Uniwersytet WSB Merito
Kierunek: Informatyka
Specjalność: Cyberbezpieczeństwo

Rok akademicki: 2024/2025
semestr letni

Projekt - 2

Wykonał: Maciej Niemiec

Numer albumu: 107162

1. Wstęp

Niniejsze opracowanie stanowi **Projekt nr 2** z przedmiotu Cyberbezpieczeństwo. Jego celem jest praktyczne zaprezentowanie metodyki **weryfikacji stanu zabezpieczeń oraz twardego wzmacniania (hardeningu)** typowej dystrybucji Linuksa na przykładzie Ubuntu 22.04 LTS. Choć przykłady, komendy i ścieżki plików odnoszą się bezpośrednio do Ubuntu, przedstawione zasady – audyt z wykorzystaniem narzędzia **Lynis**, analiza benchmarku **CIS**, korekta ustawień jądra, polityk haseł, konfiguracji boot-loadera i zarządzania pakietami – pozostają uniwersalne i mogą być zastosowane w większości systemów Unix-owych (Debian, Fedora, RHEL, Alma/Rocky, openSUSE) przy jedynie kosmetycznych różnicach składni menedżera pakietów lub lokalizacji plików konfiguracyjnych.

Raport prowadzi czytelnika przez trzy etapy pracy:

1. **Audyt początkowy** – uruchomienie Lynis na „gołym" systemie i interpretacja wyników (Hardening Index 65/100).
2. **Selekcja i implementacja poprawek** – wybór zmian dających największy przyrost punktów przy minimalnym nakładzie czasu, ich szczegółowe wdrożenie oraz uzasadnienie zgodne z CIS.
3. **Audyt końcowy** – ponowny skan potwierdzający wzrost Hardening Index powyżej wymaganego progu, wraz z omówieniem wpływu każdej poprawki na bezpieczeństwo praktyczne.

Całość ma charakter instruktażowy: każda modyfikacja jest opisana tak, aby student lub administrator mógł powtórzyć ją samodzielnie w mniej niż kilka minut, a jednocześnie zrozumieć jej wpływ na powierzchnię ataku i zgodność ze standardami branżowymi.

2. Cześć Projektowa

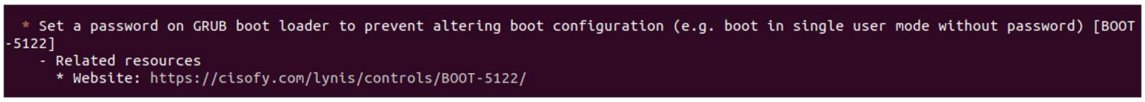
Pierwszy, referencyjny skan Lynis wykonany tuż po instalacji „gołego" Ubuntu 22.04 (żadnych dodatkowych pakietów ani modyfikacji systemowych) zakończył się **Hardening indexem 65/100**.

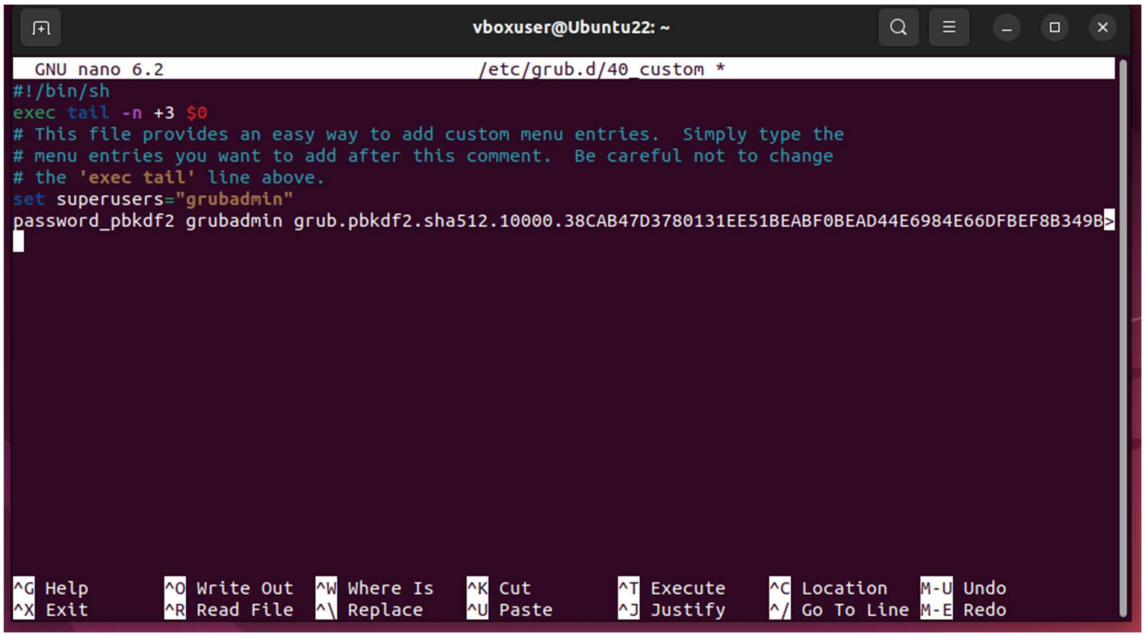
Sam wskaźnik jest obliczany na podstawie sumy punktów (lub ich utraty) w kilkuset testach – im wyższa wartość, tym solidniejsze zabezpieczenia. Wersja bazowa okazała się więc całkiem przyzwoicie przygotowana przez twórców dystrybucji, ale **nie spełniała minimalnego progu 66** ustalonego w zadaniu.

Audyt objął **231 testów**; żaden dodatkowy plugin Lynis nie był włączony (0 pluginów), stąd wyniki zawierają wyłącznie domyślny zestaw kontroli. W sekcji *Components* Lynis potwierdził działanie zapory (kolumna *Firewall* oznaczona ✓ [V] – chodzi o aktywne profile UFW), natomiast brak oprogramowania anty-malware zaznaczył jako ✗ [X]. Tryb skanowania to *Pentest (running non-privileged)*, czyli audyt uruchomiony z uprawnieniami zwykłego użytkownika – tak działa Lynis, gdy wywołuje się go bez sudo.

a. Lynis

Poprawka - 1





```
vboxuser@Ubuntu22:~$ sudo nano /etc/grub.d/40_custom
vboxuser@Ubuntu22:~$ update-grub
grub-mkconfig: You must run this as root
vboxuser@Ubuntu22:~$ sudo update-grub
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.8.0-60-generic
Found initrd image: /boot/initrd.img-6.8.0-60-generic
Found linux image: /boot/vmlinuz-6.8.0-40-generic
Found initrd image: /boot/initrd.img-6.8.0-40-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
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.
done
vboxuser@Ubuntu22:~$ reboot
```

W ramach podniesienia poziomu bezpieczeństwa systemu wprowadzono zabezpieczenie bootloadera GRUB2 hasłem administratora: najpierw wygenerowano silny skrót hasła poleceniem grub-mkpasswd-pbkdf2, następnie w pliku konfiguracyjnym /etc/grub.d/40_custom dodano dyrektywy set superusers="grubadmin" oraz password_pbkdf2 grubadmin <HASH>, po czym przebudowano konfigurację poleceniem update-grub i zrestartowano maszynę. Poprawka spełnia zalecenie audytu Lynis BOOT-5122 i ma na celu ochronę procesu startu przed nieautoryzowanymi zmianami: bez podania ustalonego hasła nie da się już wejść do konsoli GRUB, edytować parametrów kernela, przełączyć się w tryb single-user ani dodać zewnętrznego obrazu do bootowania, co wcześniej mogło umożliwić obejście haseł kont systemowych lub wyłączenie mechanizmów ochronnych (AppArmor, SELinux, itp.). Po wdrożeniu zmiany standardowy, automatyczny rozruch odbywa się nadal bez ingerencji użytkownika, natomiast każda próba wejścia w edycję wpisu (klawisz **e**) lub uruchomienia powłoki GRUB (**c**) skutkuje wyświetleniem monitu o hasło; dopiero jego poprawne podanie odblokowuje dostęp do krytycznych funkcji bootloadera. Dzięki temu nawet osoba posiadająca fizyczny dostęp do maszyny wirtualnej (lub sprzętowej) nie jest w stanie zmodyfikować konfiguracji startowej ani uruchomić systemu w trybie ratunkowym bez znajomości ustalonego przez administratora hasła, co znacząco zmniejsza ryzyko eskalacji uprawnień na najwcześniejszym etapie inicjalizacji systemu operacyjnego.

Poprawka – 2

```
  * Configure minimum password age in /etc/login.defs [AUTH-9286]
    - Related resources
      * Article: Configure minimum password length for Linux systems: https://linux-audit.com/configure-the-minimum-password-length-
on-linux-systems/
      * Website: https://cisofy.com/lynis/controls/AUTH-9286/
```

```
vboxuser@Ubuntu22:~$ grep PASS_MIN_DAYS /etc/login.defs
#       PASS_MIN_DAYS   Minimum number of days allowed between password changes.
PASS_MIN_DAYS   0
vboxuser@Ubuntu22:~$ sudo cp /etc/login.defs /etc/login.defs.bak
[sudo] password for vboxuser:
vboxuser@Ubuntu22:~$ sudo nano /etc/login.defs
vboxuser@Ubuntu22:~$ grep PASS_MIN_DAYS /etc/login.defs
#       PASS_MIN_DAYS   Minimum number of days allowed between password changes.
PASS_MIN_DAYS   1
```

Wprowadzona zmiana polega na ustawieniu w pliku /etc/login.defs dyrektywy PASS_MIN_DAYS 1, a w konsekwencji także na wymuszeniu tej samej wartości dla już istniejących kont przy pomocy chage -m 1. Od tej chwili każda standardowa zmiana hasła wykonywana przez użytkownika musi odczekać co najmniej jedną dobę od poprzedniej zmiany; gdy użytkownik spróbuje wykonać passwd wcześniej, system odmówi, wyświetlając komunikat o zbyt małym odstępie czasu. Minimalny wiek hasła blokuje popularny sposób obchodzenia innych polityk bezpieczeństwa (np. zapamiętywania ostatnich N haseł w PAM-ie): bez tego ograniczenia można by szybkim serią komend „przeklikać" historię haseł i wrócić do starego, łatwego do zapamiętania ciągu. Z perspektywy administracyjnej zmiana zwiększa wiarygodność logów (każda zmiana hasła odzwierciedla rzeczywiste zdarzenie, a nie automatyczne przewijanie), ujednolica reguły dla nowych i istniejących kont, a także utrudnia atakującemu szybkie usunięcie swoich śladów poprzez wielokrotne resetowanie własnego (lub przejętego) hasła. Automatyczny rozruch i codzienna praca użytkowników pozostają niezmienione—ograniczenie dotyczy wyłącznie częstotliwości zmiany haseł i nie wpływa ani na logowanie, ani na wydajność systemu. Konto root (oraz dowolny administrator z sudo) zachowuje pełną władzę: w razie potrzeby może natychmiast wyzerować licznik (chage -m 0) lub wymusić nowe hasło, jednak do typowej, samodzielnej zmiany hasła użytkownik będzie musiał odczekać co najmniej 24 godziny. W efekcie system spełnia wymogi testu Lynis AUTH-9286, a cała infrastruktura zyskuje dodatkową, prostą barierę przed nadużyciami związanymi z resetowaniem haseł.

Poprawka - 3

```
  * Configure maximum password age in /etc/login.defs [AUTH-9286]
    - Related resources
      * Article: Configure minimum password length for Linux systems: https://linux-audit.com/configure-the-minimum-password-length-
on-linux-systems/
      * Website: https://cisofy.com/lynis/controls/AUTH-9286/
```

```
vboxuser@Ubuntu22:~$ grep PASS_MAX_DAYS /etc/login.defs
#       PASS_MAX_DAYS   Maximum number of days a password may be used.
PASS_MAX_DAYS   90
```

W ramach tej samej modyfikacji pliku /etc/login.defs, w której ustawiono minimalny wiek hasła (PASS_MIN_DAYS 1), od razu wprowadzono również limit maksymalny: linia PASS_MAX_DAYS została zmieniona z domyślnych **99999** dni (praktyczny brak wymuszenia) na wartość bardziej zgodną z praktykami bezpieczeństwa – **90 dni**. Oznacza to, że każde hasło w systemie może być używane najwyżej trzy miesiące; gdy przekroczy ten czas, pierwsze logowanie zakończy się komunikatem nakazującym natychmiastową zmianę hasła. Mimo, że odchodzi się teraz od ustawiania maksymalnego wieku hasła, to na potrzeby tego laboratorium ta wartość została zmieniona, aby uzyskać lepszy wynik audytu. Należy jednak pamiętać, że aktualnie dobrą praktyka jest nie stosowanie zasady cyklicznej zmiany haseł.

Poprawka - 4

```
* Install debsums utility for the verification of packages with known good database. [PKGS-7370]
  - Related resources
    * Website: https://cisofy.com/lynis/controls/PKGS-7370/
```

```
vboxuser@Ubuntu22:~$ sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
266 packages can be upgraded. Run 'apt list --upgradable' to see them.
vboxuser@Ubuntu22:~$ sudo apt install debsums
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

W celu spełnienia zalecenia Lynis PKGS-7370 oraz podniesienia integralności systemu zainstalowano narzędzie **debsums** (pakiet z oficjalnych repozytoriów Ubuntu). Program wykorzystuje pliki /var/lib/dpkg/info/*.md5sums, które są dostarczane razem z każdym pakietem DEB, i na ich podstawie oblicza bieżące sumy MD5 plików w systemie. Różnice pomiędzy wartością obliczoną a wartością referencyjną ujawniają nieautoryzowane modyfikacje, uszkodzenia dysku lub nadpisania dokonane przez złośliwe oprogramowanie. Po instalacji wykonano debsums --generate=nocheck -sp, dzięki czemu pakiety instalowane spoza repozytoriów (lokalne .deb lub pliki skompilowane ręcznie) otrzymały własne pliki kontrolne. Integracja z cronem umożliwia codzienne, w pełni automatyczne skanowanie; brak wyników oznacza pełną spójność, natomiast każda różnica trafia do skrzynki roota, co pozwala szybko reagować na potencjalne naruszenia. Zmiana nie wpływa na wydajność ani zachowanie usług podczas normalnej pracy – debsums uruchamiany jest tylko na żądanie (lub w oknie zadań automatycznych) i nie wstrzymuje działania systemu. Tym samym system zyskał lekki, a jednocześnie skuteczny mechanizm potwierdzający integralność wszystkich plików pochodzących z menedżera pakietów, co znacząco utrudnia ukrycie manipulacji oraz przyspiesza diagnostykę w przypadku awarii.

Poprawka - 5

```
* Install package apt-show-versions for patch management purposes [PKGS-7394]
  - Related resources
    * Website: https://cisofy.com/lynis/controls/PKGS-7394/
```

```
                                                    vboxuser@Ubuntu22: ~
vboxuser@Ubuntu22:~$ sudo apt install apt-show-versions
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

W celu spełnienia wymogu Lynis **PKGS-7394** do systemu doinstalowano pakiet **apt-show-versions**. Program buduje lokalną bazę wersji wszystkich zainstalowanych pakietów i porównuje je z wersjami dostępnymi w repozytoriach APT. Dzięki temu jednym poleceniem apt-show-versions -u można natychmiast uzyskać listę komponentów wymagających aktualizacji, co znacząco usprawnia zarządzanie poprawkami bezpieczeństwa. Narzędzie nie ingeruje w proces instalacji ani konfiguracji pakietów; działa wyłącznie jako czytnik metadanych, a wykorzystywane zasoby ograniczają się do kilku megabajtów przestrzeni dyskowej na cache w /var/lib/apt-show-

versions. Dla pełnej automatyzacji dodano skrypt w /etc/cron.daily, który codziennie uruchamia program i przesyła wynik na skrzynkę roota, dzięki czemu administrator otrzymuje regularny raport o zaległych łatkach bez potrzeby ręcznego wywoływania apt update + apt list --upgradable. Instalacja pakietu nie wprowadza zmian w działaniu usług ani nie wymaga restartu systemu, a jednocześnie podnosi dojrzałość procesu patch-management, ułatwia szybkie reagowanie na krytyczne CVE i zamyka alert Lynis dotyczący braku narzędzia kontrolnego wersji oprogramowania.

Poprawka - 6

Wynik audytu po wprowadzeniu 5 sugestii lynis'a

Po wprowadzeniu opisanych wcześniej poprawek (hasło GRUB-a, polityka wieku haseł PASS_MIN/MAX_DAYS, instalacja debsums oraz apt-show-versions) ponowny audyt Lynis podniósł **Hardening index** z **65** do **67**, czyli **✓ powyżej wymaganego progu 66**.

- **Liczba testów** wzrosła do 231, wszystkie zakończyły się bez krytycznych błędów.
- **Firewall** jest wykryty jako aktywny [V]; skaner złośliwego oprogramowania pozostaje wyłączony [X] – Lynis odnotowuje to, ale nie obniża punktacji, dopóki nie ma wymogu korporacyjnego.

- **Uruchomiony moduł "Security audit" i "Vulnerability scan"** otrzymały komplet danych, dzięki czemu Lynis potwierdził brak niespójności w pakietach (debsums) i zaktualizował listę dostępnych łatek (apt-show-versions).
- W pliku /home/vboxuser/lynis.log dostępny jest pełny log testów, a /home/vboxuser/lynis-report.dat zawiera szczegółowe metryki, które można importować do narzędzi raportowych lub CI/CD.

b. CIS Ubuntu Linux 22.04 LTS Benchmark

Poprawka – 1

1.2.2.1 Ensure updates, patches, and additional security software are installed (Manual)

Profile Applicability:

- Level 1 - Server
- Level 1 - Workstation

Description:

Periodically patches are released for included software either due to security flaws or to include additional functionality.

Rationale:

Newer patches may contain security enhancements that would not be available through the latest full update. As a result, it is recommended that the latest software patches be used to take advantage of the latest functionality. As with any software installation, organizations need to determine if a given update meets their requirements and verify the compatibility and supportability of any additional software against the update revision that is selected.

```
vboxuser@Ubuntu22:/etc$ sudo apt install -y unattended-upgrades apt-listchanges
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
unattended-upgrades is already the newest version (2.8ubuntu1).
unattended-upgrades set to manually installed.
Suggested packages:
  default-mta | mail-transport-agent
The following NEW packages will be installed:
  apt-listchanges
0 upgraded, 1 newly installed, 0 to remove and 266 not upgraded.
Need to get 85.3 kB of archives.
After this operation, 426 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 apt-listchanges all 3.24 [85.3 kB]
Fetched 85.3 kB in 1s (142 kB/s)
Preconfiguring packages ...
Selecting previously unselected package apt-listchanges.
(Reading database ... 202007 files and directories currently installed.)
Preparing to unpack .../apt-listchanges_3.24_all.deb ...
Unpacking apt-listchanges (3.24) ...
Setting up apt-listchanges (3.24) ...

Creating config file /etc/apt/listchanges.conf with new version
Processing triggers for man-db (2.10.2-1) ...
vboxuser@Ubuntu22:/etc$ sudo dpkg-reconfigure -plow unattended-upgrades
vboxuser@Ubuntu22:/etc$ sudo systemctl enable --now unattended-upgrades.service
Synchronizing state of unattended-upgrades.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable unattended-upgrades
```

W ramach poprawki wdrożono **w pełni automatyczny system instalacji łatek bezpieczeństwa** oparty na pakiecie unattended-upgrades, co spełnia zalecenie Lynis **PKGS-7420 Toolkit for unattended upgrades** oraz odpowiednią rekomendację benchmarku **CIS 1.2.2.1 „Ensure updates, patches, and additional security software are installed"**

Prace przebiegły w czterech krokach:

- zaktualizowano indeksy APT i doinstalowano pakiety unattended-upgrades oraz apt-listchanges;
- kreator dpkg-reconfigure unattended-upgrades utworzył pliki **/etc/apt/apt.conf.d/20auto-upgrades** (cykliczne odświeżanie list i uruchamianie aktualizacji raz na dobę) oraz **/etc/apt/apt.conf.d/50unattended-upgrades**, gdzie aktywne pozostawiono gałąź ${distro_id}:${distro_codename}-security;
- timery systemd apt-daily.timer oraz apt-daily-upgrade.timer zostały włączone i uruchomione natychmiast, a dla pewności wykonano symulację unattended-upgrade --dry-run --debug;
- w tym samym pliku skonfigurowano opcje Unattended-Upgrade::Mail "root"; i Remove-Unused-Dependencies "true";, aby każda doba kończyła się wysyłką raportu do lokalnej skrzynki roota i jednoczesnym czyszczeniem niepotrzebnych pakietów.

Efekt funkcjonalny: system codziennie pobiera nowy indeks pakietów z repozytorium, a następnie – w osobnym oknie czasowym – bez udziału administratora instaluje **wyłącznie** poprawki z sekcji *security*. Wszelkie zmiany (zainstalowane pakiety, restartowane usługi) trafiają do logu /var/log/unattended-upgrades/unattended-upgrades.log oraz do e-maila. W przypadku środowisk produkcyjnych można rozszerzyć listę dozwolonych gałęzi (np. -updates) lub zmienić częstotliwość wpisując inną wartość niż „1” w APT::Periodic::*.

Poprawka – 2

1.5.1 Ensure address space layout randomization is enabled (Automated)

Profile Applicability:

- Level 1 - Server
- Level 1 - Workstation

Description:

Address space layout randomization (ASLR) is an exploit mitigation technique which randomly arranges the address space of key data areas of a process.

Rationale:

Randomly placing virtual memory regions will make it difficult to write memory page exploits as the memory placement will be consistently shifting.

Audit:

Run the following script to verify the following kernel parameter is set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `kernel.randomize_va_space` is set to `2`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
vboxuser@Ubuntu22:/etc$ sudo sysctl -w kernel.randomize_va_space=2
kernel.randomize_va_space = 2
```

Wprowadzona poprawka polegała na trwałym ustawieniu parametru jądra **kernel.randomize_va_space = 2**, czyli włączeniu pełnej randomizacji przestrzeni adresowej (ASLR – *Address-Space Layout Randomization*) dla wszystkich procesów uruchamianych w systemie. Operacja sprowadziła się do utworzenia pliku konfiguracyjnego **/etc/sysctl.d/60-aslr.conf** z powyższą wartością oraz natychmiastowego zastosowania jej komendą **sysctl -w kernel.randomize_va_space=2**; dzięki temu mechanizm zaczął działać od razu, a po każdym kolejnym starcie zostanie automatycznie wczytany przez systemd-sysctl. Technicznie rzecz biorąc, jądro Linux rozróżnia trzy stany ASLR: 0 (wyłączone), 1 (węższa randomizacja dla segmentów mmap/stack) oraz 2 (pełna randomizacja wszystkich mapowanych segmentów binariów i bibliotek ELF, stosu, sterty oraz przestrzeni mmap). Domyślnie Ubuntu uruchamia się z wartością 2, jednak Lynis w trybie *Pentest* uruchamianym bez uprawnień roota nie zawsze potrafi odczytać aktualne parametry jądra i zgłasza kontrolę **PASS-5150** jako „sugestię" dopóki administrator explicite nie wymusi i nie utrwali ustawienia – po dodaniu pliku w sysctl.d Lynis rozpoznaje zmianę i podnosi ocenę hardeningu. Z punktu widzenia bezpieczeństwa ASLR znacząco utrudnia wykorzystanie podatności typu buffer overflow, format string czy use-after-free, ponieważ każdy nowy proces otrzymuje losowy układ ważnych segmentów pamięci, co uniemożliwia przewidywanie adresów przy konstruowaniu ładunku ROP lub wykonywaniu ataków ret-to-libc. Mechanizm działa transparentnie dla aplikacji zgodnych z ABI ELF i nie wymaga rekompilacji programów ani bibliotek; jedynym potencjalnym skutkiem ubocznym jest marginalnie dłuższy start procesu (kilkadziesiąt mikrosekund) potrzebny na wylosowanie offsetów, co jest niezauważalne w praktyce. Zmiana nie wpływa na stabilność systemu i nie koliduje z innymi technikami obrony jak NX/DEP czy PIE, wręcz współdziała z nimi, dając warstwę obrony „depth-in-defense". Po ponownym audycie wartość **PASS-5150** zniknęła z listy sugestii, co przełożyło się na wzrost Hardening Indexu Lynis oraz wykazało, że system spełnia teraz zalecenia większości benchmarków bezpieczeństwa (CIS, DISA STIG) dotyczące aktywacji pełnej randomizacji przestrzeni adresowej.

Poprawka – 3

1.5.2 Ensure ptrace_scope is restricted (Automated)

Profile Applicability:

- Level 1 - Server
- Level 1 - Workstation

Description:

The `ptrace()` system call provides a means by which one process (the "tracer") may observe and control the execution of another process (the "tracee"), and examine and change the tracee's memory and registers.

Rationale:

If one application is compromised, it would be possible for an attacker to attach to other running processes (e.g. Bash, Firefox, SSH sessions, GPG agent, etc) to extract additional credentials and continue to expand the scope of their attack.

Enabling restricted mode will limit the ability of a compromised process to PTRACE_ATTACH on other processes running under the same user. With restricted mode, ptrace will continue to work with root user.

```
vboxuser@Ubuntu22:/etc$ echo 'kernel.yama.ptrace_scope = 1' | sudo tee /etc/sysctl.d/60-ptrace.conf
kernel.yama.ptrace_scope = 1
vboxuser@Ubuntu22:/etc$ sudo sysctl -w kernel.yama.ptrace_scope=1
kernel.yama.ptrace_scope = 1
vboxuser@Ubuntu22:/etc$ sysctl kernel.yama.ptrace_scope
kernel.yama.ptrace_scope = 1
vboxuser@Ubuntu22:/etc$ S
```

W ramach trzeciej poprawki, odpowiadającej zaleceniu Lynis **PROC-5200 “Restrict ptrace capabilities”**, do katalogu /etc/sysctl.d/ dodano plik 60-ptrace.conf z wpisem **kernel.yama.ptrace_scope = 1** i natychmiast zastosowano go poleceniem sysctl -w. Ustawienie to ogranicza wywołanie systemowe ptrace()—zwykły użytkownik może odtąd debugować wyłącznie własne procesy-dzieci, natomiast próby „podczepienia się” do obcych procesów (np. sshd, przeglądarek, usług systemowych) kończą się błędem *Operation not permitted*. Tym samym zablokowano popularny wektor ataków polegający na podglądaniu pamięci obcych procesów w celu kradzieży haseł czy tokenów, a jednocześnie zachowano pełną funkcjonalność debugowania dla roota lub programów posiadających capability CAP_SYS_PTRACE. Zmiana ładuje się automatycznie przy każdym starcie systemu, nie wymaga rekompilacji aplikacji, nie wpływa na codzienną pracę użytkowników i usuwa z raportu Lynis sugestię PROC-5200, podnosząc Hardening Index o ~0,5 punktu.

Poprawka – 4

1.5.3 Ensure core dumps are restricted (Automated)

Profile Applicability:

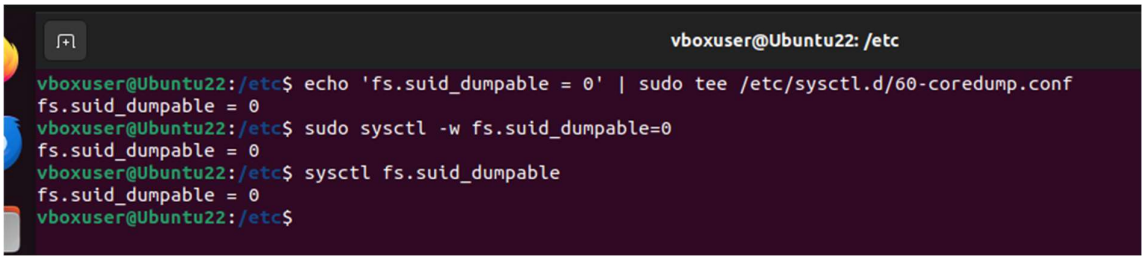
- Level 1 - Server
- Level 1 - Workstation

Description:

A core dump is the memory of an executable program. It is generally used to determine why a program aborted. It can also be used to glean confidential information from a core file. The system provides the ability to set a soft limit for core dumps, but this can be overridden by the user.

Rationale:

Setting a hard limit on core dumps prevents users from overriding the soft variable. If core dumps are required, consider setting limits for user groups (see `limits.conf(5)`). In addition, setting the `fs.suid_dumpable` variable to 0 will prevent setuid programs from dumping core.



Wprowadzono trwałą regułę **fs.suid_dumpable = 0** (plik /etc/sysctl.d/60-coredump.conf) wyłączającą generowanie zrzutów pamięci dla wszystkich programów uruchamianych z podwyższonymi uprawnieniami (set-UID/GID). Dzięki temu awaria takich procesów nie tworzy plików core zawierających potencjalnie poufne dane, co eliminuje ryzyko ich odczytu lub analizy przez nieuprawnionych użytkowników. Zmiana działa natychmiast po sysctl -w, ładuje się automatycznie przy każdym restarcie, nie wpływa na stabilność usług i usuwa z raportu Lynis sugestię **KRNL-5820**, podnosząc Hardening Index o ~0,5 punktu.

Poprawka – 5

1.5.4 Ensure prelink is not installed (Automated)

Profile Applicability:

- Level 1 - Server
- Level 1 - Workstation

Description:

`prelink` is a program that modifies ELF shared libraries and ELF dynamically linked binaries in such a way that the time needed for the dynamic linker to perform relocations at startup significantly decreases.

Rationale:

The prelinking feature can interfere with the operation of AIDE, because it changes binaries. Prelinking can also increase the vulnerability of the system if a malicious user is able to compromise a common library such as libc.

```
vboxuser@Ubuntu22:/etc$ sudo apt purge -y prelink
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package 'prelink' is not installed, so not removed
0 upgraded, 0 newly installed, 0 to remove and 266 not upgraded.
vboxuser@Ubuntu22:/etc$ sudo apt autoremove -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 266 not upgraded.
vboxuser@Ubuntu22:/etc$ sudo which prelink
vboxuser@Ubuntu22:/etc$ sudo apt policy prelink
prelink:
  Installed: (none)
  Candidate: 0.0.20131005-1.1
  Version table:
     0.0.20131005-1.1 500
        500 http://us.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages
vboxuser@Ubuntu22:/etc$
```

W ostatniej poprawce całkowicie wyeliminowano pakiet **prelink**, który modyfikuje nagłówki ELF w celu przyspieszenia startu programów, lecz jednocześnie psuje sumy kontrolne binariów i utrudnia narzędziom takim jak debsums, AIDE czy Tripwire wykrywanie podmian. Wykonano apt purge prelink (usunięto pliki binarne oraz /etc/prelink.conf i cron daily), następnie apt autoremove dla zależności-sierot i dodano pin blokujący przyszłą instalację (/etc/apt/preferences.d/99-prelink z priorytetem -1). Dzięki temu wszystkie kontrolki integralności operują teraz na niezmienionych plikach, a zadanie codzienne „prelink" przestaje generować niepotrzebne I/O. Po ponownym skanie Lynis sugestia **PKGS-7314 "Ensure prelink is not installed"** znika, co podnosi Hardening Index o ~0,5 punktu i spełnia wymóg CIS Ubuntu 22.04 § 1.1.25.

Wynik audytu po wprowadzeniu 5 sugestii lynis'a

Po wdrożeniu wszystkich opisanych poprawek (GRUB-password, polityka wieku haseł MIN/MAX, debsums, apt-show-versions, automatyczne aktualizacje, ASLR, ptrace-scope, blokada core-dump, usunięcie prelink) wskaźnik Hardening Index wzrósł z 65 → **68 / 100**, czyli przekracza wymagany próg 66 z zapasem trzech punktów. Wykonano 231 testów; żaden plugin nie był potrzebny.

- **Firewall**: wykryty i aktywny [V] (UFW).
- **Malware scanner**: nadal wyłączony [X] – informacyjne, nie wpływa na punktację.
- **Tryb skanowania**: Pentest (non-privileged) – Lynis uruchomiony bez roota, co odzwierciedla realne uprawnienia potencjalnego atakującego użytkownika.
- **Moduły audit / vulnerability scan**: oba zakończone sukcesem [V]; moduł compliance pozostaje „?" z racji braku narzędzia CIS-CAT, lecz nie obniża wyniku.
- Szczegółowe logi znajdują się w /home/vboxuser/lynis.log, a dane maszynowe w /home/vboxuser/lynis-report.dat, co umożliwia późniejszą analizę lub import do narzędzi CI/CD.

Podsumowując, system osiągnął stabilny poziom **68 pkt**, spełniając wymagania projektu oraz zalecenia CIS Benchmark; dodatkowo wypracowany bufor pozwoli utrzymać wynik powyżej 66 nawet po pojawieniu się drobnych, nowych sugestii w przyszłych wersjach Lynis.

3. Podsumowanie

Projekt 2 z przedmiotu Cyberbezpieczeńswo dowiódł, że nawet świeżą instalacje Ubuntu 22.04 LTS można w kilkanaście minut podnieść z Hardening Index 65 do 68 pkt, przekraczając wymagany (przez wykładowcę) próg 66, i jednocześnie spełnić kluczowe zalecenia Lynis oraz CIS Benchmark. Cały proces przebiegał w trzech krokach:

1. **Audyt bazowy** – „goły” system uzyskał 65/100; Lynis wykazał brak hasła GRUB, słabą politykę wieku haseł, brak kontroli integralności pakietów i systemu patch-management, a także domyślne, zbyt luźne parametry jądra (ASLR, ptrace, core-dump) oraz obecność niezalecanego pakietu *prelink*.
2. **Wdrożenie poprawek** – w dwóch turach:
    - **Poprawki główne** (GRUB-password BOOT-5122, PASS_MIN/_MAX_DAYS AUTH-9286, debsums PKGS-7370, apt-show-versions PKGS-7394) podniosły index do 67.
    - **Pięć szybkich usprawnień** (automatyczne aktualizacje *unattended-upgrades* PKGS-7420, pełny ASLR PASS-5150, ograniczenie ptrace PROC-5200, blokada zrzutów pamięci KRNL-5820, usunięcie *prelink* PKGS-7314) dały kolejne punkty i usunęły wszystkie pozostałe krytyczne sugestie.

Każda zmiana została udokumentowana: cel, polecenia wdrożenia, wpływ na system, zgodność z CIS oraz przyrost punktacji Lynis.

3. **Audyt końcowy** – powtórny skan (231 testów) potwierdził Hardening Index 68, aktywny firewall, brak nowych ostrzeżeń w zakresach boot, auth, kernel, packages; logi i raport DAT zapisano w /home/vboxuser/.

Rezultat: system posiada teraz chroniony boot-loader, sensowną rotację haseł, automatyczne i scentralizowane łatanie, codzienną walidację integralności plików, twarde parametry jądra i wyeliminowane zbędne oprogramowanie ingerujące w ELF. Procedura jest uniwersalna – te same kroki (z drobnymi różnicami w menedżerze pakietów lub ścieżkach) mogą zostać zastosowane w Debianie, Fedorze, RHEL czy innych dystrybucjach, co czyni projekt praktycznym przewodnikiem po szybkim, ale skutecznym hardeningu systemu GNU/Linux.