

# Denial of Service Practical – Slowloris Attack on DVWA (Metasploitable 2)

## Aim of the Practical

To perform a Slowloris Denial of Service attack on DVWA hosted in Metasploitable 2 and observe the impact on website network stability and availability.

---

## Description of the Practical

In this practical, a Slowloris attack is executed against the DVWA web application running on Metasploitable 2.

Slowloris is an application-layer DoS attack that opens multiple HTTP connections to the web server and keeps them half-open by sending incomplete HTTP requests. This exhausts the server's connection pool, making the website slow or unresponsive for legitimate users.

---

### Type of Attack Performed

- **Attack Name:** Slowloris
  - **Category:** Denial of Service (DoS)
  - **Attack Layer:** Application Layer (Layer 7)
  - **Target Application:** DVWA
  - **Tool Used:** Slowloris (Python-based)
- 

### Lab Setup

- **Attacker Machine:** Parrot Security OS
- **Victim Machine:** Metasploitable 2
- **Target Application:** DVWA

## **Step-by-Step Procedure**

### **Step 1: Start Metasploitable 2**

- Power on **Metasploitable 2**
  - Login to the system
  - Ensure the machine is running properly
- 

### **Step 2: Open Browser in Parrot OS**

- Start **Parrot Security OS**
- Open the browser
- Enter the following URL:

`http://<target IP>`

---

### **Step 3: Navigate to DVWA**

- From the Metasploitable web page, click on **DVWA**
  - DVWA login page will open
- 

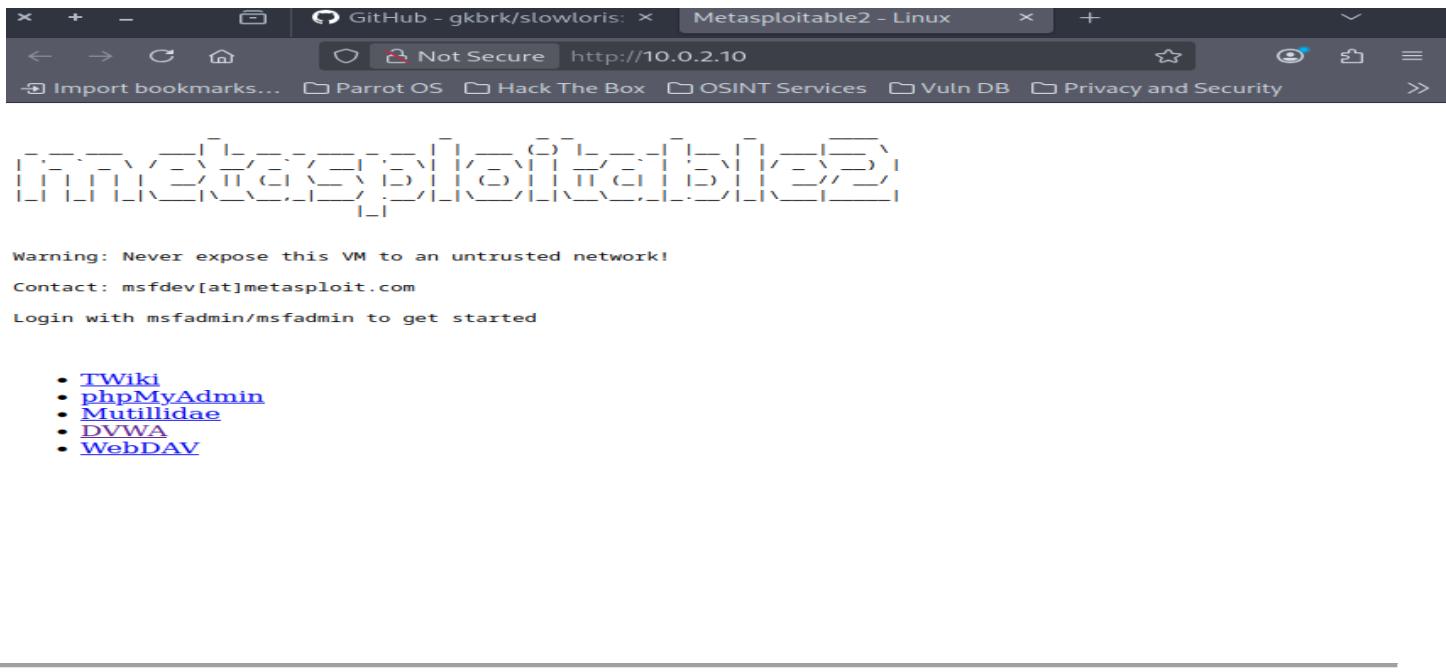
### **Step 4: Login to DVWA**

Use the default credentials:

Username: admin

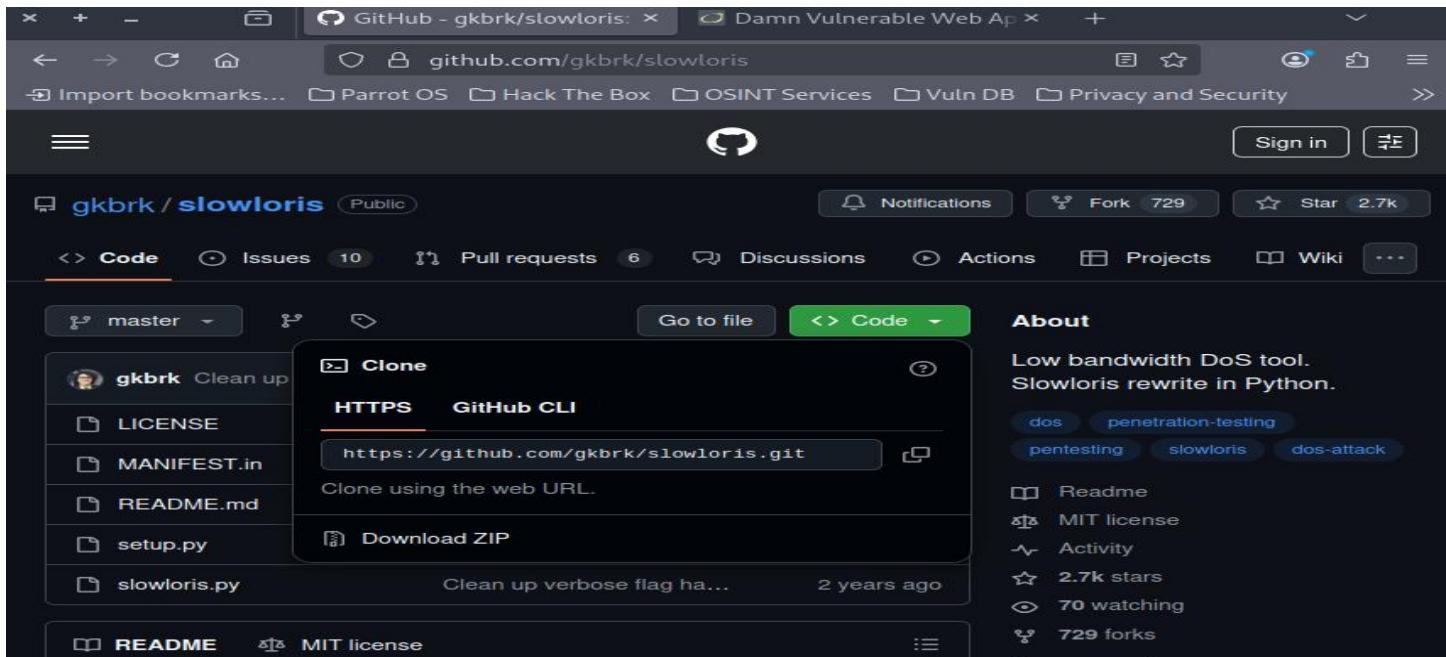
Password: password

- Login successfully
- Keep the DVWA page open



## Step 5: Search Slowloris on GitHub

- Open a new tab
- Search for **Slowloris GitHub**
- Select a trusted Slowloris repository
- Copy the **repository URL**



---

## Step 6: Clone Slowloris Repository

Open **Terminal** in Parrot OS and run:

```
git clone <slowloris_github_url>
```

---

## Step 7: Navigate to Slowloris Directory

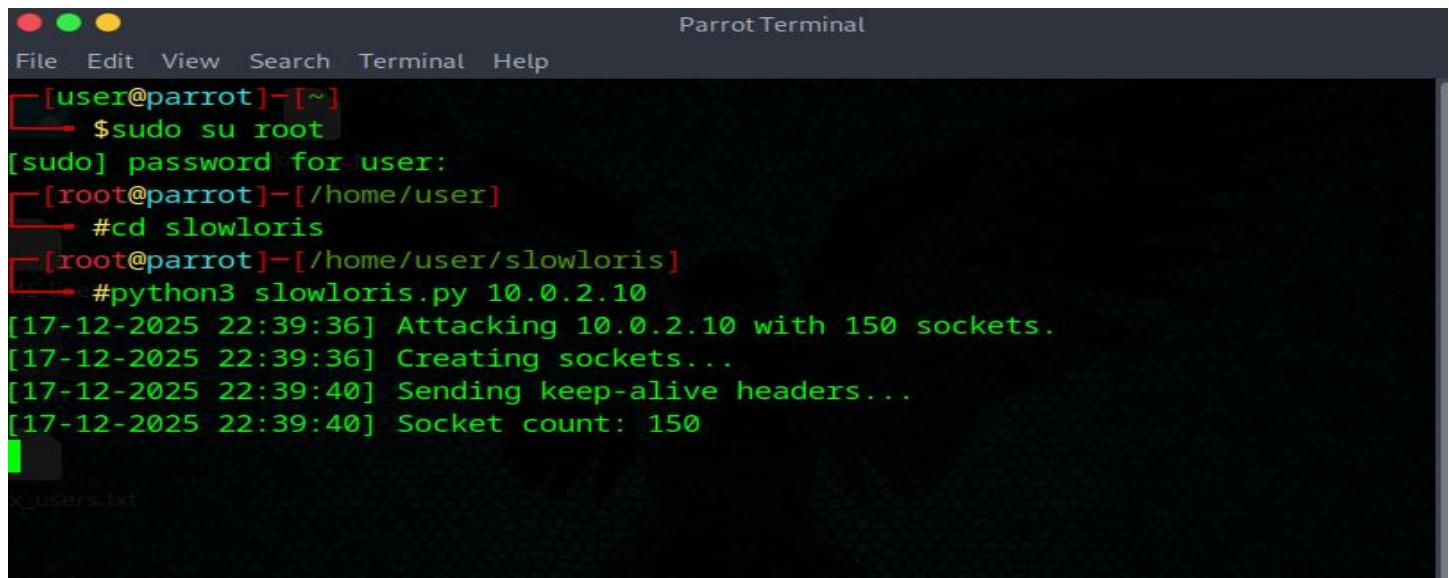
```
cd slowloris
```

---

## Step 8: Execute Slowloris Attack

Run the following command:

```
python3 slowloris.py target IP
```



The screenshot shows a terminal window titled "Parrot Terminal". The terminal window has a dark background with light-colored text. At the top, there is a menu bar with options: File, Edit, View, Search, Terminal, Help. Below the menu bar, the terminal prompt is shown: "[user@parrot]~". The user enters the command "sudo su root" to switch to root privileges. A password is required, indicated by "[sudo] password for user:". Once logged in as root, the user navigates to the directory containing the Slowloris script with the command "#cd slowloris". Finally, the user runs the attack with the command "#python3 slowloris.py 10.0.2.10". The terminal output shows the attack parameters: "[17-12-2025 22:39:36] Attacking 10.0.2.10 with 150 sockets.", "[17-12-2025 22:39:36] Creating sockets...", "[17-12-2025 22:39:40] Sending keep-alive headers...", and "[17-12-2025 22:39:40] Socket count: 150".

---

## Step 9: Observe Website Behavior

- Go back to the DVWA webpage
- Refresh the page
- Try navigating to another DVWA page
- The website loads **very slowly or becomes unstable**
- Network delay is clearly visible

The screenshot shows the DVWA homepage. The left sidebar contains a navigation menu with the following items: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected, XSS stored, DVWA Security, PHP Info, and About. The main content area features a large green banner with the text "Welcome to Damn Vulnerable Web App!". Below the banner, there is a "WARNING!" section with a detailed warning about the application's vulnerability and usage. There is also a "Disclaimer" section, a "General Instructions" section, and a note about the help button. A message at the bottom of the page states "You have logged in as 'admin'".

## Observation

- DVWA pages take longer to load
- Website becomes unstable when switching pages
- Apache server connections are exhausted
- Attack uses **low bandwidth but high impact**

## Result

The **Slowloris DoS attack** was successfully performed on **DVWA running in Metasploitable 2**.

The attack demonstrated how keeping multiple HTTP connections half-open can disrupt the availability of a vulnerable web application.

## Mitigation Techniques

- Enable mod\_reqtimeout in Apache
- Configure Web Application Firewall (WAF)
- Limit maximum concurrent connections
- Apply server hardening and patches

## Ethical Disclaimer

This practical was conducted **only in a controlled virtual lab environment** using intentionally vulnerable systems for educational purposes.