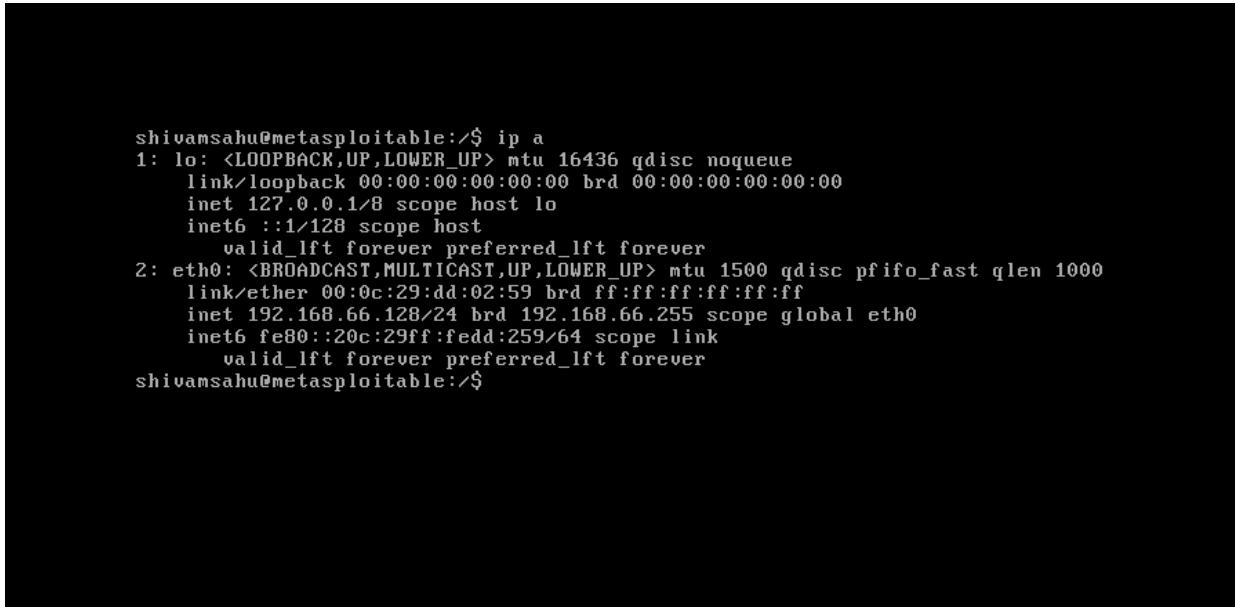


# Minor-2 Project: Metasploitable & Mutillidae II – Explanation with Screenshots

## Step 1: Metasploitable IP Configuration

A terminal window showing the output of the 'ip a' command. The output displays two network interfaces: 'lo' (loopback) and 'eth0' (ethernet). The 'lo' interface has an IPv4 address of 127.0.0.1/8 and an IPv6 address of ::1/128. The 'eth0' interface has an IPv4 address of 192.168.66.128/24 and an IPv6 address of fe80::20c:29ff:fedd:259/64. Both interfaces have a MTU of 1500 and a queueing discipline (qdisc) of pfifo\_fast. The 'eth0' interface also shows a MAC address of 00:0c:29:dd:02:59.

```
shivamsahu@metasploitable:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:dd:02:59 brd ff:ff:ff:ff:ff:ff
    inet 192.168.66.128/24 brd 192.168.66.255 scope global eth0
        inet6 fe80::20c:29ff:fedd:259/64 scope link
            valid_lft forever preferred_lft forever
shivamsahu@metasploitable:~$
```

This screenshot shows the successful network configuration of Metasploitable 2. An IPv4 address (192.168.66.128) is assigned to the eth0 interface using NAT mode. This confirms that the virtual machine can communicate with the host system. Obtaining a valid IP address is mandatory to access web applications like Mutillidae from the browser.

## Step 2: User Creation (Initial)

```
msfadmin@metasploitable:/home$ sudo adduser shivamsahu
Adding user 'shivamsahu' ...
Adding new group 'shivamsahu' (1004) ...
Adding new user 'shivamsahu' (1004) with group 'shivamsahu' ...
Creating home directory '/home/shivamsahu' ...
Copying files from '/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for shivamsahu
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [y/N] y
msfadmin@metasploitable:/home$ _
```

This image shows the creation of a new user using the adduser command. The system creates a new user, group, and home directory. User details such as password and basic information are set. This fulfills the project requirement of creating a user inside Metasploitable.

## Step 3: Mutillidae Database Error

Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 148  
Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 254  
Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 255  
Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 256

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls Login

This image shows the Mutillidae II database error. The error indicates that the required database tables do not exist. This happens because the database is not initialized by default. Identifying this error is part of the project requirement.

Here, we are shown a directory /var/www/mutillidae/ .

The directory /var/www/mutillidae/ contains all server-side files of the Mutillidae II web application. This includes PHP scripts, configuration files, and database interaction logic. Since Apache serves content from /var/www/, accessing this directory confirms that Mutillidae is correctly hosted on the web server. Any database misconfiguration directly affects files inside this directory. Analyzing this path helps in understanding how web applications are structured on Linux servers.

## Step 4: Mutillidae Database Configuration

```
msfadmin@metasploitable:~$ cd /var/www/mutillidae/  
msfadmin@metasploitable:/var/www/mutillidae$ sudo nano config.inc_
```

This screenshot shows editing of the Mutillidae configuration file. Database connection parameters such as host, user, and database name are verified. Correct configuration is necessary for proper database communication. This step helps resolve database-related issues.

## Step 5: Database Reset and OWASP Setup

The screenshot shows a terminal window with the title "GNU nano 2.0.7" and the file path "File: config.inc". The status bar indicates "Modified". The content of the file is:

```
<?php
    /* NOTE: On Samurai, the $dbpass password is "samurai" rather than blank */
    $dbhost = 'localhost';
    $dbuser = 'root';
    $dbpass = '';
    $dbname = 'owasp10';
?>
```

At the bottom of the screen, there is a menu of keyboard shortcuts:

- ^G Get Help
- ^O WriteOut
- ^R Read File
- ^Y Prev Page
- ^K Cut Text
- ^C Cur Pos
- ^X Exit
- ^J Justify
- ^W Where Is
- ^V Next Page
- ^U UnCut Text
- ^I To Spell

This image shows the database setup/reset option in Mutillidae. Resetting the database creates required tables automatically. OWASP Top 10 vulnerabilities become accessible after this step. This confirms successful database initialization.

## Step 6: Mutillidae Working Successfully

[View your details](#)

 [Back](#)

Please enter username and password  
to view account details

Name

Password

[View Account Details](#)

Dont have an account? [Please register here](#)

Username=shivam  
Password=shivam  
Signature=

This screenshot confirms that Mutillidae II is working correctly. The application loads without database errors. User data can be viewed successfully, proving full functionality. This completes the Mutillidae II fix requirement.