Day 28

Exploitation Analyst

GRUB protection and Security:

What is GRUB in Linux?

GRUB stands for GRand Unified Bootloader. It's the default bootloader for most Linux distributions.

What Does It Do?

GRUB is responsible for:

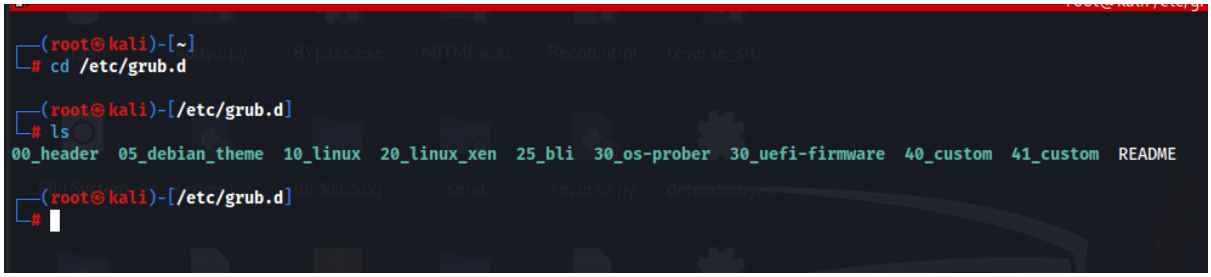
1. Loading the Linux kernel (or other OS kernels) into memory.
2. Presenting a boot menu if there are multiple OSes or kernels.
3. Passing boot-time arguments to the kernel (like single-user mode).
4. Supporting recovery options, initrd, etc.

Why Should GRUB Be Kept Secure?

Because if GRUB is compromised, an attacker can:

1. Bypass system security by booting into:
    a. Single-user mode (root access without password)
    b. A custom/initramfs shell
2. Modify kernel parameters (e.g., init=/bin/bash or disabling SELinux/AppArmor)
3. Boot their own malicious OS from USB/ISO
4. Install a rootkit via kernel-level tampering

Where to find this GRUB?



How to protect the GRUB?

GRUB is not insecure by mistake, it's insecure by assumption — the assumption that you are in control of physical access and security settings. That's why, in cybersecurity, you must explicitly secure it.

So What Should a Cybersecurity Professional Do?

1. Set a GRUB password → to prevent editing boot options.
2. Use Full Disk Encryption → GRUB can't access real content without a key.
3. Disable USB boot + Set BIOS password → avoid physical bypass.
4. Use Secure Boot (if supported) → bootloader & kernel verification.

5. Monitor /boot and GRUB files → integrity check (Tripwire, AIDE).

Protecting GRUB:

Set a GRUB password:

What Is Vulnerable by Default?

By default, anyone with physical access can:

- Press e at the GRUB menu → edit boot parameters.
- Add:

init=/bin/bash

gain **root access** without a password.

This means they can:

- Reset root passwords.
- Disable security modules (e.g., selinux=0).
- Access encrypted volumes (if not fully protected).

Steps to protect:

Generate a Secure GRUB Password Hash: use the command “grub-mkpasswd-pbkdf2”

Edit GRUB’s Custom Script File: use sudo nano /etc/grub.d/40_custom, following screen will appear:

Add these lines there:

Then update the GRUB: use the command “update-grub”

```
┌──(root㉿kali)-[/etc/grub.d]
└─# update-grub

Generating grub configuration file ...
Found theme: /boot/grub/themes/kali/theme.txt
Found background image: /usr/share/images/desktop-base/desktop-grub.png
Found linux image: /boot/vmlinuz-6.12.33+kali-amd64
Found initrd image: /boot/initrd.img-6.12.33+kali-amd64
Found linux image: /boot/vmlinuz-6.12.13-amd64
Found initrd image: /boot/initrd.img-6.12.13-amd64
Found linux image: /boot/vmlinuz-6.11.2-amd64
Found initrd image: /boot/initrd.img-6.11.2-amd64
Found linux image: /boot/vmlinuz-6.10.11-amd64
Found initrd image: /boot/initrd.img-6.10.11-amd64
Found linux image: /boot/vmlinuz-6.10.9-amd64
Found initrd image: /boot/initrd.img-6.10.9-amd64
Found linux image: /boot/vmlinuz-6.8.11-amd64
Found initrd image: /boot/initrd.img-6.8.11-amd64
Found linux image: /boot/vmlinuz-6.6.15-amd64
Found initrd image: /boot/initrd.img-6.6.15-amd64
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done

┌──(root㉿kali)-[/etc/grub.d]
└─#
```

--The End--