# Network Vulnerability Assessment

rootsh3ll Bank is a multinational bank that serves over 100 million customers and has over 13,000 branches worldwide.

Security is a crucial part of businesses that involves a huge amount of cash-flow on a regular basis. They need regular risk assessment on their devices and networks to stay safe from potential threats.

rootsh3ll Bank has hired you for a risk assessment on their network. Your job is to perform a risk/ vulnerability assessment for your client on their wired network from a perspective of an internal network attacker.

**Objective**:

1. Identify servers running vulnerable softwares.
2. Test and Identify web server for Heartbleed vulnerability without exploitation.
3. Learn what is a CVE Database and ID.

**TABLE OF CONTENTS**

# Flag 1

Q. Which server is running a vulnerable version of SSH on the network?
Hint: Exclude default gateway

Run ifconfig and get your IP address and subnet mask to identify the network range.

```
ifconfig
```

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.1.3.37  netmask 255.255.255.0  broadcast 10.1.3.255
        ether 02:42:0a:01:03:25  txqueuelen 0  (Ethernet)
        RX packets 103  bytes 9801 (9.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
```

Netmask `255.255.255.0` tells us that we have a `/24` IP address range for IP address starting from `10.1.3.1` to `10.1.3.255`.

Run an nmap scan on the complete network for open port 22.

```
nmap -T5 -v --open -sS -sV 10.1.3.37/24 -p 22
```

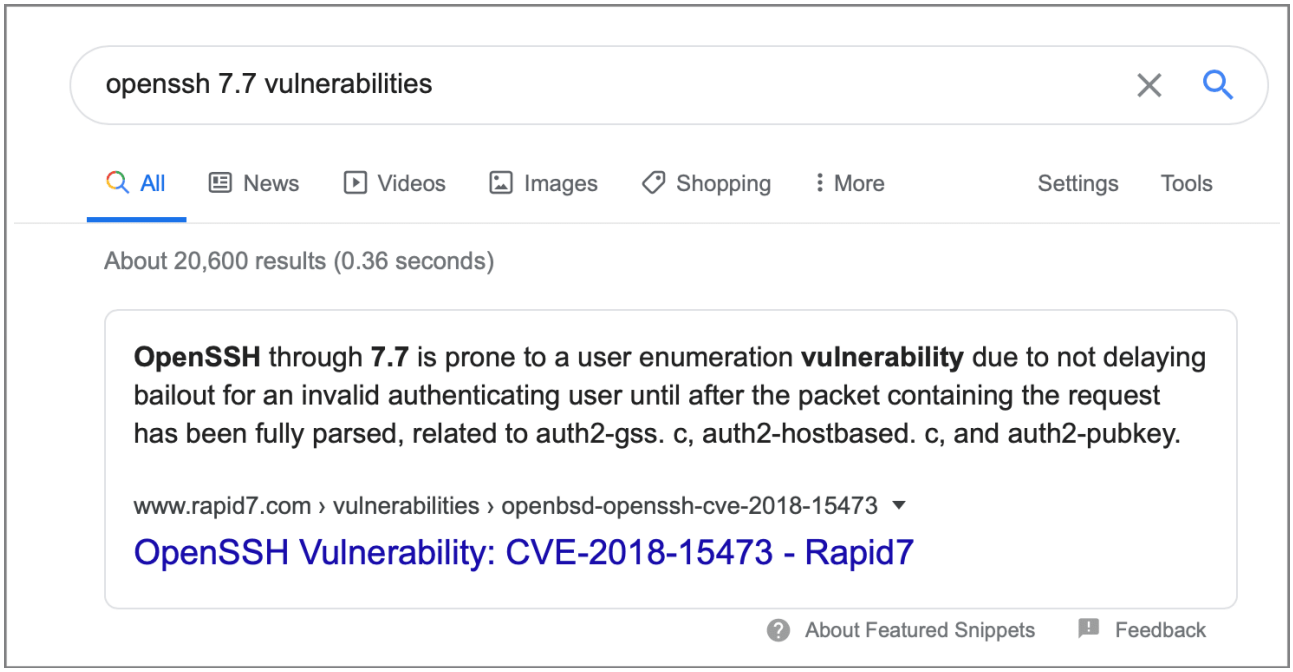
```
Nmap scan report for ip-10-1-3-1.ec2.internal (10.1.3.1)
Host is up (0.000036s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
MAC Address: 02:42:BD:B8:2D:33 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for ssh_server.lab (10.1.3.118)
Host is up (0.000011s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.7 (protocol 2.0)
MAC Address: 02:42:0A:01:03:76 (Unknown)
```

On simply searching "*openssh 7.7 vulnerabilities*" on Google, we get an article that confirms the existence of vulnerability in this specific version of SSH (Ignore SSH server on `10.1.3.1` as its default gateway)



# Flag 2

Q. What CVE ID is associated with the vulnerable version of SSH server?

From the URL we received above, we know now know that version 7.7 of OpenSSH is vulnerable to username enumeration attack. To which a CVE ID is associated, which is **CVE-2018-15473**

# Flag 3

Q. Identify web server version vulnerable to SSL Heartbleed attack.
**Hint**: use **ssltest.py** located on **/root/Desktop/**

Scan your subnet for servers running on port 443, since `Heartbleed` is an SSL vulnerability and it needs to be running on SSL/TLS port.

```
Nmap scan report for file_server.lab (10.1.3.82)
Host is up (0.000020s latency).

PORT    STATE SERVICE  VERSION
443/tcp open  ssl/http nginx 1.11.13
MAC Address: 02:42:0A:01:03:52 (Unknown)
```

Go to `/root/Desktop` and run `ssltest.py` file on the target web server IP address.

```
python ssltest.py 10.1.3.82
```

```
  3ee0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3ef0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3f90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3fa0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3fb0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  3ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................

WARNING: server returned more data than it should - server is vulnerable!
```

On successful confirmation enter the Web server's version in *Verify Flag* section of your lab.

# Flag 4

Q. Identify CVE ID for the vulnerable MySQL server on your network
Hint: A Tragically Comedic Security Flaw

Run an nmap service version scan for MySQL's default port (3306) on your entire subnet and look for any version of mysql identified by nmap.

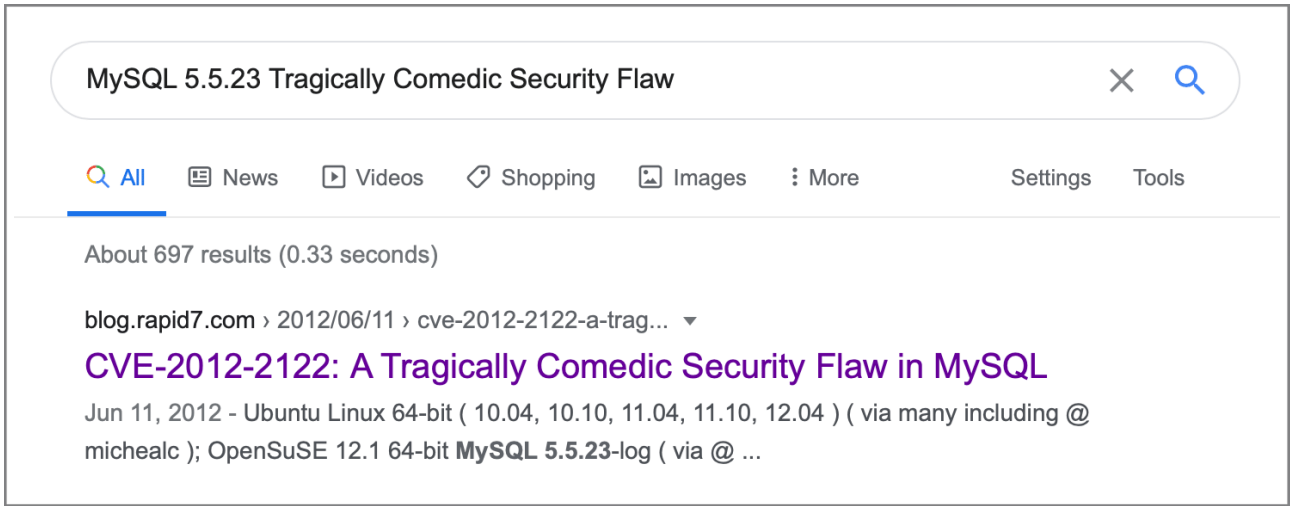
```
nmap -T5 --open -v 10.1.3.1/24 -p 3306
```

```
Nmap scan report for mysql_server.lab (10.1.3.183)
Host is up (0.000013s latency).

PORT     STATE SERVICE VERSION
3306/tcp open  mysql   MySQL 5.5.23
MAC Address: 02:42:0A:01:03:B7 (Unknown)
```

Nmap reveals that there is indeed one MySQL server running on the subnet and it's version is **5.5.23**
Let's find out if this version is known to have a vulnerability.

Combined the MySQL version with the help string we discover a URL from google, confirming the vulnerability. And it is indeed a comedic one!




We'll cover the exploitation in the following lab where you are supposed to write a custom exploit script to exploit the vulnerability. If you are new to writing exploit, this would be a great starting point for you to get your hands into exploit development.

Happy hacking!