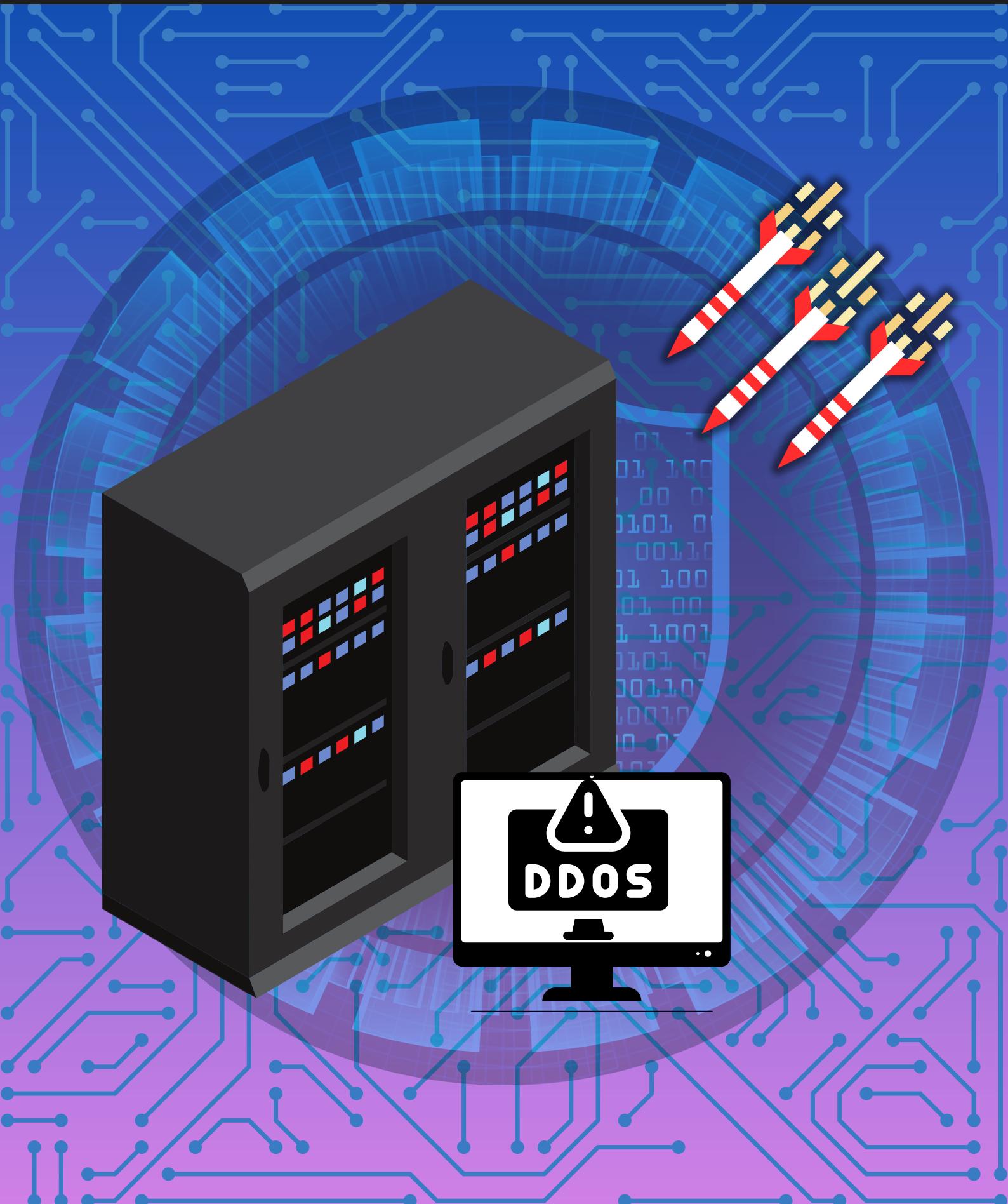


MODUL SIMULASI DDOS



Kelompok 8
Keamanan Data



BAB I

PENDAHULUAN

Ketersediaan layanan jaringan merupakan aspek krusial dalam sistem informasi modern. Salah satu kelas ancaman terhadap ketersediaan adalah DDoS (Distributed Denial of Service), keluarga serangan yang berupaya membuat layanan tidak tersedia dengan membanjiri bandwidth, sumber daya protokol, atau aplikasi target. DDoS terbagi menjadi beberapa kelas utama: volumetric attacks (menghabiskan bandwidth), protocol attacks (menyalahgunakan mekanisme protokol seperti SYN Flood yang memenuhi SYN backlog), dan application-layer / low-and-slow attacks (menargetkan aplikasi web dengan koneksi parsial atau lambat , contoh terkenal: Slowloris).

Modul ini menyajikan panduan eksperimen laboratorium yang menitikberatkan pada simulasi DDoS dalam lingkungan virtual terisolasi, dengan pengenalan tiap tipe serangan dan studi kasus mendalam pada Slowloris untuk memahami bagaimana koneksi HTTP parsial dapat menghabiskan slot koneksi server. Dengan pendekatan hands-on, peserta akan menyiapkan topologi VM terisolasi sesuai dokumen, menjalankan skenario pengujian yang terkontrol, mengumpulkan metrik sistem dan jaringan, serta menyusun analisis hasil uji yang semuanya dengan penekanan kuat pada etika dan keselamatan: pengujian hanya pada infrastruktur milik sendiri atau yang memiliki izin eksplisit, dan dilarang menargetkan sistem publik atau pihak lain tanpa izin.

BAB II

MENGENAL DDoS

Apa itu DDoS?

DDoS (Distributed Denial of Service) itu jenis serangan di dunia jaringan/komputer yang tujuannya bikin suatu layanan, server, atau website jadi nggak bisa diakses.

Caranya biasanya dengan membanjiri target pakai traffic berlebihan yang datang dari banyak sumber (komputer atau perangkat yang sudah jadi bagian botnet). Karena serangannya terdistribusi, server target jadi kewalahan, resource habis, dan akhirnya down atau sangat lambat.

Singkatnya:

- DoS = satu sumber nyerang, banjirin traffic.
- DDoS = banyak sumber (ribuan/bahkan jutaan), jauh lebih susah ditangkal.

Apa saja jenisnya?

Volumetric Attack

- Target: bandwidth jaringan.
- Contoh: UDP Flood, ICMP Flood.
- Dampak: jalur komunikasi penuh, akses normal terganggu.

Protocol Attack

- Target: kelemahan protokol & resource server.
- Contoh: SYN Flood, Ping of Death.
- Dampak: server/perangkat jaringan tidak mampu memproses permintaan.

Application Layer Attack

- Target: aplikasi/layer 7.
- Contoh: HTTP Flood, Slowloris.
- Dampak: layanan menjadi lambat atau tidak dapat diakses.

BAB II

MENGENAL DDoS

Apa saja serangan DDoS?

Contoh Serangan DDoS

- UDP Flood, penyerang mengirim paket UDP dalam jumlah besar ke port acak di server. Server dipaksa mencari aplikasi penerima yang sebenarnya tidak ada, sehingga sumber daya habis.
- ICMP Flood (Ping Flood), target dibanjiri permintaan ICMP Echo (ping) berulang kali. Akibatnya, bandwidth dan CPU server tersita hanya untuk membalas ping.
- SYN Flood, penyerang mengirim banyak permintaan TCP SYN tanpa menyelesaikan proses handshake. Server menyimpan koneksi “setengah jadi” sampai penuh, sehingga koneksi sah ditolak.
- Ping of Death, mengirim paket ICMP berukuran melebihi standar (fragmentasi), sehingga sistem target crash saat mencoba menyusunnya kembali.
- Smurf Attack, penyerang mengirim paket ICMP dengan alamat IP palsu (IP korban) ke banyak host. Semua host membalas ke korban, menyebabkan banjir traffic.
- HTTP Flood, membanjiri server web dengan permintaan HTTP GET/POST berulang. Server kewalahan memproses permintaan sehingga layanan melambat atau berhenti.
- Slowloris, menjaga koneksi HTTP tetap terbuka dengan mengirim data sangat lambat. Server menunggu penyelesaian koneksi hingga resource habis.

BAB III

CONTOH KASUS DAN SKENARIO LAB

Contoh Kasus

Salah satu contoh serangan DDoS yang sering terjadi adalah ketika sebuah situs kampus atau toko daring tiba-tiba mengalami perlambatan parah saat jam sibuk, serangan seperti itu akan direplikasi dengan menjalankan Slowloris dari satu mesin penyerang ke server web (Nginx) pada mesin korban; Slowloris membuka banyak koneksi HTTP yang tidak selesai sehingga Nginx kehabisan slot koneksi untuk melayani pengguna sah, membuat layanan menjadi lambat atau tidak dapat diakses. Kondisi seperti ini mengganggu aktivitas akademik maupun transaksi bisnis, menurunkan kepercayaan pengguna, dan berpotensi menyebabkan kerugian finansial. Skala serangan mengikuti skenario satu VM attacker terhadap satu VM victim, dan mitigasi dilakukan sesuai prosedur tim dengan penyesuaian konfigurasi Nginx sehingga layanan pulih dan kembali dapat diakses oleh pengguna.

Skenario Lab

Untuk mempelajari serangan dan mitigasinya, dilakukan simulasi dalam lingkungan laboratorium yang terisolasi. Infrastruktur terdiri atas dua mesin virtual:

1. Victim - VM Ubuntu 22.04 menjalankan Nginx (instal via sudo apt install nginx-full).
2. Attacker - VM Linux Kali — clone dan jalankan Slowloris dari GitHub.

BAB III

CONTOH KASUS DAN SKENARIO LAB

Langkah eksperimen dijalankan secara bertahap:

- Baseline: verifikasi server dalam kondisi normal — buka `http://<VICTIM_IP>` (tampil “Welcome to nginx!”), catat metrik awal (CPU, mem, response time).
- Attack: pada VM ATTACK, jalankan Slowloris (hasil clone dari <https://github.com/gkbrk/slowloris>) terhadap IP Victim dengan parameter sockets yang sesuai (contoh di dokumen: `python3 slowloris.py 192.168.56.10 10000`).
- Mitigasi: edit file konfigurasi Nginx (mis. `sudo nano /etc/nginx/nginx.conf` atau `nginx.d` seperti di dokumen) untuk menerapkan timeout/limit (contoh: `client_header_timeout`, `client_body_timeout`, `keepalive_timeout`, atau `limit_conn/limit_req`), lalu `nginx -t` dan `systemctl restart nginx`.
- Re-run Attack: jalankan Slowloris lagi untuk mengevaluasi mitigasi — catat apakah halaman normal kembali dan metrik membaik.
- Kesimpulan singkat: melalui skenario ini peserta memahami mekanisme serangan low-and-slow (Slowloris), dampaknya terhadap ketersediaan Nginx, serta efektivitas mitigasi konfigurasi server di lingkungan lab terisolasi.

BAB IV

TOOLS

Tools

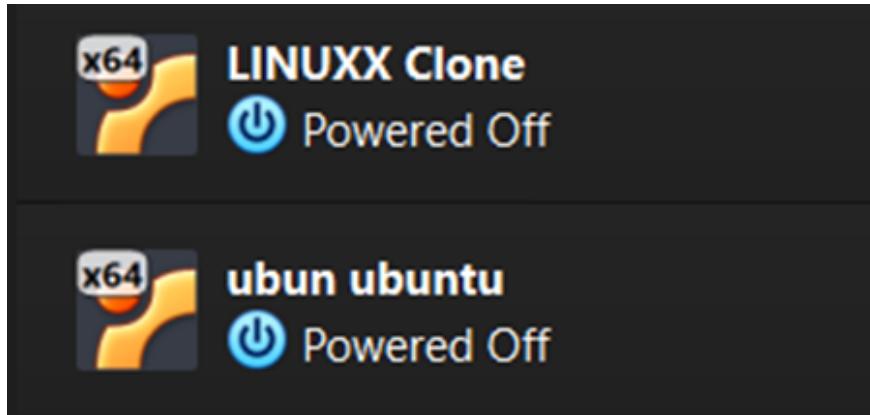
- VM (Oracle Virtual Box)
- Ubuntu 22.04.5
- Kali-linux 2025.3
- Slowloris (github)
- Nginx

Prasyarat Simulasi

- VM berjalan di VirtualBox/VMware dengan jaringan NAT & Bridge Adapter
- Hak sudo/root pada kedua VM
- Semua pengujian hanya di lingkungan lab terisolasi (jangan di jaringan publik)

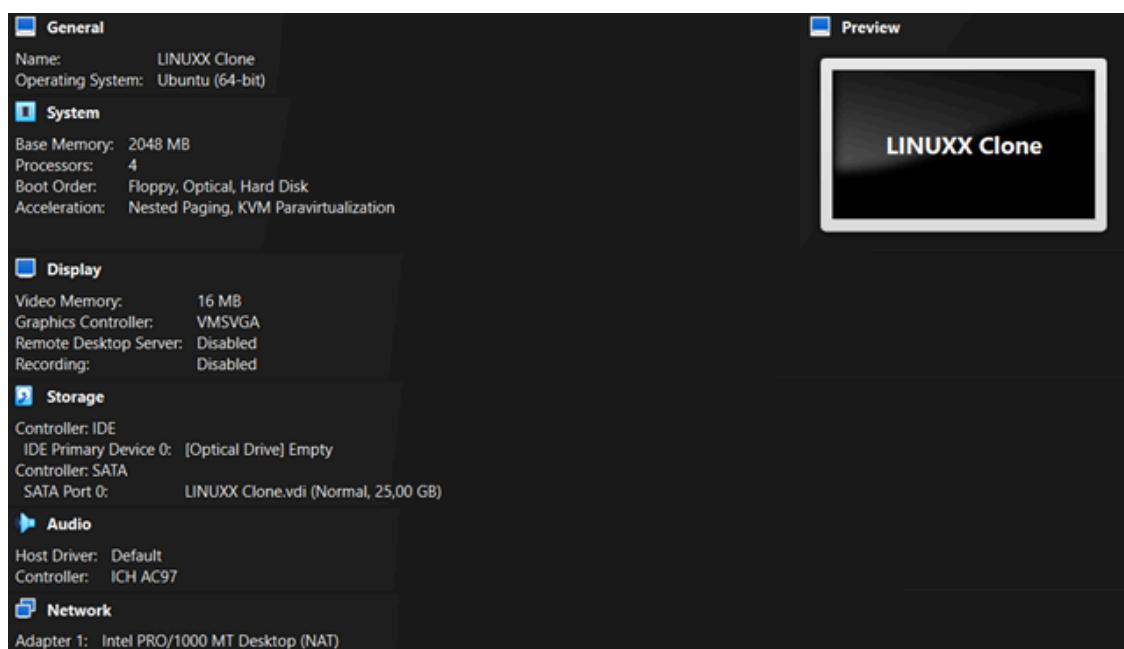
BAB V

STEP BY STEP SIMULASI



Buat dua VM di Oracle VirtualBox: ATTACK (distro Linux untuk penyerang) dan victim (Ubuntu).

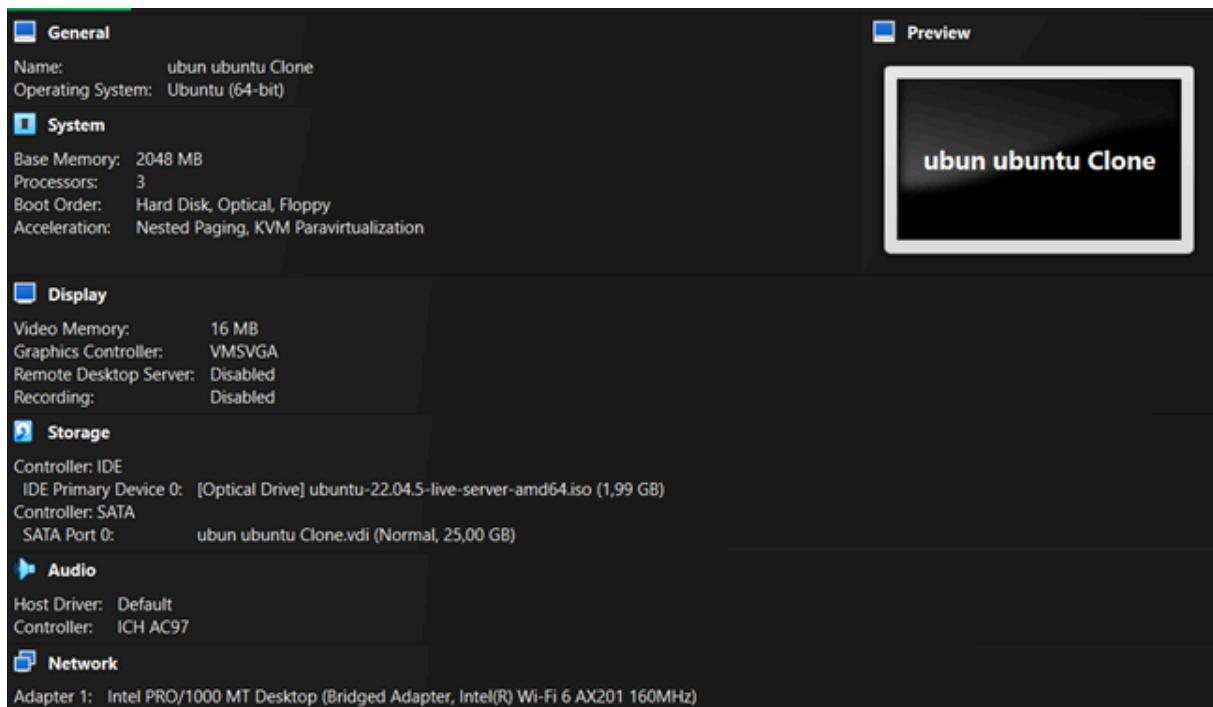
- Klik New → buat VM ATTACK (Tipe: Linux, pilih distro), alokasikan RAM & disk, pasang ISO, install OS.
- Ulangi untuk VM victim (Tipe: Linux → Ubuntu), install OS.



LINUXX Clone — VM Ubuntu (64-bit) yang berperan sebagai ATTACK dikonfigurasi dengan 2 GB RAM, 4 vCPU, dan disk virtual 25 GB; virtualisasi dioptimalkan (Nested Paging, KVM paravirtualization) dan jaringan menggunakan Intel PRO/1000 (NAT); sebelum simulasi pastikan akses sudo/root, ambil snapshot VM, verifikasi koneksi ke Victim (ping / ip addr), dan siapkan tool dengan git clone <https://github.com/gkbrk/slowloris> lalu cd slowloris.

BAB V

STEP BY STEP SIMULASI



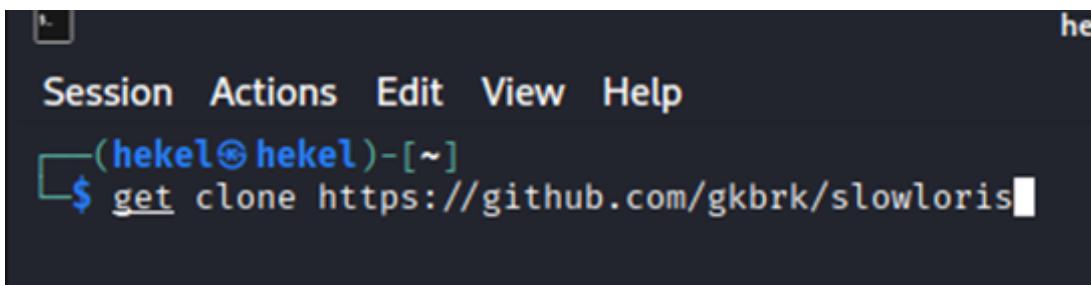
VM bernama ubun ubuntu Clone dikonfigurasi sebagai Ubuntu (64-bit) dengan spesifikasi utama dan pengaturan sebagai berikut:

- Memori (RAM): 2048 MB (2 GB)
- Prosesor: 3 vCPU
- Urutan boot: Hard Disk → Optical → Floppy (boot utama dari disk virtual)
- Akselerasi: Nested Paging, KVM Paravirtualization
- Display: Video Memory 16 MB, Graphics Controller: VMSVGA
- Storage:
 - Optical Drive: ubuntu-22.04.5-live-server-amd64.iso (mounted, 1.99 GB) — digunakan untuk instalasi.
 - Virtual disk (SATA): ubun ubuntu Clone.vdi, 25 GB (Normal).
- Audio: Host Driver: Default, Controller: ICH AC97
- Network: Adapter 1 = Intel PRO/1000 MT Desktop (Bridged Adapter — terhubung ke jaringan fisik host melalui Wi-Fi)

Langkah selanjutnya, jalankan VM Linux dan ubuntu yang telah disetup.

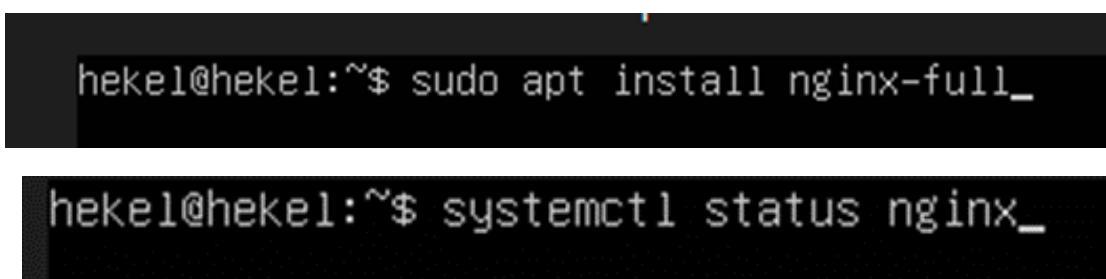
BAB V

STEP BY STEP SIMULASI

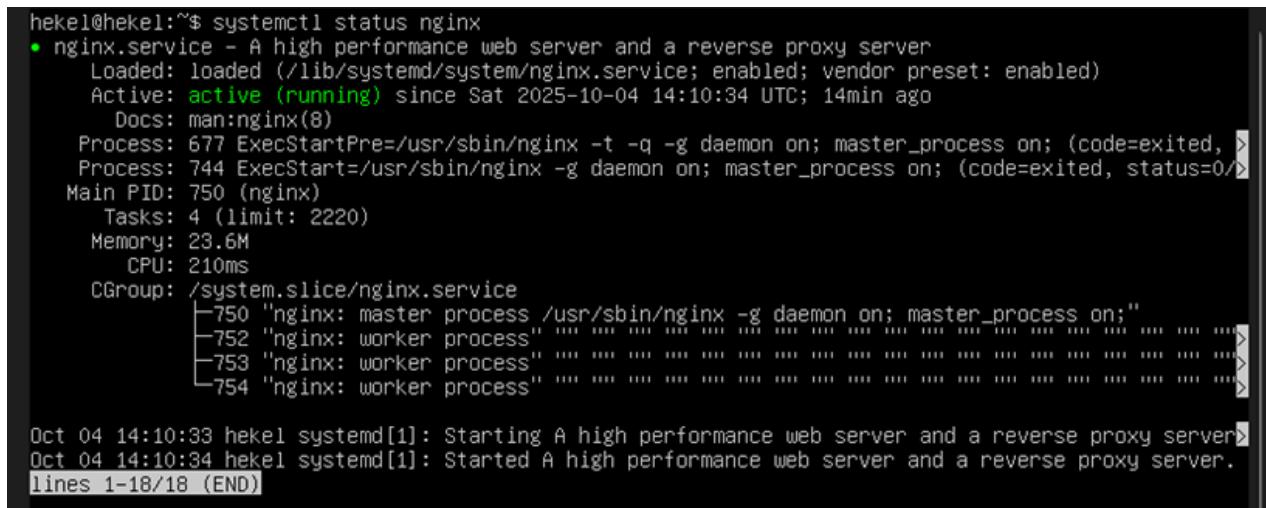


```
hekel@hekel:~$ git clone https://github.com/gkbrk/slowloris
```

Setelah berhasil login pada VM ATTACK, buka terminal dan jalankan perintah git clone https://github.com/gkbrk/slowloris untuk mengunduh skrip Slowloris dari GitHub; alat ini akan digunakan untuk mensimulasikan serangan lapisan aplikasi terhadap server web Nginx.



```
hekel@hekel:~$ sudo apt install nginx-full
hekel@hekel:~$ systemctl status nginx
```



```
hekel@hekel:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
  Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2025-10-04 14:10:34 UTC; 14min ago
    Docs: man:nginx(8)
   Process: 677 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, -->
   Process: 744 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/0)
 Main PID: 750 (nginx)
    Tasks: 4 (limit: 2220)
   Memory: 23.6M
      CPU: 210ms
     CGroup: /system.slice/nginx.service
             ├─750 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─752 "nginx: worker process"
             ├─753 "nginx: worker process"
             ├─754 "nginx: worker process"
```

```
Oct 04 14:10:33 hekel systemd[1]: Starting A high performance web server and a reverse proxy server
Oct 04 14:10:34 hekel systemd[1]: Started A high performance web server and a reverse proxy server.
lines 1-18/18 (END)
```

Pada VM Victim, instal Nginx dengan perintah sudo apt install nginx-full lalu verifikasi status layanan menggunakan systemctl status nginx. Langkah ini menyiapkan server web yang akan diamati selama skenario serangan dan mitigasi.

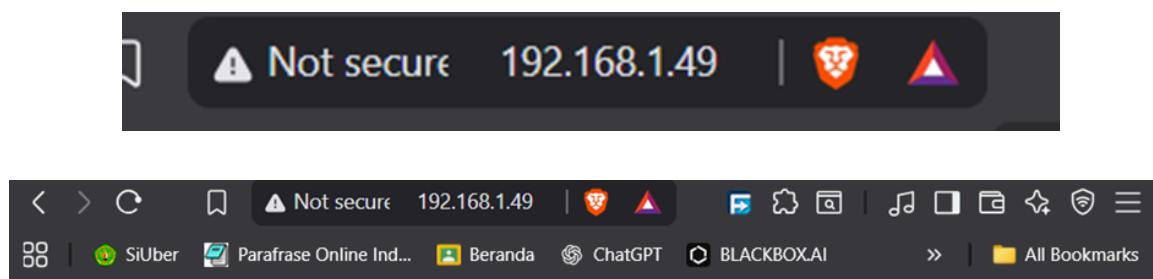
BAB V

STEP BY STEP SIMULASI

```
hekel@hekel:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.49 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe45:5eb prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:c4:55:eb txqueuelen 1000 (Ethernet)
            RX packets 83000 bytes 115886804 (115.8 MB)
            RX errors 5409 dropped 0 overruns 0 frame 5409
            TX packets 54876 bytes 4089739 (4.0 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 156 bytes 16715 (16.7 KB)
```

Gunakan ipconfig pada VM Victim untuk mengetahui alamat IP-nya. Catat IP tersebut karena akan digunakan untuk mengakses layanan web pada tahap verifikasi dan sebagai parameter target saat mengeksekusi skrip serangan.



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

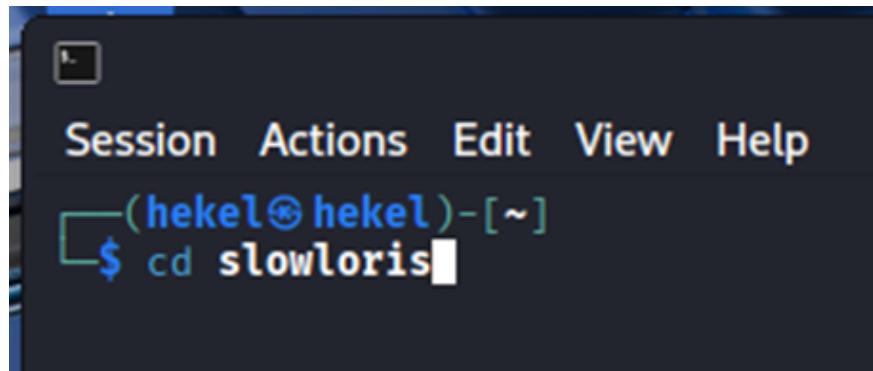
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Masukkan alamat IP Victim yang telah diperoleh ke peramban web pada mesin host atau client untuk membuka halaman default Nginx; halaman ini (mis. "Welcome to nginx!") menjadi indikator bahwa layanan web telah berhasil dijalankan dan siap untuk diuji.

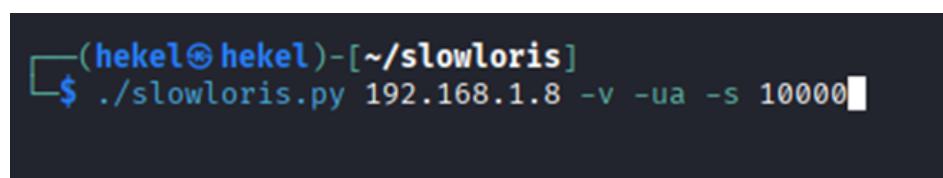
BAB V

STEP BY STEP SIMULASI



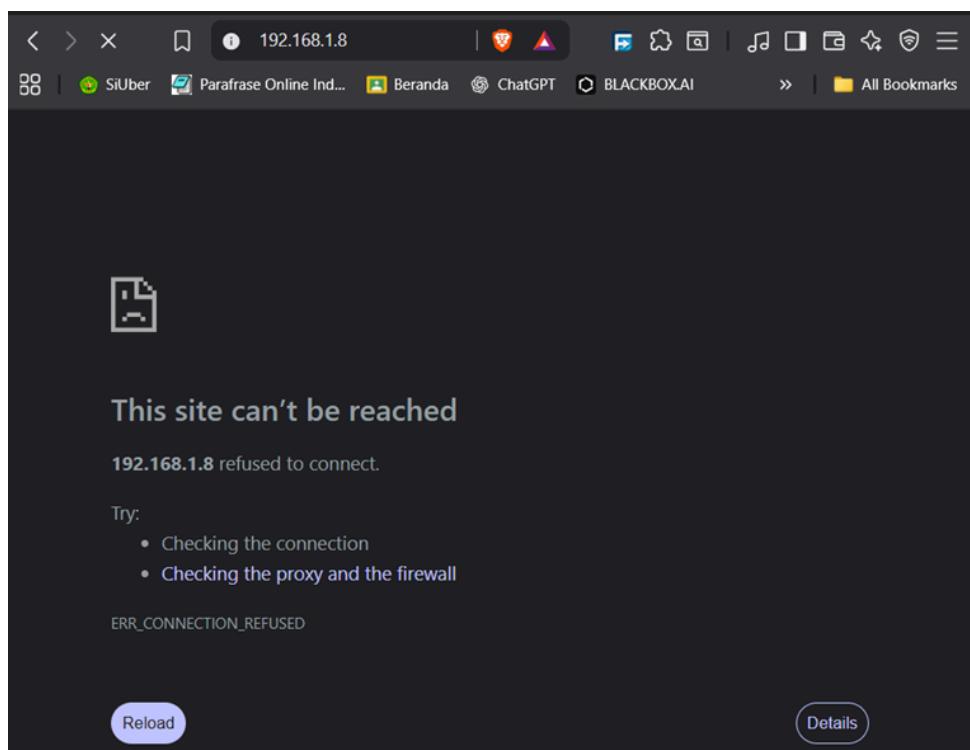
```
Session Actions Edit View Help
└─(hekel㉿hekel)-[~]
$ cd slowloris
```

Kembali ke VM ATTACK, masuk ke direktori hasil clone dengan perintah cd slowloris untuk mempersiapkan eksekusi skrip. Dalam direktori ini tersedia skrip dan opsi yang digunakan untuk melancarkan serangan simulasi.



```
└─(hekel㉿hekel)-[~/slowloris]
$ ./slowloris.py 192.168.1.8 -v -ua -s 10000
```

Jalankan skrip serangan sesuai contoh; gunakan IP Victim sebagai target dan parameter numerik yang diperlukan (misal 10000 pada dokumentasi) sebagai argumen skrip. Sesuaikan parameter ini dengan kondisi lab dan pastikan serangan dijalankan hanya terhadap VM yang berizin.



BAB V

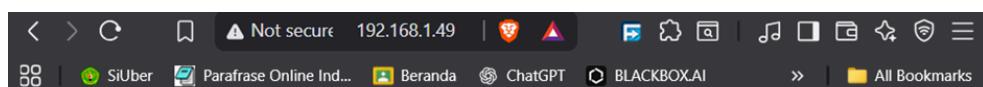
STEP BY STEP SIMULASI

Jika eksekusi serangan berhasil, halaman Nginx pada Victim akan menjadi tidak responsif atau tidak dapat diakses oleh pengguna, menggambarkan efek denial-of-service yang diharapkan dari skenario Slowloris. Catat kondisi ini untuk analisis pasca-ujicoba.

```
hekel@hekel:~$ sudo nano nginx.d_
GNU nano 6.2                                     nginx
limit_conn_zone $binary_remote_addr zone=addr;10m;

server {
    # ...
    location /store/ {
        limit_conn addr 10;
        # ...
    }
    client_body_timeout 5s;
    client_headers_timeout 5s;
    # ...
}
```

Untuk mitigasi, pada VM Victim buka editor konfigurasi (dokumen mencontohkan sudo nano nginx.d) dan masukkan konfigurasi mitigasi yang sesuai untuk mengembalikan ketersediaan layanan. Simpan perubahan tersebut sesuai panduan dokumen.



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Terakhir, setelah menerapkan konfigurasi mitigasi yang dicatat dalam dokumen, verifikasi bahwa halaman Nginx kembali pulih dan dapat diakses oleh pengguna.

BAB VI

PENUTUP

Sebagai rangkuman, modul ini telah menjelaskan tujuan, skenario, dan prosedur praktikum yang berfokus pada simulasi serangan DDoS termasuk kasus low-and-slow (Slowloris) dan skenario protokol dalam lingkungan laboratorium yang terisolasi. Eksperimen dirancang untuk memberi pengalaman praktis: menyiapkan topologi VM, menjalankan serangan terkontrol, mengumpulkan metrik performa, menerapkan mitigasi pada Nginx, serta mengevaluasi efektifitas tindakan pemulihan.

Dari hasil pembelajaran di modul ini, peserta diharapkan mampu: (1) membedakan kelas serangan DDoS (volumetric, protocol, application-layer), (2) mengoperasikan dan memantau lingkungan uji terisolasi, (3) mendeteksi indikator serangan pada level sistem dan aplikasi, serta (4) merancang dan menguji konfigurasi mitigasi dasar pada server web (mis. timeout, limit_conn/limit_req). Pemahaman tersebut penting untuk merancang kebijakan operasional yang meningkatkan ketersediaan layanan pada lingkungan produksi.

Catatan penting—etika dan keamanan eksperimen—tetap menjadi prioritas: seluruh pengujian wajib dilakukan hanya pada infrastruktur milik sendiri atau yang telah mendapat izin eksplisit, dengan isolasi jaringan yang memadai (Host-Only / Internal), snapshot cadangan sebelum pengujian, dan pencatatan log untuk audit. Pelanggaran terhadap ketentuan ini dapat menimbulkan konsekuensi hukum dan etis.