Dasar Linux “NgeHek”

Aditya “ghxyss” IT (Ilmu Tukang)

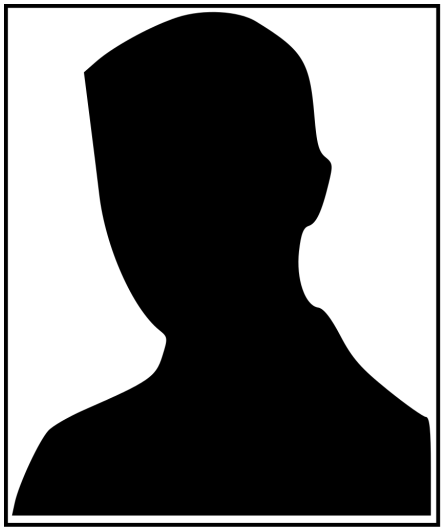
whoami!

Aditya

IT Consultant & Trainer

Security Engineer with 4+ years Experience

https://www.linkedin.com/in/adityyaaa/



Outline

- Paket Manager (Kang Paket)
- Arsitektur Linux (Kang Mandor)
- Kernel vs User Space (Tempat Bermain)
- Linux Hardening (Default ? no no no !!)
- Administrator & Service (Kang Bebersih)
- Shell & Scripting (Kang Santai)

Dasar Linux “Ehe”

- Paket Manager

apt install [name-paket]

Distro	Manager	Contoh Perintah
Ubuntu/Debian	`apt`, `dpkg`	`apt install nmap`
RHEL/CentOS	`dnf`, `yum`	`dnf install nmap`
Arch Linux	`pacman`	`pacman -S nmap`
Alpine Linux	`apk`	`apk add nmap`

Dasar Linux “Ehe”

- Paket Manager

cat /etc/apt/source.list

```
# newer versions of the distribution.
deb http://id.archive.ubuntu.com/ubuntu/ jammy main restricted
# deb-src http://id.archive.ubuntu.com/ubuntu/ jammy main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://id.archive.ubuntu.com/ubuntu/ jammy-updates main restricted
# deb-src http://id.archive.ubuntu.com/ubuntu/ jammy-updates main restricted
```

Dasar Linux “Ehe”

- Paket Manager

Komponen	Keterangan
`main`	Resmi, didukung penuh oleh Ubuntu, open-source
`restricted`	Driver proprietary (misalnya NVIDIA), masih didukung Ubuntu
`universe`	Komunitas-maintained, **tidak** didukung resmi oleh Ubuntu
`multiverse`	Software non-free (proprietary), **tidak ada review keamanan resmi**
`backports`	Versi baru dari software, **belum stabil**, tidak dijamin aman
`security`	Pembaruan keamanan resmi Ubuntu

Dasar Linux “Ehe”

- Cara testing repository bagus gimana ya ? biar **paket** nya secure !!

Key Takeaway !

- JANGGAN INSTALL SEMBARANG HEY !!.
- Paket itu harus di update, kalo dibiarkan default sama aja.
- Supply Chain Attack (Komponen barang di paket A, Aplikasinya ada di B, Dihack di Komponen Paket A, Aplikasinya Lemah !)
- Sumpah masih mau pake aplikasi bajakan ? ehe

Arsitektur Linux

- Arsitektur Linux

User Applications ← Contoh: Firefox, nano, Python

GNU Tools & Shell ← Contoh: bash, ls, grep

System Libraries ← Contoh: glibc, libcrypto

Kernel ← Jantung Linux: driver, scheduler

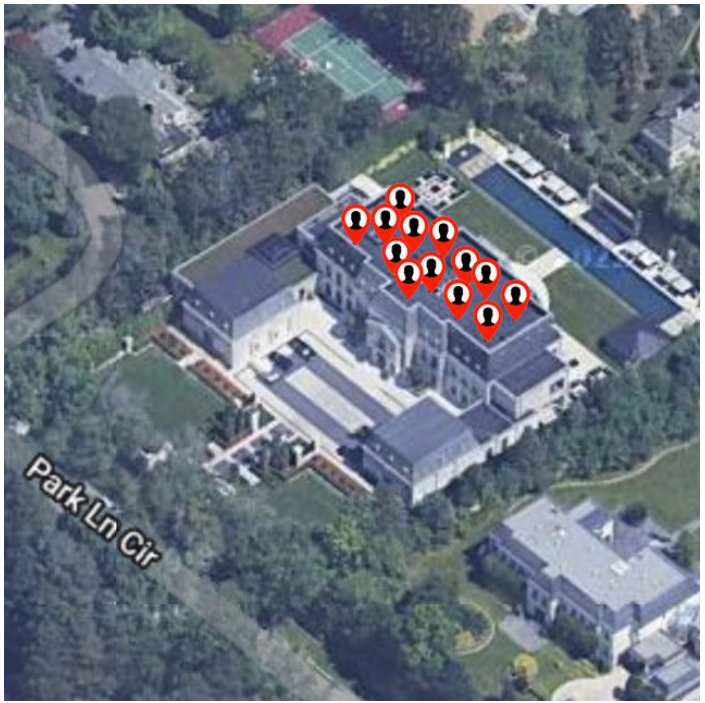
Hardware ← CPU, RAM, Disk, NIC, dll

Arsitektur Linux

- Rumah Idaman

Rumah Tipe 163

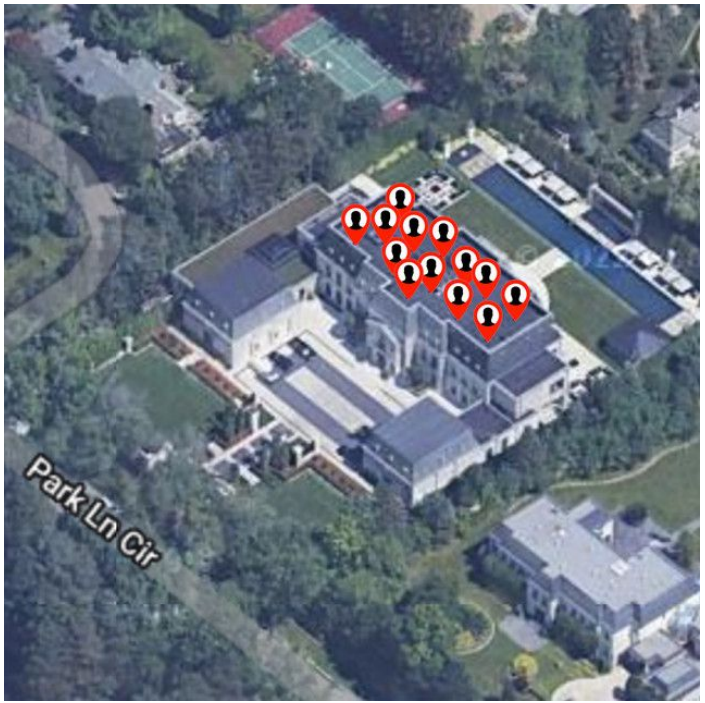
- Kamar 3
- Ruang Tamu 2
- Dapur bersih 1
- Dapur kotor 1
- Tempat Bermain
- Luas Bagunan 159 Meter persegi



Arsitektur Linux

- Arsitektur Linux

kalo misalnya kita pengen makan mie di dapur kotor yang ada di belakang rumah, tapi kamar kita ada di lantai 2 dan untuk makan mie saja harus lewatin semua ruangan, mager ga ?



Arsitektur Linux

- Arsitektur Linux

Area Rumah	Representasi Sistem Linux
Ruang Tamu & Kamar	**User Space** – Tempat penghuni tinggal & beraktivitas
Dapur & Listrik	**Kernel Space** – Mesin utama rumah, nggak boleh sembarang diakses
Telepon Rumah	**System Call** – Satu-satunya cara komunikasi resmi ke dapur
Anak Kos	Aplikasi biasa (user)
Pemilik Rumah	Root user / kernel

Arsitektur Linux

- Arsitektur Linux

Anak kos (user space) **tidak boleh masuk dapur langsung**, karena:

- Bisa bikin **kompor meledak**
- Bisa **ngacak-ngacak kulkas**
- Bisa **nggak ngerti cara nyalain MCB** dan bikin short

Kalau anak kos mau masak mie:

- Dia **telepon pemilik rumah (syscall)**:
  "Tolong nyalain kompor, saya mau rebus mie !"

Pemilik rumah (kernel) akan:

- Cek permintaan
- Kalau aman → laksanakan
- Kalau nggak jelas → "Kamu siapa? Pergi sana!"

Key Takeaways !

- Masak Mie, yang bener !
- Makan Mie tuh pelan-pelan yak...
- Syscal berfungsi sebagai penghubung antar **user** dan **kernel** space
- Kernel Exploit -> **Privilege Escalation (Overflow)**
- Pwn !!! (Love uu)

Kernel vs user space

- Kernel
    - Fungsi : Hardisk, Manajemen Proses, Memory (HUHUHUU)
- User
    - Fungsi : Aplikasi, Interaksi User !

Cara mereka bekerja sama kek mana ya ?

Mak Comblang Syscall

- System Call

Aspek	Kernel Space	User Space
Fungsi	Kontrol CPU, memory, I/O	Jalankan aplikasi/shell
Akses	Privileged (ring 0)	Limited (ring 3)
Interaksi	Melalui syscall	Melalui tools/shell
Risiko Crash	Bisa rusak sistem	Hanya aplikasi

Mak Comblang Syscall

- Contoh Interaksi Syscall

Kita Lakukan Command Dibawah :

```
cat /etc/passwd
```

1. `cat` (berada di user space) butuh baca file.
2. `cat` memanggil **system call** `open()`, `read()`.
3. Kernel menerima system call itu, lalu baca data dari disk.
4. Kernel kirimkan data hasil baca ke aplikasi `cat`.
5. `cat` menampilkan ke layar.

Mak Comblang Syscall

- Output Syscall

Link Docs :
https://docs.google.com/document/d/1S8vfivJ1X3TmSBqYMjmWXYF-9cxj6sZ7JmePyEthyY8/edit?usp=sharing

Key Takeaways !

- Mau Belajar Pwn ? ya iki lo ndoo !
- Malware Analyst ? ya ini brok !!
- Reverse Eng ? ini cok !!
- Blue Team Jaya !!!

Linux Hardening !

- Default ? No no no !!
- Port 22 ? no no no -> Ubah jadi port jahanam
- Hardening ? nyaman ? BIG NOOOOOOOOO

Linux Hardening !

- Hardening intinya nguatin.
- CIS & STIG Beanchmarking

Key Takeaways !

- Key Takeaways nya nguatin service itu jangan dibiarkan default !
- dah itu aja cukup.

Administrator Service

User & Grup:

- adduser, passwd, usermod, groups
- Permission: chmod, chown, umask

Disk & Partisi:

- lsblk, df -h, mount, umount

Network:

- ip a, ss -tuln, ping, traceroute, dig

Backup:

- tar, rsync, scp

Monitoring:

- top, htop, ps aux, free -h

Key Takeaways !

- Males ah ngasih tau nya..
- Bikin sendiri
  - https://forms.gle/wiqfhKPNPY2QDQAm7

Linux Shell & Scripting

Jenis Shell:

- `bash`, `zsh`, `dash`, `sh`

Command Penting:

- `echo`, `read`, `if`, `for`, `while`
- `#!/bin/bash` = Shebang line

Script Sederhana:

```
#!/bin/bash
for i in {1..5}; do
  echo "User $i"
done
```

Resource

- Done Kawan !

- Linux 101 https://training.linuxfoundation.org/training/introduction-to-linux/
- Linux Syscall (REDACTED)
- Linux Administrasi (LPIC Resource -> Cari Sendiri)