 2024 17:35:35 GMT+0000 (Coordinated Universal Time)	data: result=Notification has been displayed

The screenshot shows two instances of the BeEF Control Panel interface. The top instance is running locally at 0.0.0.0:3000, displaying a blank page with a message about required plug-ins. The bottom instance is running on a remote host at 127.0.0.1:3000, showing a detailed view of hooked browsers and module results.

**Hooked Browsers:**

- Online Browsers:
  - ? 192.168.13.1
  - ? 192.168.13.1
- Offline Browsers:
  - ? 192.168.13.1
  - ? 192.168.13.1
  - ? 192.168.13.1

**Module Tree:**

- Search
- Browser (58)
- Chrome Extensions (6)
- Debug (9)
- Exploits (110)
- Host (24)
  - Detect Antivirus
  - Detect CUPS
  - Detect Coupon Printer
  - Detect Google Desktop
  - Get Geolocation (Third-Part)
  - Hook Default Browser
  - Get Geolocation
  - Get System Info (Java)
  - Get Wireless Keys
  - Hook Microsoft Edge
  - Get Internal IP (Java)
  - Detect Airdroid
  - Detect Default Browser
  - Detect Hewlett-Packard
  - Detect Local Drives
  - Detect Software
  - Detect Users
  - Get Battery Status
  - Get Clipboard

**Module Results History:**

id	date	label
0	2024-09-15 17:41	command 1
1	2024-09-15 17:42	command 2

**Command results:**

```
1 Sun Sep 15 2024 17:41:13 GMT+0000 (Coordinated Universal Time)
data: result={"status":"success","country":"United States","countryCode":"US","region":"VA","regionName":"Virginia","city":"Boynton","zip":2391 New_York,"isp":"Microsoft Corporation","org":"Microsoft Azure Cloud (eastus2)","as":"AS8075 Microsoft Corporation","query":"20.190.245.95"}
```

**Status:** Ready

Write two or three sentences outlining mitigation strategies for this vulnerability:

1. Input validation is a common method used to mitigate cross site-scripting.
2. Using extensions to block browser exploits.

© 2023 edX Boot Camps LLC. Confidential and Proprietary. All Rights Reserved.