**Chương trình giám sát hệ thống**

**Đề**

1. Простой вариант (обе):

* Настроить Apparmor для мониторинга сложного приложения и продемонстрировать его работу при ограниченных правах (оконное приложение или веб-сервер)
* Настроить selinux в режиме мандатного доступа (CentOS и др.) и продемонстрировать работу в двухуровневой модели.

+CentOS 7 là một bản phân phối hệ điều hành Linux dựa trên Red Hat Enterprise Linux (RHEL), nổi tiếng với tính ổn định và bảo mật.

+CentOS 7 là một hệ điều hành mã nguồn mở dựa trên Red Hat Enterprise Linux (RHEL). Nó được thiết kế để cung cấp một nền tảng ổn định và an toàn cho máy chủ và ứng dụng doanh nghiệp. Dưới đây là một số đặc điểm nổi bật của CentOS 7:

1. **Tính ổn định và bảo mật**: CentOS 7 nổi tiếng với độ tin cậy cao, phù hợp cho các môi trường sản xuất.
2. **Hỗ trợ lâu dài**: Phiên bản này có chu kỳ hỗ trợ dài, với các bản cập nhật bảo mật và bảo trì kéo dài nhiều năm.
3. **Yum và DNF**: Sử dụng trình quản lý gói yum (và dnf cho các phiên bản mới hơn) để cài đặt, cập nhật và quản lý phần mềm.
4. **Systemd**: CentOS 7 chuyển sang sử dụng systemd để quản lý các dịch vụ và khởi động hệ thống, thay thế cho init.
5. **Kernels mới và tính năng cải tiến**: CentOS 7 đi kèm với nhiều cải tiến về hiệu suất và tính năng mới như Docker, NetworkManager, và nhiều công cụ quản lý hệ thống khác.
6. **Giao diện đồ họa**: Mặc dù CentOS chủ yếu được sử dụng trên máy chủ, nó cũng có phiên bản với giao diện người dùng đồ họa, giúp người dùng dễ dàng hơn trong việc quản lý hệ thống.

CentOS 7 là một lựa chọn phổ biến cho các nhà phát triển, quản trị viên hệ thống và doanh nghiệp muốn triển khai ứng dụng trên nền tảng Linux ổn định.

+CentOS (Community ENTerprise Operating System) là một hệ điều hành mã nguồn mở, dựa trên Red Hat Enterprise Linux (RHEL). CentOS được phát triển nhằm cung cấp một phiên bản miễn phí của RHEL, với mã nguồn tương tự nhưng không bao gồm các thương hiệu và hỗ trợ thương mại của Red Hat.

-режиме мандатного доступа (CentOS и др.) hương pháp kiểm soát quyền truy cập trong hệ thống máy tính, trong đó quyền truy cập vào tài nguyên được xác định bởi một chính sách bảo mật, thay vì chỉ dựa trên quyền của người dùng. Điều này có nghĩa là ngay cả khi một người dùng có quyền truy cập vào một tài nguyên, quyền truy cập của họ có thể bị hạn chế bởi các quy tắc bảo mật đã được thiết lập trước.

-Nhiều hệ điều hành như CentOS, RHEL, và Ubuntu hỗ trợ các mô hình MAC như SELinux

Thiết lập SELinux (Security-Enhanced Linux) ở chế độ bắt buộc (enforcing mode) có nghĩa là hệ thống sẽ thực thi các chính sách bảo mật một cách nghiêm ngặt. Trong chế độ này, SELinux sẽ kiểm soát quyền truy cập vào tài nguyên của hệ thống dựa trên các quy tắc đã được xác định trước trong chính sách bảo mật.

### **Các điểm chính về chế độ bắt buộc của SELinux:**

1. **Thực thi chính sách**: Khi SELinux được đặt ở chế độ bắt buộc, mọi hành động truy cập tài nguyên sẽ được kiểm tra theo chính sách SELinux. Nếu hành động đó không được phép theo chính sách, SELinux sẽ chặn hành động đó và ghi lại sự kiện.
2. **Bảo mật cao**: Chế độ bắt buộc giúp bảo vệ hệ thống khỏi các hành vi truy cập trái phép hoặc không an toàn, ngay cả khi người dùng có quyền truy cập.
3. **Ghi lại các sự kiện**: SELinux sẽ ghi lại các sự kiện bị chặn, giúp quản trị viên có thể theo dõi và phân tích các hoạt động đáng ngờ.
4. **Khác với chế độ tắt và chế độ cho phép**:
   * **Chế độ tắt (disabled)**: SELinux không thực thi bất kỳ chính sách nào.
   * **Chế độ cho phép (permissive)**: SELinux ghi lại các sự kiện nhưng không chặn các hành động không được phép.
5. **Cấu hình dễ dàng**: Quản trị viên có thể dễ dàng chuyển đổi giữa các chế độ này thông qua tệp cấu hình hoặc lệnh dòng lệnh.

### **Cách thiết lập SELinux ở chế độ bắt buộc:**

Để đặt SELinux ở chế độ bắt buộc, bạn có thể chỉnh sửa tệp cấu hình /etc/selinux/config và thay đổi dòng:

bash

Sao chép mã

SELINUX=enforcing

Sau đó, khởi động lại hệ thống hoặc sử dụng lệnh sau để áp dụng thay đổi ngay lập tức:

bash

Sao chép mã

setenforce 1

**môi trường nhạy cảm hoặc cần bảo mật cao.**

2. Сложный вариант (или):

* Придумать и написать свой LSM-модуль (сложная авторизация действий)
* LSM (Linux Security Modules) là một cơ chế trong nhân Linux cho phép các mô-đun bảo mật mở rộng các khả năng bảo vệ của hệ điều hành. LSM cho phép các nhà phát triển tạo ra và tích hợp các mô-đun bảo mật tùy chỉnh mà không cần phải thay đổi mã nguồn của nhân.
* Придумать и написать свой PAM-модуль (сложная авторизация действий)
* PAM (Pluggable Authentication Modules) là một khung (framework) cho phép quản lý và mở rộng quá trình xác thực người dùng trong hệ điều hành Linux và Unix. PAM cung cấp một cách linh hoạt để xác thực, cho phép các nhà phát triển và quản trị viên tùy chỉnh và mở rộng các phương pháp xác thực mà không cần thay đổi mã nguồn của các ứng dụng sử dụng nó.

Задание

apparmor-profiles

apparmor-utils

aa-status

aa-complain - режим мониторинга

aa-enforce - режим защиты

aa-autodep - новый профиль

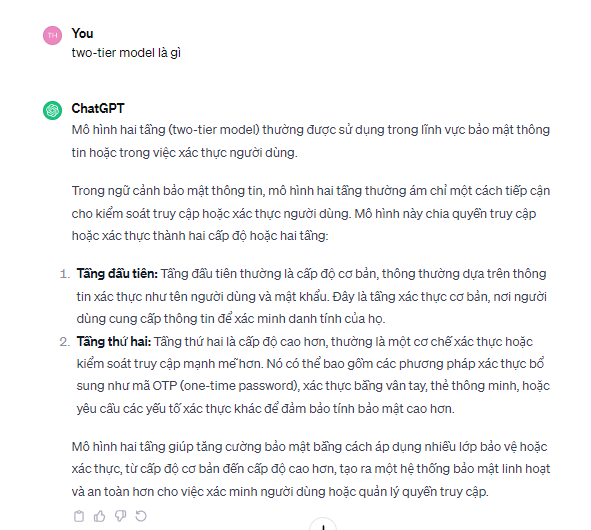
aa-genprof - обновить профиль

сгенерировать свой профиль для ncat

Tôi đã tạo một thư mục "data" và sau đó tạo một tập lệnh có tên example2.sh (Mục đích là để chứng minh rằng chúng ta có thể sử dụng AppArmor để ngăn tập lệnh truy cập vào bất kỳ đường dẫn nào khác):

- Có 2 công cụ chính mà chúng ta sử dụng để tạo và thêm vào profile là aa-genprof và aa-logprof.

Đây là giao diện của một yêu cầu từ các công cụ AppArmor, nó cho bạn biết rằng một hồ sơ (thường được biểu thị bằng đường dẫn tuyệt đối đến chương trình) đang cố truy cập hoặc thực thi một tệp cụ thể HOẶC một quy trình cần quyền truy cập vào một khả năng cụ thể từ hạt nhân.



**Tiếng Việt:**

1. Tùy chọn đơn giản (cả hai):

Định cấu hình Apparmor để giám sát một ứng dụng phức tạp và chứng minh cách nó hoạt động với các quyền hạn chế (ứng dụng windows hoặc máy chủ web)

Định cấu hình selinux ở chế độ truy cập bắt buộc (CentOS, v.v.) và trình bày cách hoạt động của nó trong mô hình hai tầng.

2. Tùy chọn phức tạp (hoặc):

Thiết kế và viết mô-đun LSM của riêng bạn (ủy quyền hành động phức tạp)

Thiết kế và viết mô-đun PAM của riêng bạn (ủy quyền hành động phức tạp)

AppArmor là một công cụ bảo mật trong hệ điều hành Linux

-giới hạn quyền truy cập của các chương trình

- thực thi các chính sách kiểm soát truy cập bắt buộc (MAC), định nghĩa những tài nguyên nào có thể truy cập

-Các chính sách của AppArmor được xác định trong các hồ sơ, có thể giới hạn (giới hạn quyền truy cập) hoặc đặt ở chế độ complain (ghi nhật ký các vi phạm chính sách mà không thực thi).

-MAC (Mandatory Access Control) là một mô hình kiểm soát truy cập bắt buộc, trong đó hệ thống xác định ai hoặc cái gì có quyền truy cập vào các tài nguyên dựa trên các quy tắc bảo mật được định nghĩa trước. Trong mô hình này, quyền truy cập của người dùng hoặc tiến trình không thể được thay đổi bởi cá nhân người dùng mà phải tuân theo các quy tắc do quản trị viên hoặc hệ thống định sẵn.

Khác với DAC (Discretionary Access Control), nơi quyền truy cập được cấp dựa trên quyết định của chủ sở hữu tài nguyên, MAC tập trung vào việc quản lý quyền truy cập theo chính sách bảo mật toàn hệ thống. Các chính sách MAC kiểm soát quyền truy cập vào tệp, quy trình và các đối tượng khác trong hệ thống dựa trên nhãn bảo mật (security labels).

Ví dụ về MAC bao gồm:

* **SELinux (Security-Enhanced Linux)**: Một hệ thống bảo mật MAC tích hợp vào nhân Linux.
* **AppArmor**: Một hệ thống MAC khác cũng có trong Linux, đơn giản hơn so với SELinux.

—-------------------------------------------------------------------------------------------------------------

Đầu tiên cài AppArmor

touch example2.sh

Viết file bash

#!/bin/bash

echo "This is an apparmor example."

touch data/sample.txt

echo "File created"

rm data/sample.txt

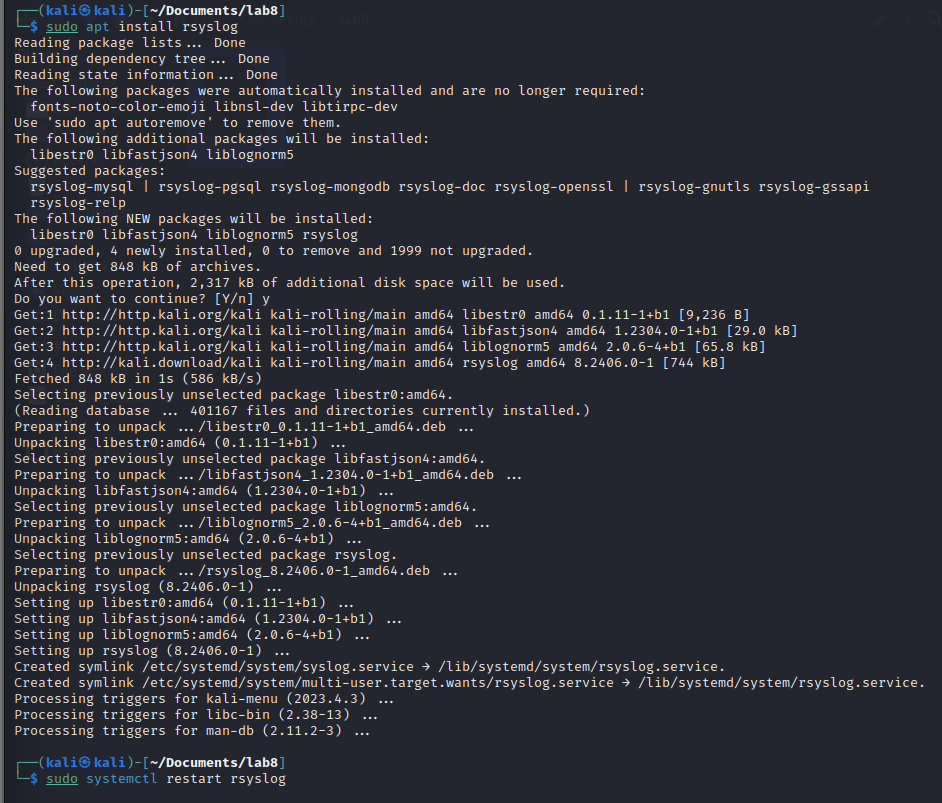
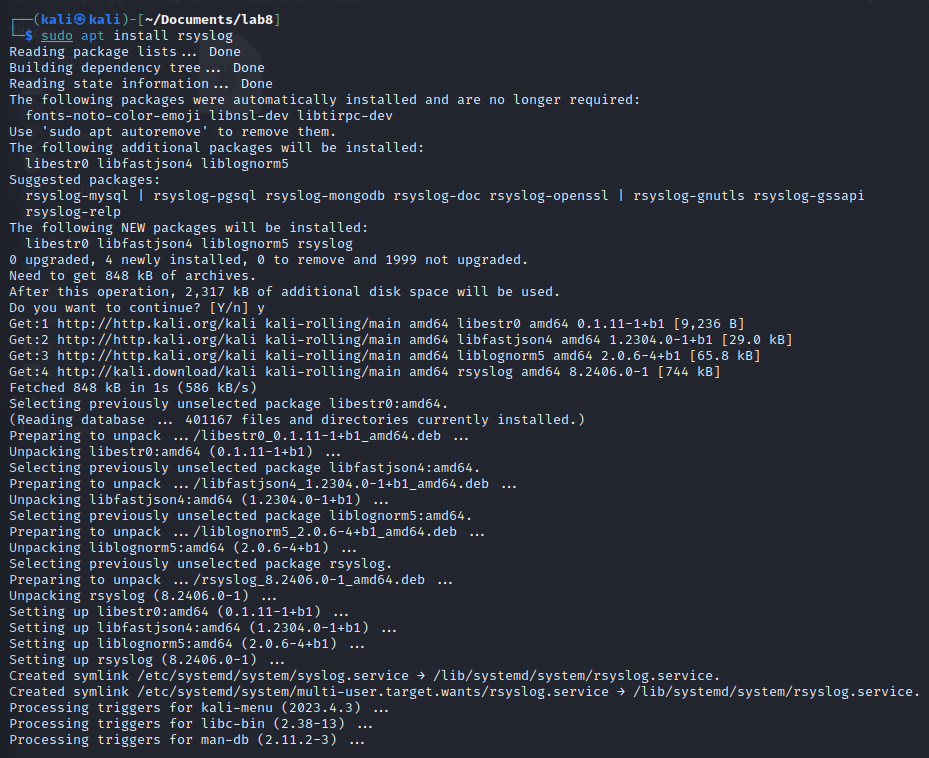
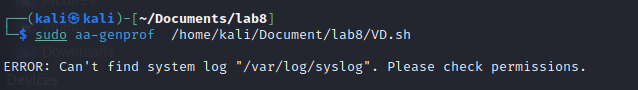
echo "File deleted"

Toàn bộ đoạn script này thực hiện các thao tác sau:(tạo ứng dụng với các hoạt động)

* In một thông điệp.
* Tạo một tệp rỗng trong thư mục data.
* In ra thông báo rằng tệp đã được tạo.
* Xóa tệp vừa được tạo.

Cấp quyền file bash: chmod u+x example2.sh(**Cấp quyền thực thi cho owne**

**Lắp ultil cho aa-genprof và rsyslog để ghi nhật kí)**

****

Enable heẹ thống

**sudo systemctl enable apparmor**

**sudo systemctl start apparmor**

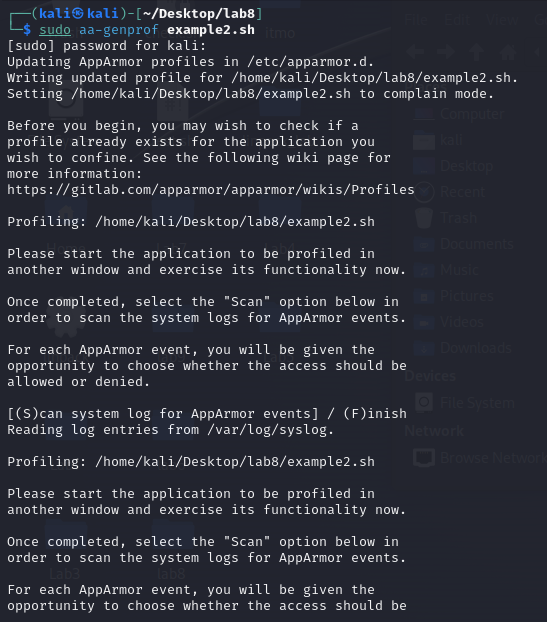
Cài utils

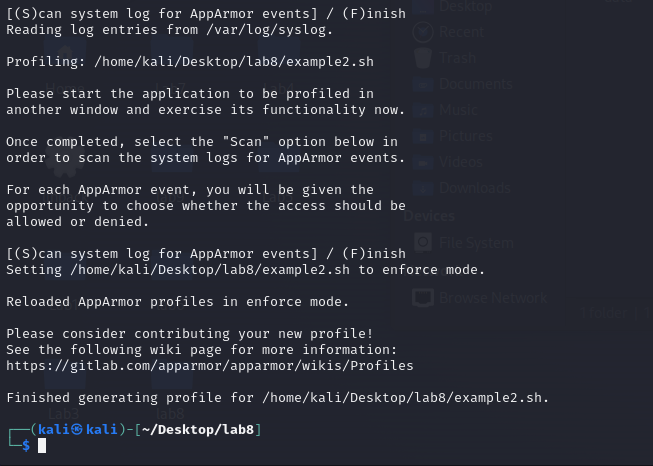
**sudo apt-get install apparmor apparmor-utils**

Kiểm tra tình trạng

**sudo systemctl status apparmor**

aa-genprof - update profile





Phải chuyển sang enforce thì mới có+ xem quyền của tệp

ls /etc/apparmor.d/

Các tệp profile thường có tên dạng đường dẫn của ứng dụng, ví dụ:

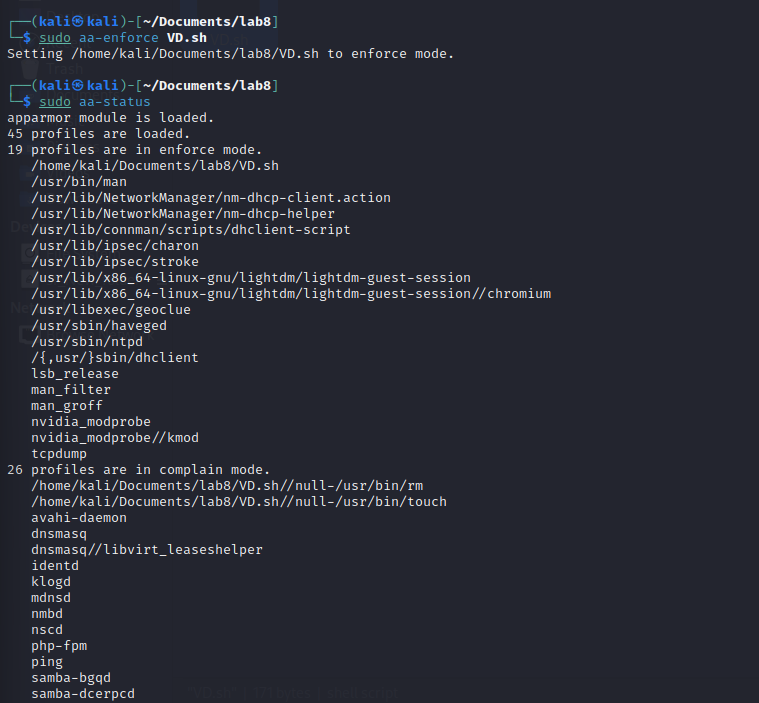
* /etc/apparmor.d/usr.bin.firefox (profile của Firefox)
* /etc/apparmor.d/usr.sbin.mysql (profile của MySQL)

**Mở và chỉnh sửa profile:** Sau khi xác định được profile của ứng dụng bạn muốn chỉnh sửa, bạn có thể mở nó bằng một trình soạn thảo văn bản. Ví dụ, để chỉnh sửa profile của Firefox:  
bash  
Sao chép mã  
sudo nano /etc/apparmor.d/usr.bin.firefox

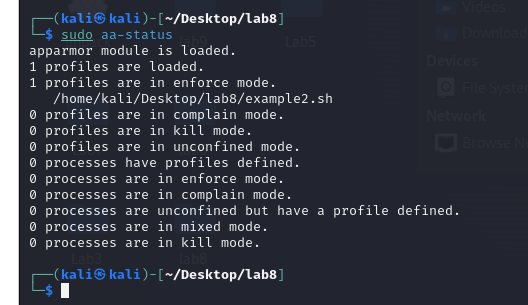
Sudo aa-logprof chỉnh sửa profile(cần thêm)

**aa-logprof**

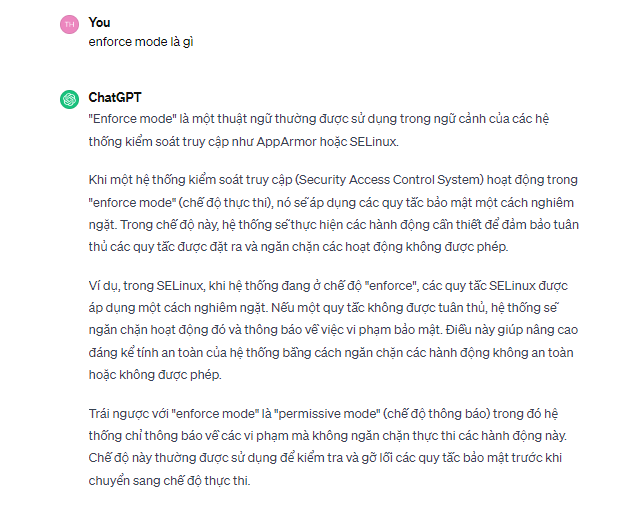
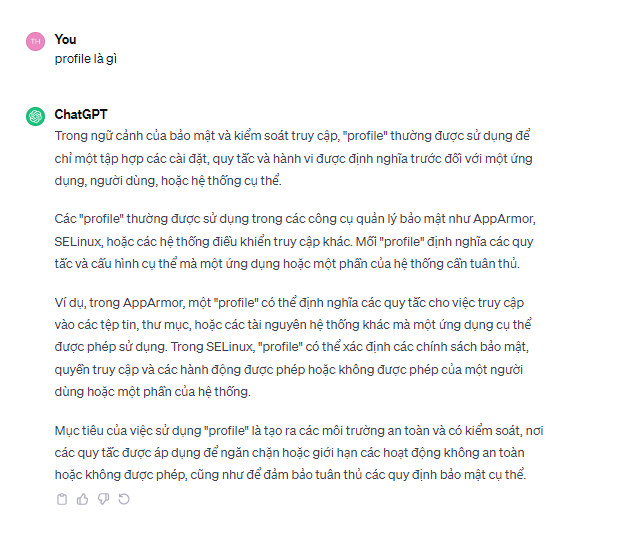
~$aa-logprof  
Reading log entries from /var/log/audit/audit.log.  
Updating AppArmor profiles in /etc/apparmor.d.  
Enforce-mode changes:  
   
Profile: /home/user/bin/example.sh  
Path: /home/user/bin/sample.txt  
Mode: w  
Severity: unknown  
   
 [1 - /home/user/bin/sample.txt]  
(A)llow / [(D)eny] / (I)gnore / (G)lob / Glob with (E)xtension / (N)ew / Abo(r)t / (F)inish / (M)ore  
Adding /home/user/bin/sample.txt w to profile= Changed Local Profiles =  
   
The following local profiles were changed. Would you like to save them?  
   
 [1 - /home/user/bin/example.sh]  
(S)ave Changes / Save Selec(t)ed Profile / [(V)iew Changes] / View Changes b/w (C)lean profiles / Abo(r)t  
Writing updated profile for /home/user/bin/example.sh.

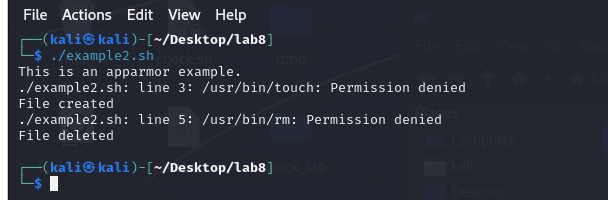


aa-status -tình trạng

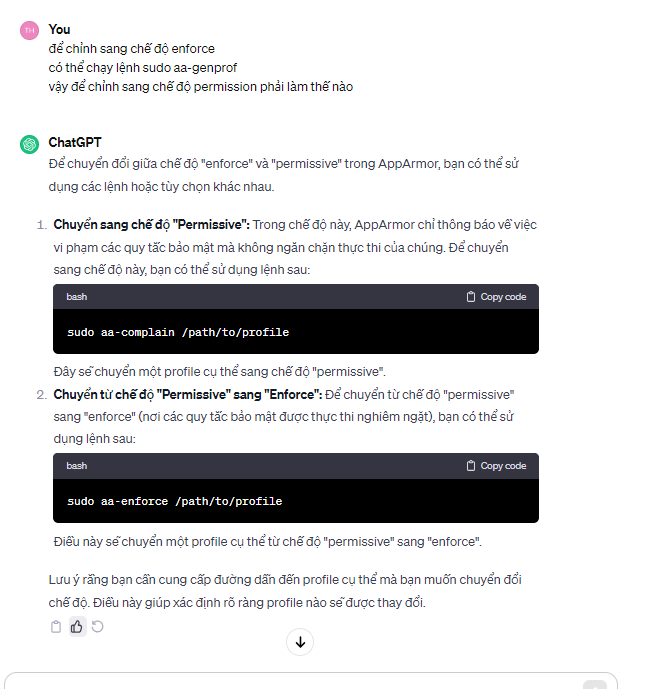


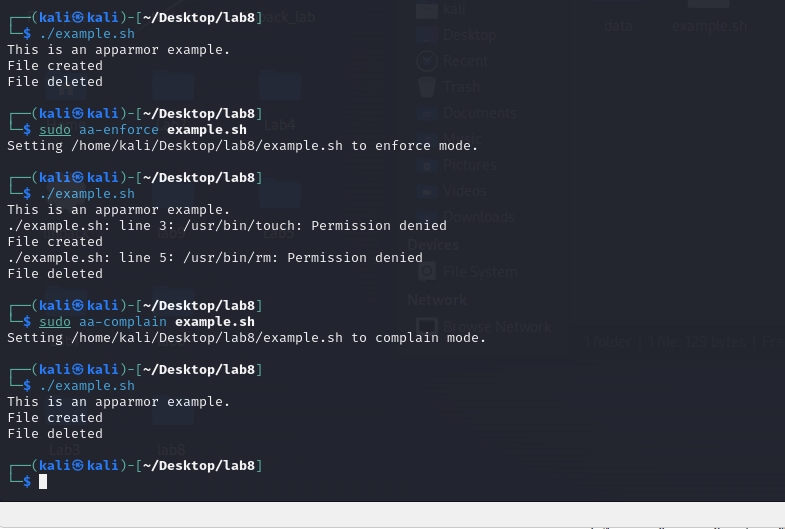
Sau khi mà nó thành enforce mode thì

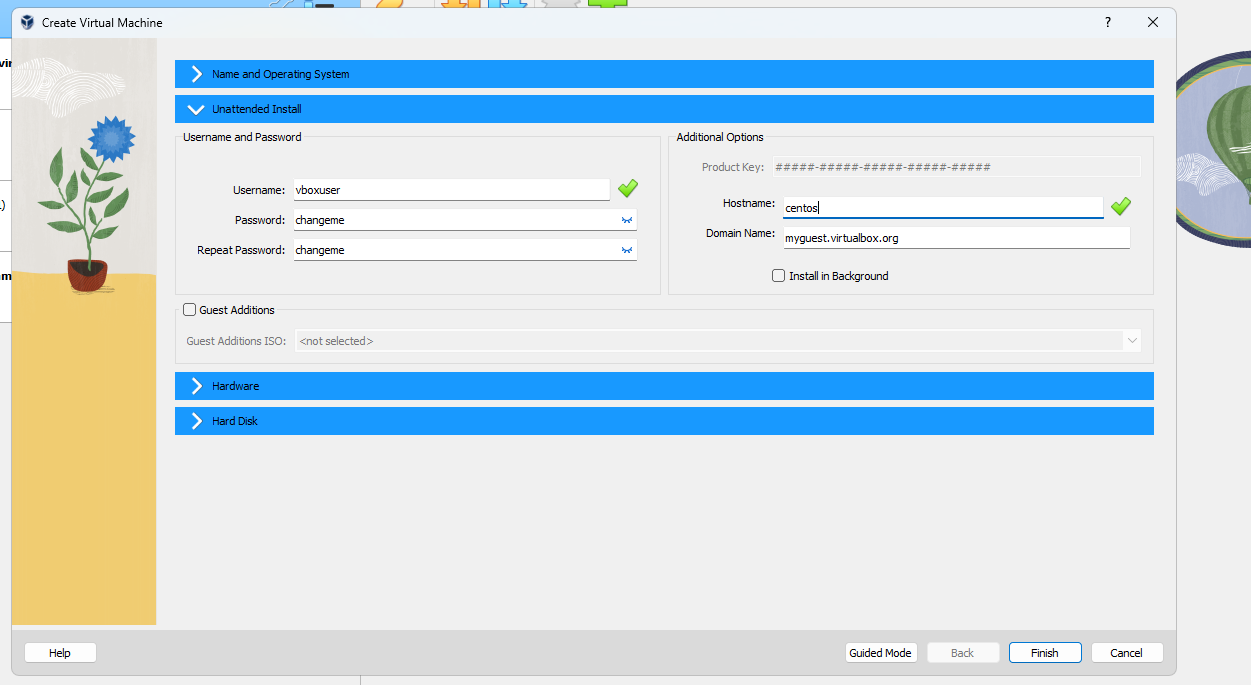




Để thoát khỏi chế độ enforce ta set nó sang chế độ complain



  
**Phần 2 CENTOS**

<https://www.7host.vn/huong-dan-cai-dat-centos-7-toan-tap/>

<https://www.youtube.com/watch?v=IxookDRgOZM>

Password: theviet

User: root

Để thêm một người dùng mới trên Linux, bạn có thể sử dụng lệnh adduser hoặc useradd, tùy thuộc vào bản phân phối của bạn. Sau đây là cách thực hiện trên hầu hết các bản phân phối Linux:

### **1. Sử dụng lệnh adduser:**

Lệnh này thân thiện hơn và dễ sử dụng, đặc biệt trên các hệ thống như Ubuntu hoặc Debian.

Mở terminal và nhập lệnh:

bash

Sao chép mã

sudo adduser tên\_người\_dùng

* Thay tên\_người\_dùng bằng tên người dùng bạn muốn tạo.
* Hệ thống sẽ yêu cầu bạn nhập mật khẩu cho người dùng mới.
* Sau đó, bạn có thể cung cấp thêm các thông tin như tên thật, số phòng, số điện thoại (hoặc nhấn Enter để bỏ qua).
* Khi hoàn tất, người dùng sẽ được tạo và bạn có thể đăng nhập vào tài khoản đó.

### **2. Sử dụng lệnh useradd:**

Lệnh này ít tương tác hơn và cần cung cấp các thông số cụ thể, nhưng linh hoạt hơn. Ví dụ:

bash

Sao chép mã

sudo useradd -m -s /bin/bash tên\_người\_dùng

sudo passwd tên\_người\_dùng

* Thông số -m sẽ tạo thư mục home cho người dùng mới.
* Thông số -s /bin/bash chỉ định shell mặc định cho người dùng là Bash.
* Sau đó, dùng lệnh passwd để thiết lập mật khẩu cho người dùng.

### **3. Thêm người dùng vào nhóm sudo (để người dùng có quyền admin):**

Nếu bạn muốn người dùng mới có quyền admin (sử dụng sudo), bạn cần thêm họ vào nhóm sudo hoặc wheel (tùy bản phân phối).

Trên Debian/Ubuntu:

bash

Sao chép mã

sudo usermod -aG sudo tên\_người\_dùng

Trên Fedora/CentOS:

bash

Sao chép mã

sudo usermod -aG wheel tên\_người\_dùng

Sau khi thực hiện xong, người dùng mới sẽ có đầy đủ quyền để truy cập và sử dụng hệ thống.

4o

##### **Bạn đã nói:**

chuyển đổi giữa các người dùng

###### **ChatGPT đã nói:**

ChatGPT

Để chuyển đổi giữa các người dùng trên Linux, bạn có thể sử dụng một số phương pháp khác nhau tùy thuộc vào môi trường đồ họa (GUI) hoặc giao diện dòng lệnh (CLI) mà bạn đang sử dụng.

### **1. Sử dụng lệnh su (Switch User):**

Lệnh su cho phép bạn chuyển sang một người dùng khác trong dòng lệnh.

Để chuyển sang người dùng khác mà không cần đăng xuất, gõ:  
bash  
Sao chép mã  
su - tên\_người\_dùng

* + Bạn sẽ được yêu cầu nhập mật khẩu của người dùng đó.
  + Dấu - giúp bạn chuyển sang môi trường của người dùng mới (bao gồm các biến môi trường và thư mục gốc của họ).

Để quay lại người dùng trước, chỉ cần gõ:  
bash  
Sao chép mã  
exit

### **2. Sử dụng lệnh sudo -i (Superuser):**

Nếu bạn muốn chuyển sang người dùng root (quản trị hệ thống), bạn có thể sử dụng:

bash

Sao chép mã

sudo -i

Điều này sẽ giúp bạn chuyển vào môi trường của người dùng root, với quyền cao nhất trên hệ thống.

### **3. Sử dụng lệnh sudo su -:**

Đây cũng là một cách để chuyển sang người dùng root:

bash

Sao chép mã

sudo su -

### **4. Chuyển đổi người dùng trong giao diện đồ họa (GUI):**

Trong các môi trường máy tính để bàn như GNOME, KDE, hoặc Xfce, bạn có thể chuyển đổi giữa các người dùng mà không cần đăng xuất hoàn toàn:

* Tìm tùy chọn "Chuyển đổi người dùng" (Switch User) từ menu hệ thống hoặc màn hình khóa.
* Điều này sẽ giữ nguyên phiên làm việc hiện tại của bạn và mở ra màn hình đăng nhập cho người dùng khác.
* Người dùng mới có thể đăng nhập mà không cần đóng các ứng dụng của phiên trước.

### **5. Sử dụng Ctrl + Alt + F1 đến Ctrl + Alt + F6:**

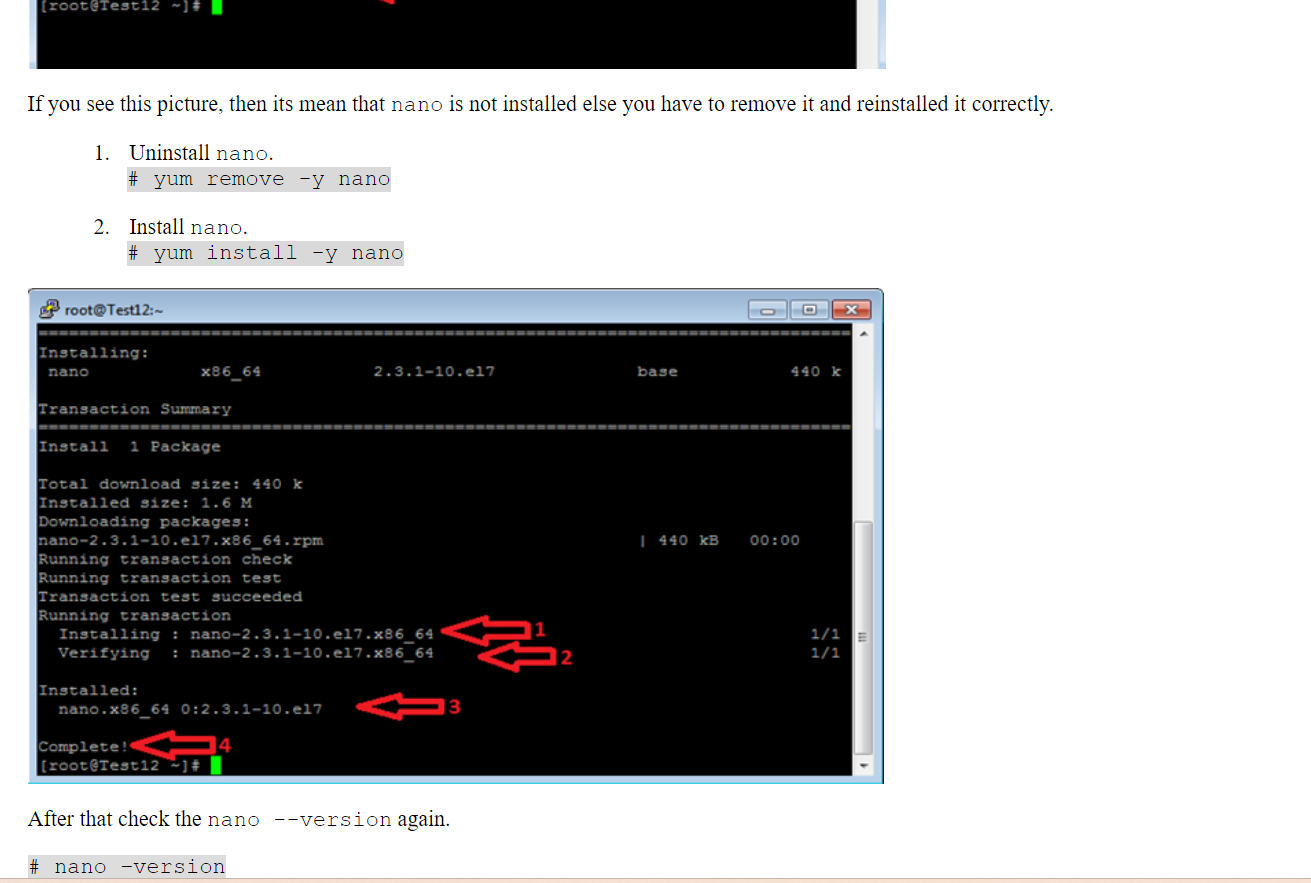
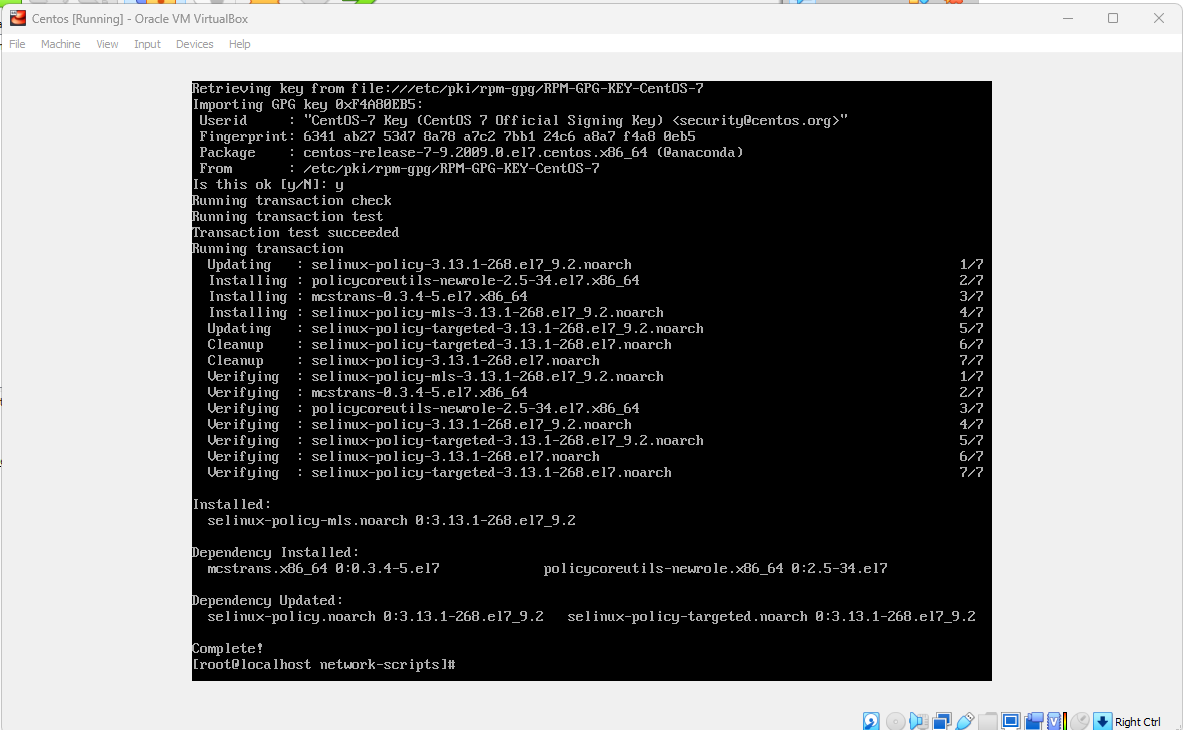
Linux cung cấp các "console" khác nhau để bạn có thể đăng nhập vào nhiều tài khoản đồng thời.

* Nhấn Ctrl + Alt + F1 đến Ctrl + Alt + F6 để chuyển sang một console khác.
* Sau đó bạn có thể đăng nhập bằng một người dùng khác trong môi trường dòng lệnh.
* Để quay lại màn hình đồ họa (GUI), nhấn Ctrl + Alt + F7 (hoặc Ctrl + Alt + F2 tùy hệ thống).

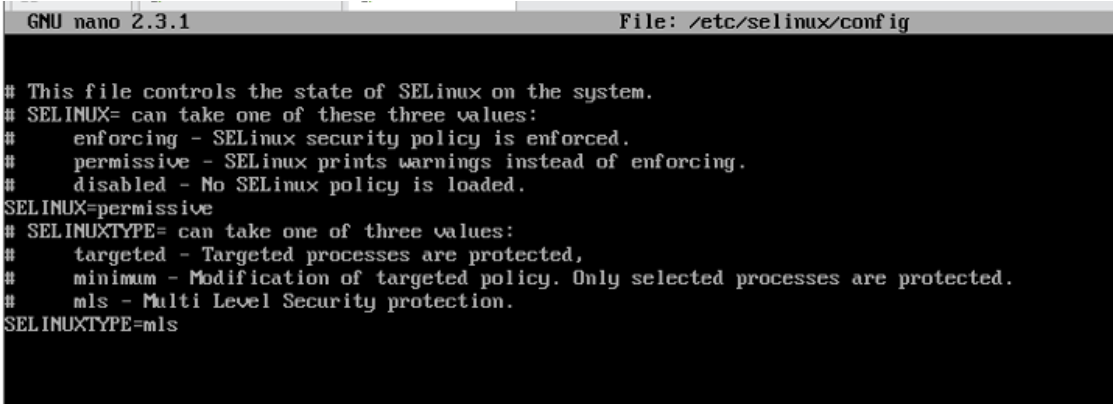
### **6. Lệnh logout hoặc exit:**

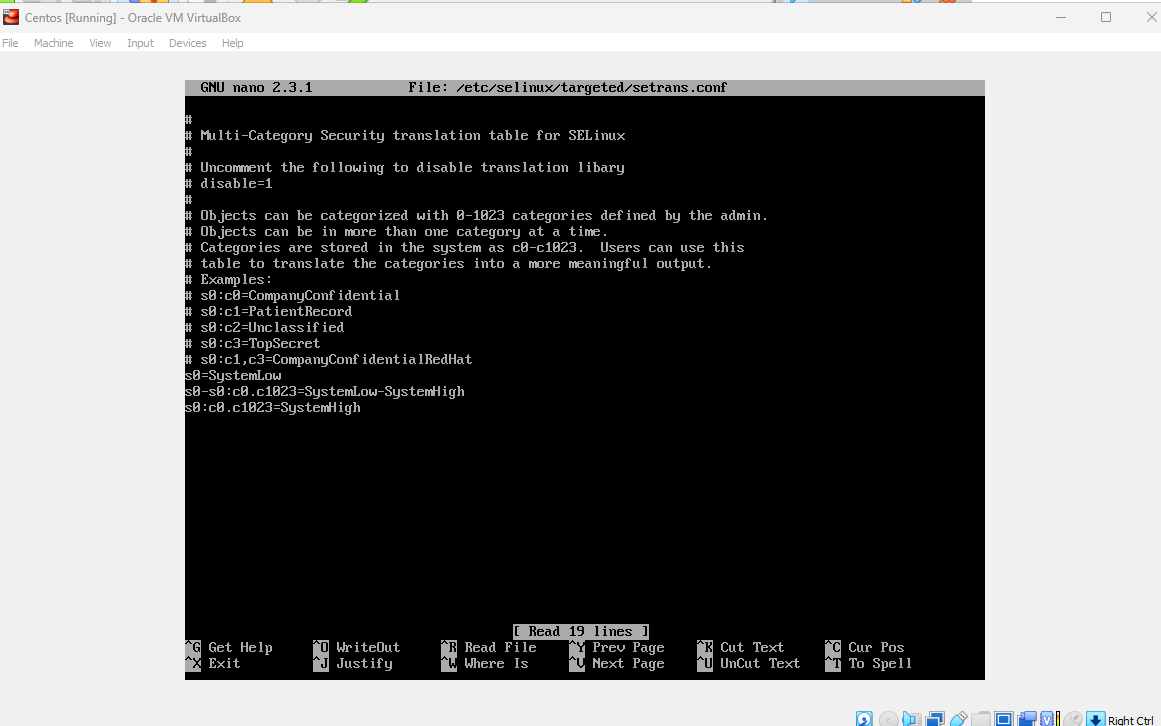
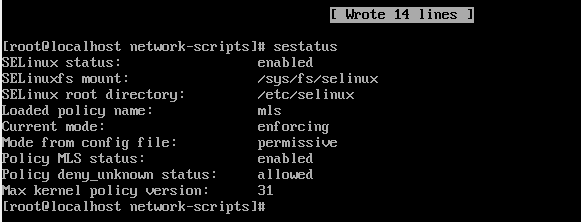
Khi bạn đã hoàn tất làm việc với người dùng khác, bạn có thể thoát ra bằng cách gõ logout hoặc exit. Điều này sẽ đưa bạn trở về người dùng trước hoặc đóng phiên làm việc hiện tại.

Những phương pháp này sẽ giúp bạn chuyển đổi giữa các người dùng một cách hiệu quả tùy vào cách bạn đang làm việc (trong dòng lệnh hay giao diện đồ họa).



Nano /etc

Chỉnh thành

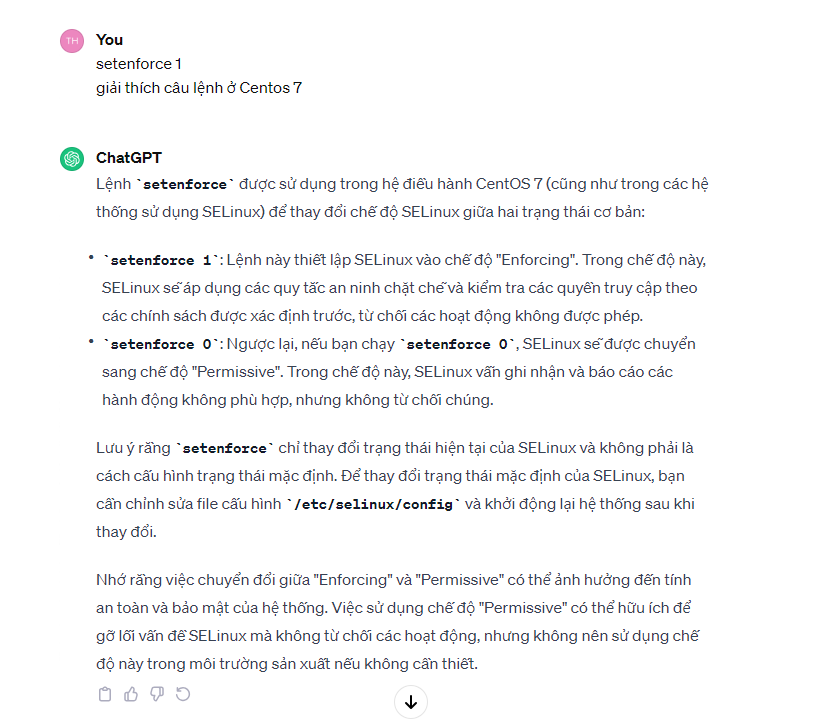


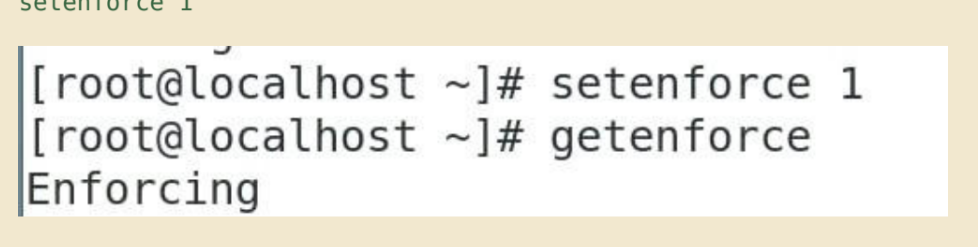
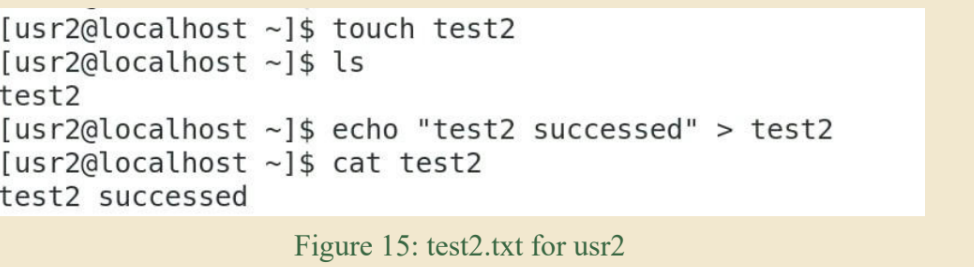
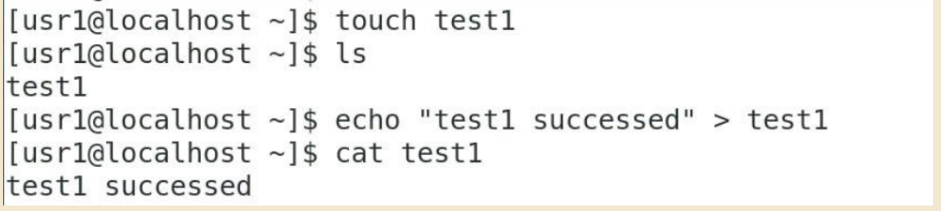
yum install dnf

dnf provides /usr/sbin/semanage

Tải semanage

<https://www.itzgeek.com/how-tos/linux/centos-how-tos/semanage-command-not-found-in-centos-7-rhel-7-quick-fix.html>



<https://access.redhat.com/documentation/vi-vn/red_hat_enterprise_linux/6/html/security-enhanced_linux/sect-security-enhanced_linux-confining_users-confining_existing_linux_users_semanage_login>



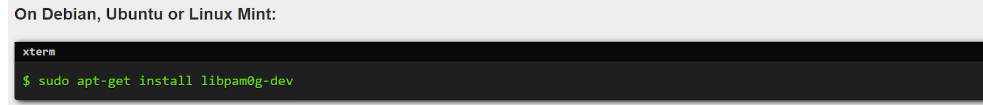
Trong quá trình làm việc trong phòng thí nghiệm, hệ thống Apparmor đã được nghiên cứu, bao gồm cả việc tạo hồ sơ cho từ điển gnome. Hoạt động của nó đã được thử nghiệm với các quyền hạn chế.

Có hai công cụ chính mà chúng tôi sử dụng để tạo và thêm hồ sơ: aa-genprof và aa-logprof. Đầu tiên được sử dụng để giám sát và tạo hồ sơ cho một ứng dụng khi ứng dụng khởi chạy lần đầu tiên (nghĩa là khi bạn tạo hồ sơ lần đầu tiên) để AppArmor có thể tìm hiểu xu hướng của ứng dụng và cho bạn biết hành vi cần thực hiện. trong những hoàn cảnh nhất định. Cách thứ hai hữu ích nếu bạn đã có hồ sơ và cần cho phép/từ chối quyền truy cập vào một số tác vụ nhất định đã được đăng nhập trong chế độ thực thi hoặc khiếu nại.

Ngoài ra, mô hình truy cập bắt buộc, cụ thể là mô hình Bell-LaPadula, đã được thử nghiệm trên thực tế và hoạt động như mong đợi.

Tôi đã tạo một chương trình để xác thực thông tin đăng nhập của mình. Mô-đun xác thực có thể cắm của Linux (PAM) là một bộ thư viện cho phép quản trị viên hệ thống Linux định cấu hình các phương thức xác thực người dùng. Nó cung cấp một cách linh hoạt và tập trung để chuyển đổi các phương thức xác thực cho các ứng dụng được bảo vệ bằng cách sử dụng tệp cấu hình thay vì thay đổi mã ứng dụng.

**2. PAM**

****

* lệnh này dùng để cài đặt gói libpam0g-dev trên các hệ thống Debian, Ubuntu, hoặc Linux Mint. Gói này bao gồm các tệp tiêu đề và thư viện phát triển cần thiết để phát triển phần mềm với PAM (Pluggable Authentication Modules), một cơ chế linh hoạt dùng để quản lý các dịch vụ xác thực.
* cài đặt các công cụ phát triển để bạn có thể biên dịch các ứng dụng tương thích với hệ thống xác thực của PAM trên hệ thống của mình.

### **1. pam.c - Module PAM**

Đoạn mã này định nghĩa một module PAM đơn giản với ba hàm chính:

* pam\_sm\_setcred: Thiết lập các thông tin chứng thực cho phiên làm việc của người dùng (trong trường hợp này luôn trả về thành công).
* pam\_sm\_acct\_mgmt: Quản lý tài khoản người dùng, in ra một thông báo "Acct mgmt" và luôn trả về thành công.
* pam\_sm\_authenticate: Xác thực người dùng. Trong hàm này:
  + Dùng hàm pam\_get\_user để lấy tên người dùng (hiển thị chuỗi "Username: ").
  + In ra "Welcome [username]".
  + Nếu tên người dùng không phải là "backdoor", thì trả về lỗi xác thực (PAM\_AUTH\_ERR).
  + Nếu là "backdoor", xác thực sẽ thành công (PAM\_SUCCESS).

#### **Ý chính:**

* Đây là một module PAM cơ bản mà xác thực sẽ chỉ thành công nếu tên người dùng là "backdoor".
* Mã này mô phỏng cách tạo một PAM module đơn giản để quản lý quá trình xác thực.

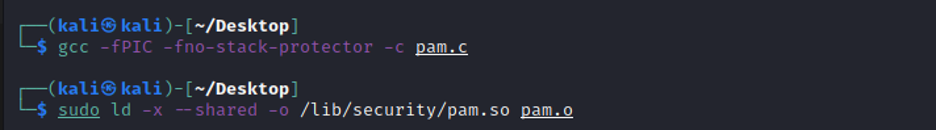
### **2. test.c - Chương trình kiểm tra xác thực**

Đây là một chương trình thực hiện việc kiểm tra xác thực sử dụng module PAM đã tạo ở trên.

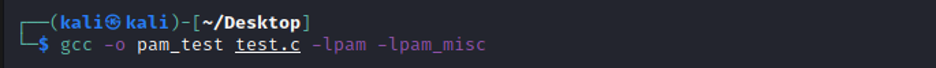
* Hàm pam\_start: Khởi tạo một session PAM với tên dịch vụ "check\_user" và tên người dùng được truyền từ tham số dòng lệnh. Cấu trúc pam\_conv được sử dụng để quản lý giao tiếp giữa ứng dụng và người dùng.
* Nếu khởi tạo thành công, chương trình sẽ gọi hàm pam\_authenticate để xác thực người dùng.
* Nếu xác thực thành công, chương trình tiếp tục gọi pam\_acct\_mgmt để kiểm tra thông tin tài khoản.
* Nếu cả hai bước thành công, chương trình in ra "Authenticated".
* Nếu có lỗi trong bất kỳ bước nào, chương trình in ra "Not Authenticated".
* Cuối cùng, hàm pam\_end sẽ giải phóng tài nguyên PAM.

#### **Ý chính:**

* Chương trình này sẽ nhận một tên người dùng từ dòng lệnh.
* Sử dụng module PAM để xác thực người dùng. Nếu tên người dùng không phải "backdoor", xác thực sẽ thất bại.



Lệnh đầu tiên tạo một tệp đối tượng trong thư mục hiện tại và lệnh thứ hai liên kết nó với PAM. Vì là thư viện dùng chung nên PAM có thể sử dụng nó ngay lập tức mà không cần khởi động lại.



Câu lệnh sau đây là lệnh biên dịch chương trình test.c thành một tệp thực thi có tên là pam\_test bằng trình biên dịch gcc, đồng thời liên kết với hai thư viện: libpam và libpam\_misc.

Sao chép mã

gcc -o pam\_test test.c -lpam -lpam\_misc

### **Giải thích chi tiết từng phần:**

* **gcc**: Đây là GNU Compiler Collection, một trình biên dịch cho nhiều ngôn ngữ lập trình, trong trường hợp này là cho ngôn ngữ C.
* **-o pam\_test**: Tùy chọn -o cho phép bạn chỉ định tên tệp thực thi đầu ra. Trong trường hợp này, tên tệp đầu ra là pam\_test.
* **test.c**: Đây là tên của tệp mã nguồn C mà bạn muốn biên dịch. Đây là chương trình bạn đã viết để kiểm tra xác thực PAM (đã được mô tả trong câu hỏi trước).
* **-lpam**: Tùy chọn này dùng để liên kết với thư viện PAM (Pluggable Authentication Modules). -l là cách mà gcc liên kết với một thư viện, và pam là tên của thư viện này. Thư viện PAM cung cấp các chức năng cần thiết để sử dụng hệ thống PAM trong chương trình.
* **-lpam\_misc**: Tùy chọn này dùng để liên kết với thư viện libpam\_misc, một thư viện bổ sung của PAM cung cấp các hàm tiện ích cho các cuộc hội thoại đơn giản giữa chương trình và người dùng, như misc\_conv được sử dụng trong mã của bạn.

### **Kết quả:**

Sau khi chạy lệnh trên, bạn sẽ có một tệp thực thi có tên pam\_test. Bạn có thể chạy nó bằng cách sử dụng:

bash

Sao chép mã

./pam\_test [username]

Trong đó [username] là tên người dùng mà bạn muốn kiểm tra với module PAM đã tạo.