Lab 09 Requirements

- Internet connectivity & VMware Workstation version 15.5.7 or above

---

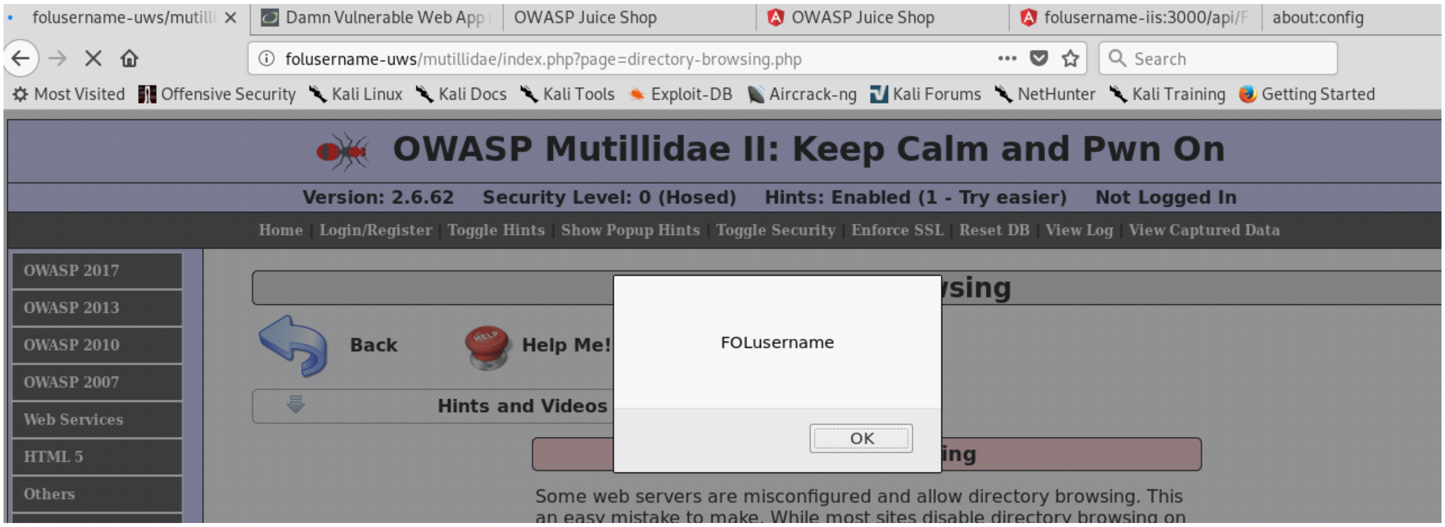
Part 01: Change the User-Agent HTTP Header in Firefox

Start up your Kali Linux VM and the Ubuntu Web Server

Ensure that you can browse from Kali's Firefox to FOLusername-uws/mutillidae

Once you have the home page loaded, turn your Burp Suite intercept on and capture the packets as you navigate to another page (I am using folusername-uws/mutilllidae/index.php?page=directory-browsing.php)

Modify the User-Agent Header in the HTTP packet to reflect your FOLusername as the page loads



FOLusername

Some web servers are misconfigured and allow directory browsing. This
an easy mistake to make. While most sites disable directory browsing on

Slide 01:

- Take a screenshot showing all of the above and place it into slide 01

You will notice that Mutillidae uses the user-agent information to populate the content in the footer on every page. Could this pose a problem?

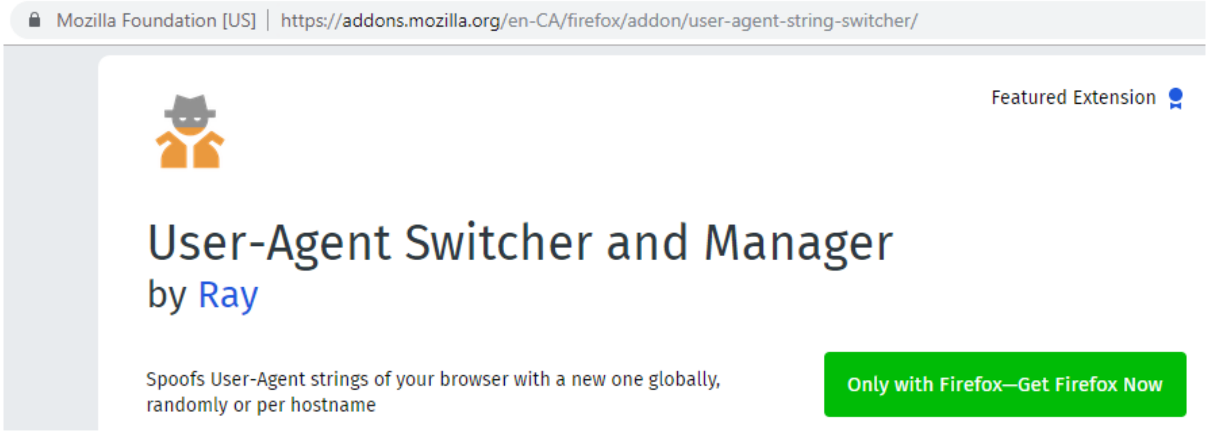
Download a Firefox add-on in Kali that will automatically change the User-Agent field in the HTTP header

I will be using the **User-Agent Switcher and Manager** by Ray for my examples, but you can feel free to use another add-on as long as it accomplishes the same task

This allows you to customize your User-Agent header field

Start by customizing it to some JavaScript code that you can inject

Intercept the packets in Burp Suite to ensure that the add-on is working properly



Mozilla Foundation [US] | https://addons.mozilla.org/en-CA/firefox/addon/user-agent-string-switcher/

Featured Extension

User-Agent Switcher and Manager
by Ray

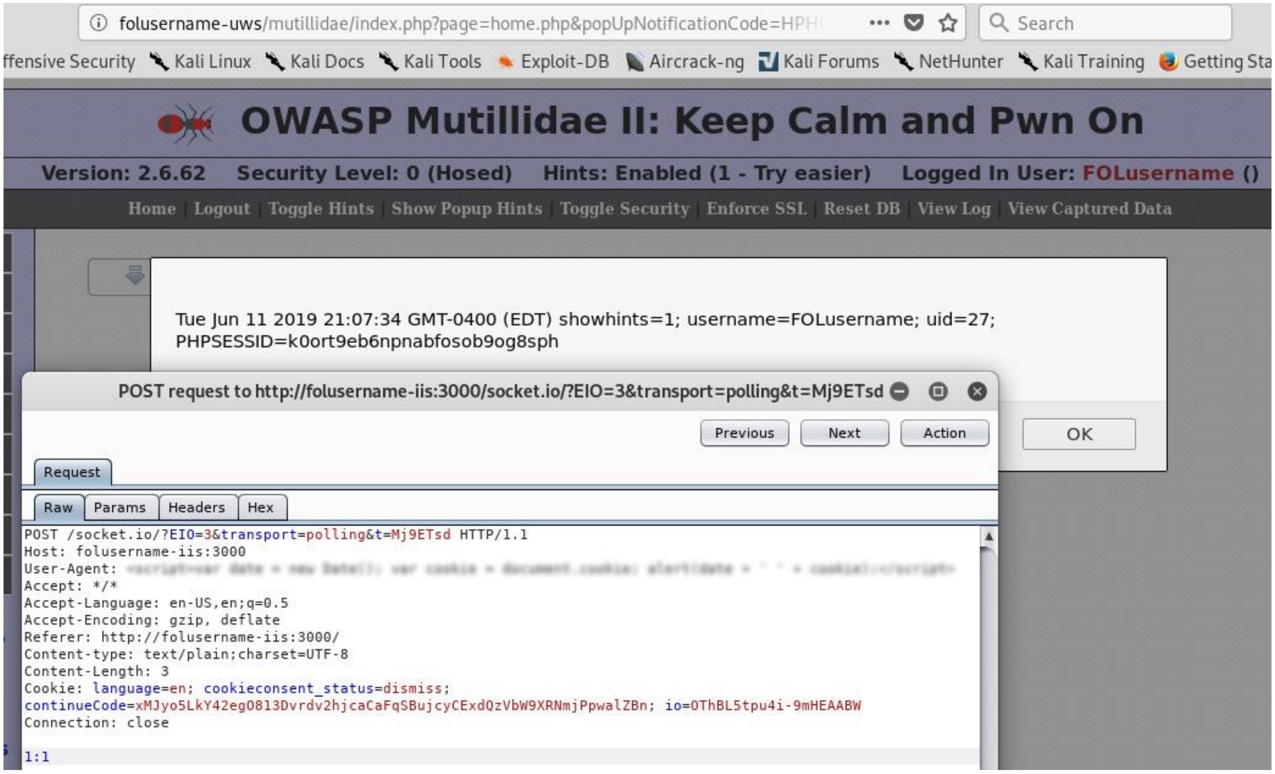
Spoofs User-Agent strings of your browser with a new one globally, randomly or per hostname

Create a new user in Mutillidae with the username of your FOLusername

Login as that user and edit the add-on to use a custom script as opposed to the default user agent information

Set the User-Agent field to use JavaScript to display today’s date and time and the cookie information for the session

Capture the **Request** packet using the **POST** method in Burp Suite



folusername-uws/mutillidae/index.php?page=home.php&popUpNotificationCode=HPH

Search
ffensive Security
Kali Linux
Kali Docs
Kali Tools
Exploit-DB
Aircrack-ng
Kali Forums
NetHunter
Kali Training
Getting Sta

OWASP Mutillidae II: Keep Calm and Pwn On

Version: 2.6.62 Security Level: 0 (Hosed) Hints: Enabled (1 - Try easier) Logged In User: FOLusername ()

Home | Logout | Toggle Hints | Show Popup Hints | Toggle Security | Enforce SSL | Reset DB | View Log | View Captured Data

Tue Jun 11 2019 21:07:34 GMT-0400 (EDT) showhints=1; username=FOLusername; uid=27; PHPSESSID=k0ort9eb6npnabfosob9og8sph

POST request to http://folusername-iis:3000/socket.io/?EIO=3&transport=polling&t=Mj9ETsd

```
POST /socket.io/?EIO=3&transport=polling&t=Mj9ETsd HTTP/1.1
Host: folusername-iis:3000
User-Agent: [redacted]
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://folusername-iis:3000/
Content-type: text/plain;charset=UTF-8
Content-Length: 3
Cookie: language=en; cookieconsent_status=dismiss;
continueCode=xMJyo5LkY42eg0813Dvrdv2hjcaCaFqSBujcyCExdQzVbW9XRNmjPpwalZBn; io=OThBL5tpu4i-9mHEAABW
Connection: close

1:1
```

Slide 02:

- Take a screenshot showing all of the above and place it into slide 02

Part 02: Skipfish

On Kali Linux, change to the /usr/share/skipfish directory

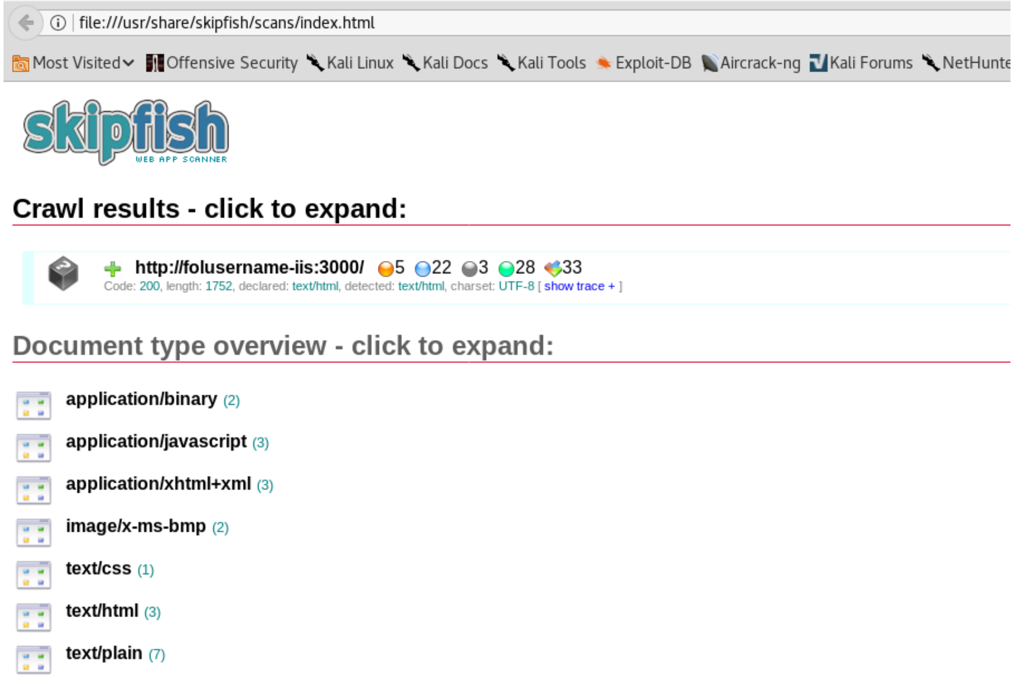
Run `skipfish –h` to see a list of available options and the proper syntax

Create a command that will:

- Use a read-write wordlist (create an empty .wl file to use, for example: **test.wl**)
- Use a supplemental read-only wordlist
- Output the results to /usr/share/skipfish/folusername/
- Use the Juice Shop URL http://folusername-iis:3000/#/search

Once you have constructed the appropriate command, run it

This scan may take some time but you can continue with the lab in a new terminal window



file:///usr/share/skipfish/scans/index.html

Crawl results - click to expand:

http://folusername-iis:3000/ 5 22 3 28 33
Code: 200, length: 1752, declared: text/html, detected: text/html, charset: UTF-8 [show trace +]

Document type overview - click to expand:

application/binary (2)
application/javascript (3)
application/xhtml+xml (3)
image/x-ms-bmp (2)
text/css (1)
text/html (3)
text/plain (7)

Slide 03:

- Take a screenshot showing the results of your skipfish command and place it into slide 03

Part 03: Golismero

Run `golismero –h` to see a list of available options. If you need to install it, use the script on FOL

Create a command that will:

- Output the results to /usr/share/skipfish/folusername/
- Use the Ubuntu Server URL http://folusername-uws/mutillidae

You may need to change limits prior to running Golismero with the command `ulimit –n 50000` first

Once you have constructed the appropriate Golismero command, run it

Slide 04:

- Take a screenshot showing the output and the golismero command you used and place it into slide 04

Part 04: Nikto

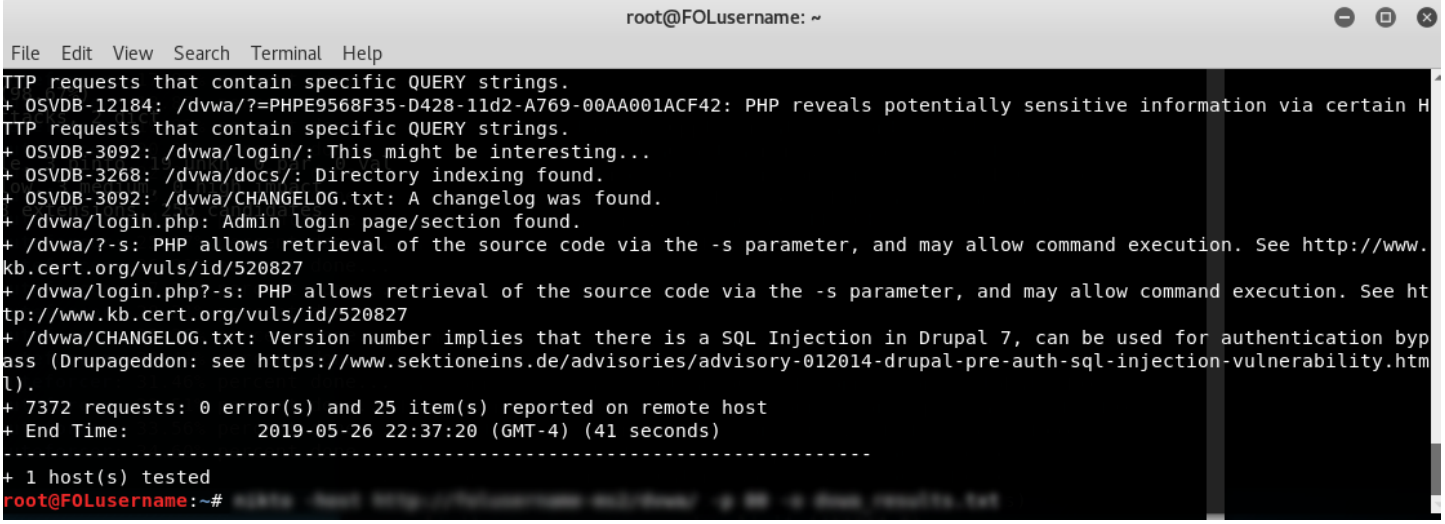
Run `nikto –h` to see a list of available options

Create a command that will:

- Write output to dvwa_results.txt
- Scan the http://folusername-ms2/dvwa/ host

Once you have constructed the appropriate command, run it

Once finished, use the up arrow to show the command you used and take a screenshot



```
TTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dvwa/?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain H
TTP requests that contain specific QUERY strings.
+ OSVDB-3092: /dvwa/login/: This might be interesting...
+ OSVDB-3268: /dvwa/docs/: Directory indexing found.
+ OSVDB-3092: /dvwa/CHANGELOG.txt: A changelog was found.
+ /dvwa/login.php: Admin login page/section found.
+ /dvwa/?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See http://www.
kb.cert.org/vuls/id/520827
+ /dvwa/login.php?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See ht
tp://www.kb.cert.org/vuls/id/520827
+ /dvwa/CHANGELOG.txt: Version number implies that there is a SQL Injection in Drupal 7, can be used for authentication byp
ass (Drupageddon: see https://www.sektioneins.de/advisories/advisory-012014-drupal-pre-auth-sql-injection-vulnerability.htm
l).
+ 7372 requests: 0 error(s) and 25 item(s) reported on remote host
+ End Time:           2019-05-26 22:37:20 (GMT-4) (41 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@FOLusername:~# [redacted]
```

Slide 05:

- Take a screenshot showing the output and the command you constructed and place it into slide 05

If you take a look at the results from Nikto, you will see that DVWA has a vulnerability with Drupageddon

What results do you get? Was the Nikto scan accurate?

Part 05: Mirror websites with HTTrack

Create a new directory **/home/kali/websites/mutillidaephish**

Start HTTrack and select the following sites to mirror:

Folusername-uws,folusername-iis,folusername-ms2

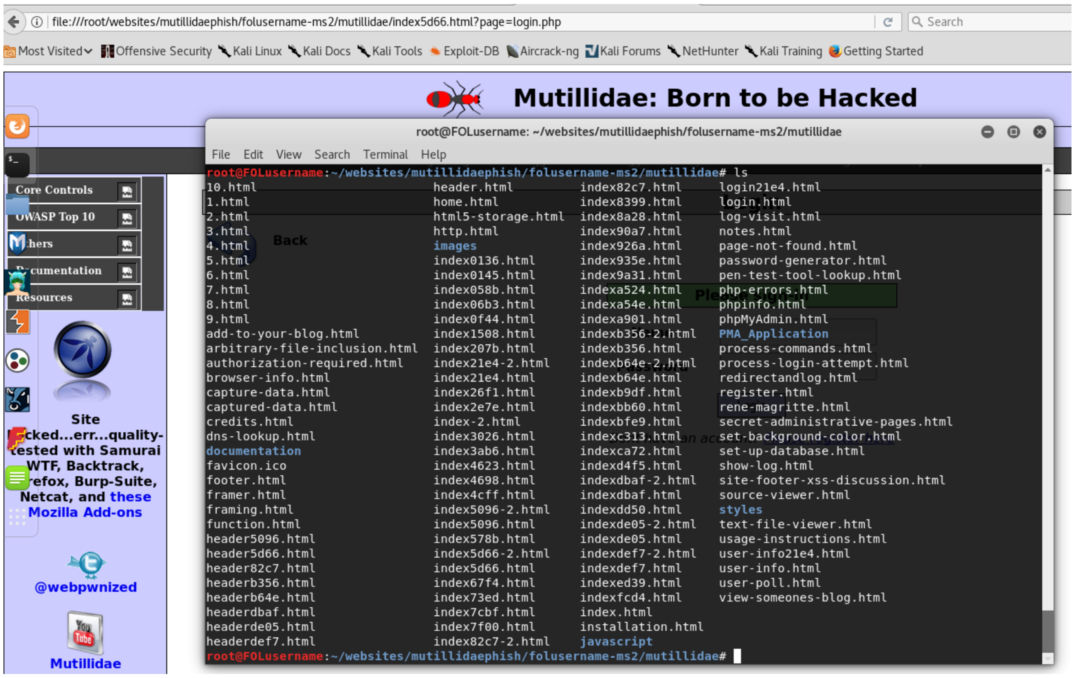
- Choose defaults
- Ready to launch the mirror? (Y/n) = SELECT y

It should now start mirroring the websites

Once it finishes, **cd** into the mutillidae directory that was pulled from folusername-ms2

Navigate to the login page on your Kali localhost and issue the **ls** command to show a list of the files that were mirrored

Check out the other two sites, on Ubuntu and IIS. What did you find there? Should you have done something differently to get a better result?


file:///root/websites/mutillidaephish/folusername-ms2/mutillidae/index5d66.html?page=login.php
Search
Most Visited
Offensive Security
Kali Linux
Kali Docs
Kali Tools
Exploit-DB
Aircrack-ng
Kali Forums
NetHunter
Kali Training
Getting Started
Mutillidae: Born to be Hacked
root@FOLusername: ~/websites/mutillidaephish/folusername-ms2/mutillidae
File
Edit
View
Search
Terminal
Help
Core Controls
OWASP Top 10
Documentation
Resources
Back
Site
Tested with Samurai WTF, Backtrack, Burp-Suite, Netcat, and these Mozilla Add-ons
@webpwnized
Mutillidae

```
root@FOLusername:~/websites/mutillidaephish/folusername-ms2/mutillidae# ls
10.html                        header.html         index82c7.html     login21e4.html
1.html                         home.html           index8399.html     login.html
2.html                         html5-storage.html  index8a28.html     log-visit.html
3.html                         http.html           index90a7.html     notes.html
4.html                         images              index926a.html     page-not-found.html
5.html                         index0136.html      index935e.html     password-generator.html
6.html                         index0145.html      index9a31.html     pen-test-tool-lookup.html
7.html                         index058b.html      indexa524.html     php-errors.html
8.html                         index06b3.html      indexa54e.html     phpinfo.html
9.html                         index0f44.html      indexa901.html     phpMyAdmin.html
add-to-your-blog.html          index1508.html      indexb356-2.html   PMA_Application
arbitrary-file-inclusion.html  index207b.html      indexb356.html     process-commands.html
authorization-required.html    index21e4-2.html    indexb64e-2.html   process-login-attempt.html
browser-info.html              index21e4.html      indexb64e.html     redirectandlog.html
capture-data.html              index26f1.html      indexb9df.html     register.html
captured-data.html             index2e7e.html      indexbb60.html     rene-magritte.html
credits.html                   index-2.html        indexbfe9.html     secret-administrative-pages.html
dns-lookup.html                index3026.html      indexc313.html     set-background-color.html
documentation                  index3ab6.html      indexca72.html     set-up-database.html
favicon.ico                    index4623.html      indexd4f5.html     show-log.html
footer.html                    index4698.html      indexdbaf-2.html   site-footer-xss-discussion.html
framer.html                    index4cff.html      indexdbaf.html     source-viewer.html
framing.html                   index5096-2.html    indexdd50.html     styles
function.html                  index5096.html      indexde05-2.html   text-file-viewer.html
header5096.html                index578b.html      indexde05.html     usage-instructions.html
header5d66.html                index5d66-2.html    indexdef7-2.html   user-info21e4.html
header82c7.html                index5d66.html      indexdef7.html     user-info.html
headerb356.html                index67f4.html      indexed39.html     user-poll.html
headerb64e.html                index73ed.html      indexfcd4.html     view-someones-blog.html
headerdbaf.html                index7cbf.html      index.html
headerde05.html                index7f00.html      installation.html
headerdef7.html                index82c7-2.html    javascript
root@FOLusername:~/websites/mutillidaephish/folusername-ms2/mutillidae# 
```

Slide 06:

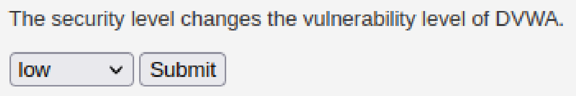
- Take a screenshot showing the files that were copied and place it into slide 06

Part 06: Challenge: Command injection in DVWA

Navigate to your MS2 using the browser: http://FOLusername-ms2

Click on DVWA from the list and login as admin/password

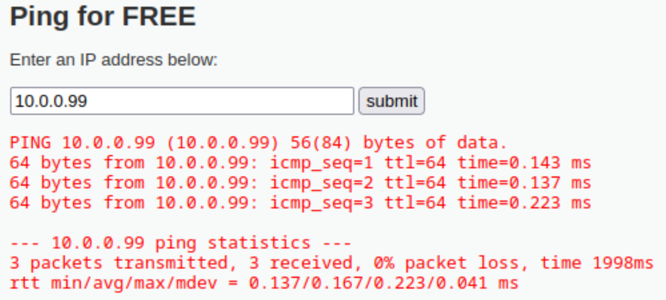
Click on **DVWA Security** and select **low** then submit


The security level changes the vulnerability level of DVWA.

Navigate to the **Command Execution** page

You will see an input field and the application allows a user to ping another system by entering an IP address. If used as intended, you will receive output based on the results of a ping…

Enter Kali's IP address into the input field and click on submit


Ping for FREE

Enter an IP address below:

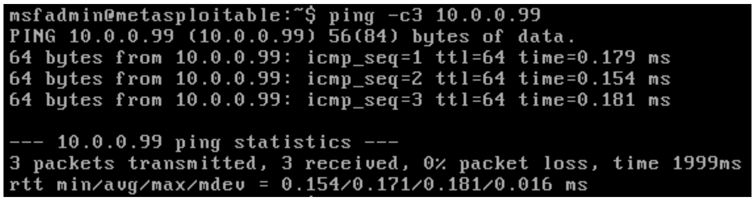
10.0.0.99

```
PING 10.0.0.99 (10.0.0.99) 56(84) bytes of data.
64 bytes from 10.0.0.99: icmp_seq=1 ttl=64 time=0.143 ms
64 bytes from 10.0.0.99: icmp_seq=2 ttl=64 time=0.137 ms
64 bytes from 10.0.0.99: icmp_seq=3 ttl=64 time=0.223 ms

--- 10.0.0.99 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.137/0.167/0.223/0.041 ms
```

Execute the following command in MS2:

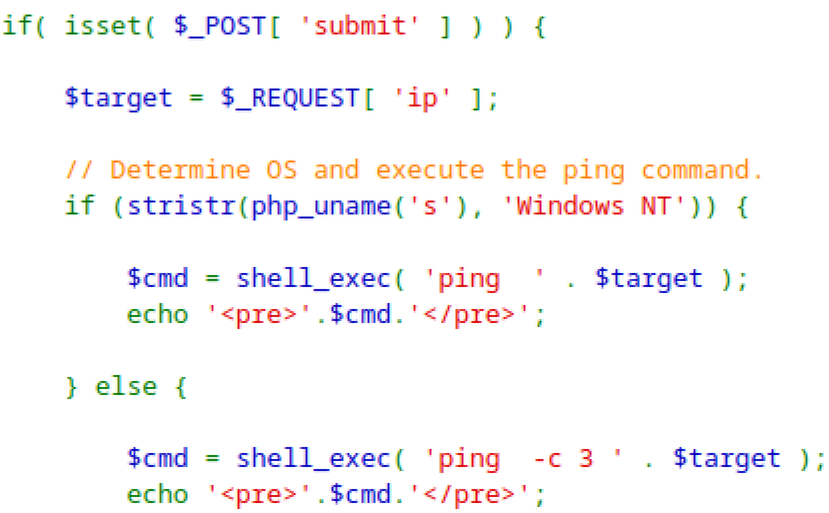
```
ping -c3 10.0.0.99
```

You will notice the same output…



```
msfadmin@metasploitable:~$ ping -c3 10.0.0.99
PING 10.0.0.99 (10.0.0.99) 56(84) bytes of data.
64 bytes from 10.0.0.99: icmp_seq=1 ttl=64 time=0.179 ms
64 bytes from 10.0.0.99: icmp_seq=2 ttl=64 time=0.154 ms
64 bytes from 10.0.0.99: icmp_seq=3 ttl=64 time=0.181 ms

--- 10.0.0.99 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.154/0.171/0.181/0.016 ms
```

This looks like the web application is passing shell commands through the web interface. Much like SQL injection, we can try to inject commands through this input field. You can click on **View Source** in the bottom right to see the logic the application is using behind this web page:

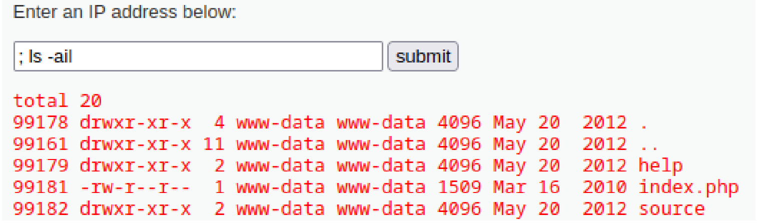


```
if( isset( $_POST[ 'submit' ] ) ) {

    $target = $_REQUEST[ 'ip' ];

    // Determine OS and execute the ping command.
    if (stristr(php_uname('s'), 'Windows NT')) {

        $cmd = shell_exec( 'ping  ' . $target );
        echo '<pre>'.$cmd.'</pre>';

    } else {

        $cmd = shell_exec( 'ping  -c 3 ' . $target );
        echo '<pre>'.$cmd.'</pre>';
```

- ➔ **if** statement checks if the form has been submitted
- ➔ Takes the value of **ip** parameter in the global variable and assigns it to **$target**
- ➔ Checks to see if the operating system is Windows
- ➔ If the OS is Windows, it executes the **ping x.x.x.x (**value of **$target)**
- ➔ If the OS is not Windows, it executes the **ping -c 3 x.x.x.x** command where the x.x.x.x value is **$target**

Looks like you can test different shell commands through this input field. When it comes to injection attacks, meta characters play an important role.

Test this out by entering semi-colon (;) and then another terminal command, such as **ls -ail** to see what results you get:

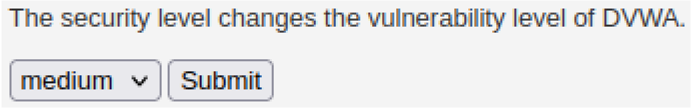

Enter an IP address below:

; ls -ail

```
total 20
99178 drwxr-xr-x  4 www-data www-data 4096 May 20  2012 .
99161 drwxr-xr-x 11 www-data www-data 4096 May 20  2012 ..
99179 drwxr-xr-x  2 www-data www-data 4096 May 20  2012 help
99181 -rw-r--r--  1 www-data www-data 1509 Mar 16  2010 index.php
99182 drwxr-xr-x  2 www-data www-data 4096 May 20  2012 source
```

If done correctly, you will see a directory listing of the current working directory. Try a few other combinations to see what other information you can obtain.

Change the security level to medium

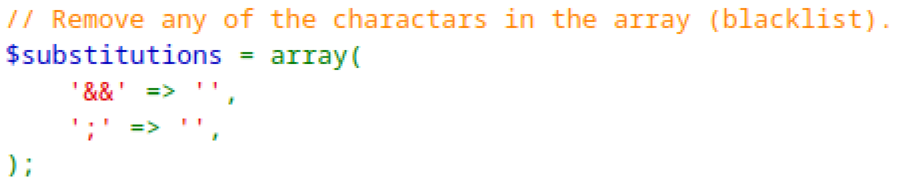

The security level changes the vulnerability level of DVWA.

medium

Navigate back to the **Command Execution** page

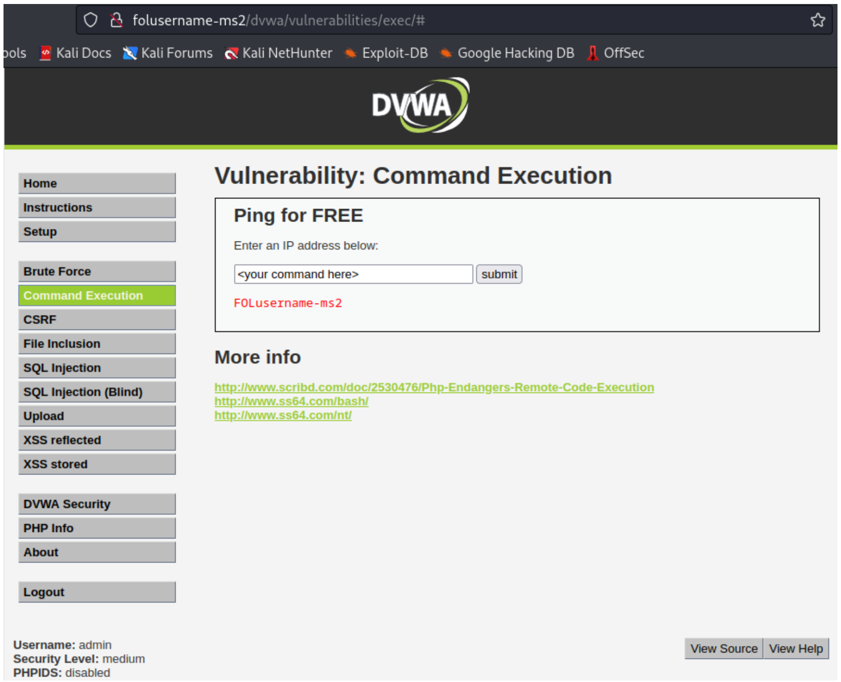
Check to see if you can get a directory listing using the same method: **; ls -ail**

You will see that it failed. Click on **View Source** in the bottom right again to check what has changed:


```
// Remove any of the charactars in the array (blacklist).
$substitutions = array(
    '&&' => '',
    ';' => '',
);
```

Looks like it is now filtering the **&&** and **;** characters from the input

CHALLENGE: Adjust your input to bypass these additional security controls and successfully return the output of the **hostname** command



Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

<your command here>

```
FOLusername-ms2
```

More info

http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
http://www.ss64.com/bash/
http://www.ss64.com/nt/

Username: admin
Security Level: medium
PHPIDS: disabled

Slide 07:
Take a screenshot showing the following and place it into slide 07:

- The command you used shown in the input field
- The output of FOLusername-ms2
- **Security Level:** medium
- URL with your FOLusername

*** Take a snapshot of all the VMs named **After Lab 09** ***