

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----□□□□-----



**BÁO CÁO BÀI TẬP LỚN**  
**MÔN: NGUYÊN LÝ HỆ ĐIỀU HÀNH**

**ĐỀ TÀI: NGHIÊN CỨU VÀ TÌM HIỂU HỆ THỐNG BẢO VỆ TRONG**  
**HỆ ĐIỀU HÀNH LINUX**

Giảng viên hướng dẫn: **PhD. Nguyễn Bá Nghiễn**

Lớp: 20212IT6025004

Nhóm thực hiện: Nhóm 14

Hà Nội - 2022

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----□□□□-----



**BÁO CÁO BÀI TẬP LỚN**  
**MÔN: NGUYÊN LÝ HỆ ĐIỀU HÀNH**

**ĐỀ TÀI: NGHIÊN CỨU VÀ TÌM HIỂU HỆ THỐNG BẢO VỆ TRONG  
HỆ ĐIỀU HÀNH LINUX**

Giảng viên hướng dẫn: **PhD. Nguyễn Bá Nghiễn**

Lớp: 20212IT6025004

Nhóm thực hiện: Nhóm 14

Thành viên trong nhóm:

1. Hoàng Minh An 2020602174
2. Phạm Minh Biên 2020607303
3. Nguyễn Hoàng Minh 2020608127
4. Hoàng Kỳ Phong 2020603588
5. Đinh Tấn Hưng 2020603858

Hà Nội - 2022

---

# MỤC LỤC

<b>LỜI MỞ ĐẦU.....</b>	<b>3</b>
<b>Chương I: An toàn hệ thống .....</b>	<b>4</b>
1. Các vấn đề an toàn hệ thống .....	4
2. Các cơ chế an toàn hệ thống.....	4
2.1. Kiểm định danh tính.....	4
2.2. Ngăn chặn nguyên nhân từ phía các chương trình.....	5
2.3. Ngăn chặn nguyên nhân từ phía hệ thống.....	5
<b>Chương II. Bảo vệ hệ thống .....</b>	<b>6</b>
1.. Mục tiêu của bảo vệ hệ thống .....	6
2.. Miền bảo vệ .....	7
2.1. Khái niệm miền bảo vệ.....	7
2.2. Cấu trúc miền bảo vệ .....	7
3.Ma trận quyền truy nhập.....	8
4. Virus máy tính .....	9
4.1. Khái niệm virus.....	9
4.2. Phân loại virus .....	9
4.3. Cơ chế hoạt động virus .....	10
<b>Chương III.Các bước ban đầu để thiết lập một hệ thống bảo vệ trong Linux.....</b>	<b>11</b>
1.Sử dụng chế độ bảo mật mặc định Kernel.....	11
2.Ngắt kết nối tới các mạng không mong muốn.....	13
3.Vô hiệu hóa các Service không sử dụng.....	14
4.Sử dụng TCP Wrapper.....	18
5. An toàn cho các giao dịch trên mạng.....	20
<b>Chương IV: Cơ chế quản lí tài nguyên phân quyền .....</b>	<b>21</b>
1. Quyền truy nhập thư mục và file .....	21
2. Chế độ truy nhập .....	23
2.1. Cách xác lập tương đối.....	24
2.2. Cách xác lập tuyệt đối .....	24
3. Một số lệnh thay đổi chế độ truy cập .....	26
3.1. Thay đổi quyền sở hữu file với lệnh chown.....	26
3.2. Thay đổi quyền truy nhập file với lệnh chmod .....	27
<b>Kết luận.....</b>	<b>29</b>
<b>Tài liệu tham khảo .....</b>	<b>30</b>

---

## Danh mục hình ảnh minh họa lệnh

Hình 1: Minh họa vô hiệu hoá tính năng "ip_foward" .....	13
Hình 2: Minh họa gỡ bỏ một gói phần mềm.....	15
Hình 3: Minh họa Liệt kê danh sách những gói đã được cài đặt với thông tin chi tiết cho mỗi gói. ....	16
Hình 4: Minh họa Liệt kê thông tin chính xác các File của gói đã được chỉ định.....	16
Hình 5: Minh họa Hiển thị thông tin về một gói phần mềm .....	17
Hình 6: Minh họa Kiểm tra tính toàn vẹn cho một gói phần mềm .....	18
Hình 7: Minh họa Cài đặt một gói phần mềm mới .....	18
Hình 8: Minh họa chặn tất cả các kết nối đến và chỉ cho phép một vài máy chủ hoặc mạng cụ thể .....	19
Hình 9: Minh họa thêm địa chỉ mạng cụ thể muốn kết nối.....	20
Hình 10: Minh họa quyền xem thư mục với câu lệnh: ls-l .....	22
Hình 11: Minh họa thay đổi quyền sở hữu file với lệnh chown .....	27
Hình 12: Minh họa thay đổi quyền truy nhập file với lệnh chmod .....	28

---

## LỜI MỞ ĐẦU

Linux là một phần mềm Hệ điều hành mã nguồn mở đã và đang phát triển rất mạnh mẽ trên thế giới. Nó đã tạo ra một sự bùng nổ trong tin học và ngày càng trở nên phổ biến hiện nay. Các ưu điểm mà hệ điều hành này mang lại là vô cùng to lớn, tuy nhiên ở Việt Nam hệ điều hành này vẫn chưa được nhiều người biết đến và chưa nhiều người sử dụng một cách thành thạo nó như các hệ điều hành khác (Windows là một hệ điều hành rất nổi tiếng mà phần lớn máy tính ở Việt Nam được cài đặt và sử dụng).

Ngày nay, nhu cầu trao đổi dữ liệu qua mạng máy tính trở nên vô cùng quan trọng trong mọi hoạt động xã hội, song song với sự phát triển bùng nổ của mạng máy tính nói chung và mạng Internet nói riêng thì nguy cơ phải đối mặt với hàng loạt các đe dọa tiềm tàng như virus, sâu máy tính, các kiểu tấn công, xâm nhập, vv là rất lớn. Vấn đề bảo đảm an ninh, an toàn cho thông tin trên mạng ngày càng là mối quan tâm hàng đầu của các công ty, các tổ chức, các nhà cung cấp dịch vụ. Việc bảo vệ an toàn dữ liệu là một vấn đề cấp thiết, vì vậy việc lựa chọn một hệ điều hành phù hợp, có khả năng bảo mật tốt, độ tin cậy cao là rất quan trọng. Hệ điều hành Linux ra đời mang theo nhiều đặc tính an toàn bao hàm các cơ chế bảo mật, cùng với tính chất của một mã nguồn mở đã được đánh giá là một trong những hệ điều hành bảo mật tốt nhất hiện nay.

*Mong nhận được sự nhận xét và đóng góp của thầy để nhóm có thể hoàn thành tốt đề tài.*

**Chân thành cảm ơn thầy !**

---

## **Chương I: An toàn hệ thống**

Bảo vệ hệ thống là một cơ chế kiểm soát việc sử dụng tài nguyên của các chủ thể (tiền trình và người sử dụng) để đối phó với các tình huống lỗi có thể phát sinh trong hệ thống. Trong khi đó khái niệm an toàn hệ thống muốn đề cập tới mức độ tin cậy mà hệ thống cần duy trì khi phải đối phó không những với các vấn đề nội bộ mà còn cả với những tác động đến từ môi trường bên ngoài.

### **1. Các vấn đề an toàn hệ thống**

Hệ thống được coi là an toàn nếu các tài nguyên được sử dụng đúng quy định trong mọi hoàn cảnh, điều này khó có thể đạt được trong thực tế. Thông thường, cơ chế an toàn hệ thống bị vi phạm vì các nguyên nhân vô tình hoặc cố ý. Việc ngăn chặn các nguyên nhân cố ý là rất khó khăn và dường như không thể đạt hiệu quả hoàn toàn.

Bảo đảm an toàn ở hệ thống cấp cao như chống lại các nguyên nhân hỏa hoạn, thiên tai, mất điện... cần được thực hiện ở mức độ vật lý và nhân sự. Nếu an toàn môi trường được đảm bảo thì an toàn của hệ thống sẽ được duy trì nhờ các cơ chế của hệ điều hành.

Cần chú ý rằng nếu bảo vệ hệ thống có thể đạt độ tin cậy 100% thì các cơ chế an toàn hệ thống được cung cấp chỉ nhằm ngăn chặn bớt các tình huống bất lợi hơn là đạt đến độ an toàn tuyệt đối.

### **2. Các cơ chế an toàn hệ thống**

#### **2.1. Kiểm định danh tính**

Để đảm bảo an toàn, hệ điều hành cần phải giải quyết tốt các vấn đề kiểm định danh tính (authentication). Hoạt động của hệ thống bảo vệ phụ thuộc vào khả năng xác định các tiến trình đang xử lý. Khả năng này, đến lượt nó lại phụ thuộc vào khả năng xác định các tiến trình đang sử dụng hệ thống để có thể kiểm tra xem người dùng này được phép thao tác trên những tài nguyên nào.

Cách tiếp cận phổ biến nhất để giải quyết vấn đề là sử dụng mật khẩu (password) để kiểm tra danh tính của người sử dụng. Mỗi khi người dùng muốn sử dụng một tài nguyên, hệ thống sẽ so sánh mật khẩu của họ nhập vào với mật khẩu lưu

---

trữ nếu đúng mới được phép sử dụng tài nguyên. Mật khẩu có thể được áp dụng để bảo vệ cho từng đối tượng trong hệ thống, thậm chí cùng đối tượng sẽ có các mật khẩu khác nhau tương ứng với các quyền truy nhập khác nhau.

Cơ chế mật khẩu rất đơn giản và dễ sử dụng do đó được các hệ điều hành áp dụng rộng rãi, tuy nhiên điểm yếu nghiêm trọng của nó là khả năng bảo mật mật khẩu rất khó đạt được sự hoàn hảo. Những tác nhân tiêu cực có thể tìm ra mật khẩu của người khác nhờ nhiều cách thức khác nhau.

## **2.2. Ngăn chặn nguyên nhân từ phía các chương trình**

Trong môi trường hoạt động mà một chương trình được tạo lập bởi một người lại có thể được người khác sử dụng rất có thể sẽ xảy ra các tình huống sử dụng sai chức năng, từ đó dẫn tới hậu quả không lường trước. Hai trường hợp điển hình gây mất an toàn hệ thống có thể là:

Ngựa thành Troy: Khi người sử dụng A kích hoạt một chương trình (do người sử dụng B viết) dưới danh nghĩa của mình, chương trình này có thể trở thành “chú ngựa thành Troy” vì khi đó các đoạn lệnh trong chương trình có thể thao tác với các tài nguyên mà người sử dụng A có quyền nhưng người sử dụng B lại bị cấm.

Cánh cửa nhỏ (Trap-door): mối đe dọa đặc biệt nguy hiểm và khó chống đỡ do vô tình hoặc cố ý của các lập trình viên khi xây dựng các chương trình. Các lập trình viên có thể đã để lại một “cánh cửa nhỏ” trong phần mềm của họ để thông qua đó can thiệp vào hệ thống. Chính “cánh cửa nhỏ” này đã tạo cơ chế cho các hacker thâm nhập và phá hoại hệ thống của người sử dụng.

## **2.3. Ngăn chặn nguyên nhân từ phía hệ thống**

Hầu hết các tiến trình hoạt động trong hệ thống đều có thể tạo ra các tiến trình con. Trong cơ chế hoạt động này, các tài nguyên hệ thống rất dễ bị sử dụng sai mục đích gây mất an toàn cho hệ thống. Hai mối đe dọa phổ biến là:

Các chương trình sâu (Worm): một chương trình sâu là chương trình lợi dụng cơ chế phát sinh ra các tiến trình con của hệ thống để đánh bại chính hệ thống.

Các chương trình virus: virus là một chương trình phá hoại khá nguy hiểm đối với các hệ thống thông tin.

---

## Chương II. Bảo vệ hệ thống

### 1.. Mục tiêu của bảo vệ hệ thống

Mục tiêu của việc bảo vệ hệ thống là:

- **Bảo vệ chống lỗi của tiến trình** : khi có nhiều tiến trình cùng hoạt động, lỗi của một tiến trình j phải được ngăn chặn không cho lan truyền trên hệ thống làm ảnh hưởng đến các tiến trình khác. Đặc biệt , qua việc phát hiện các lỗi tiềm ẩn trong các thành phần của hệ thống có thể tăng cường độ tin cậy hệ thống ( reliability) .
- **Chống sự truy xuất bất hợp lệ** : Bảo đảm các bộ phận tiến trình sử dụng tài nguyên theo một cách thức hợp lệ được qui định cho nó trong việc khai thác các tài nguyên này .

Vai trò của bộ phận bảo vệ trong hệ thống là cung cấp một *cơ chế* để áp dụng các *chiến lược* quản trị việc sử dụng tài nguyên . Cần phân biệt khái niệm cơ chế và chiến lược:

- Cơ chế : xác định làm thế nào để thực hiện việc bảo vệ, có thể có các cơ chế phần mềm hoặc cơ chế phần cứng.
- Chiến lược: quyết định việc bảo vệ được áp dụng như thế nào : những đối tượng nào trong hệ thống cần được bảo vệ, và các thao tác thích hợp trên các đối tượng này

Để hệ thống có tính tương thích cao , cần phân tách các cơ chế và chiến lược được sử dụng trong hệ thống. Các chiến lược sử dụng tài nguyên là khác nhau tùy theo ứng dụng, và thường dễ thay đổi . Thông thường các chiến lược được lập trình viên vận dụng vào ứng dụng của mình để chống lỗi truy xuất bất hợp lệ đến các tài nguyên, trong khi đó hệ thống cung cấp các cơ chế giúp người sử dụng có thể thực hiện được chiến lược bảo vệ của mình.



---

## 2.. Miền bảo vệ

### 2.1. Khái niệm miền bảo vệ

Một hệ thống máy tính bao gồm:

- Chủ thể (subject's) bao gồm các tiến trình và người sử dụng
- Khách thể (object's) là các tài nguyên của máy tính (như bộ nhớ, ổ đĩa, dữ liệu ... )

Để đảm bảo an toàn hệ thống thì hệ điều hành chỉ cho phép các chủ thể truy nhập tới các khách thể mà nó có quyền sử dụng và vào những thời điểm cần thiết (nguyên lý **need - to - know**) nhằm hạn chế các lỗi xảy ra do tranh chấp tài nguyên.

Mỗi chủ thể trong hệ thống sẽ hoạt động trong một miền bảo vệ (protection domain) nào đó. Một miền bảo vệ sẽ xác định các khách thể mà chủ thể trong miền đó được phép truy nhập và thực hiện các thao tác.

### 2.2. Cấu trúc miền bảo vệ

Các khả năng thao tác mà chủ thể có thể thực hiện trên các thể được gọi là quyền truy nhập(access right). Mỗi quyền truy nhập được định nghĩa bởi một bộ hai thành phần<đối tượng{quyền thao tác}>.Như vậy ta có thể hình dung miền bảo vệ là một tập hợp các quyền truy nhập,xác định các thao tác mà chủ thể có thể thực hiện trên các khách thể . Các miền bảo vệ khác nhau có thể giao nhau một số quyền truy nhập.

Một tiến trình hoạt động và miền bảo vệ có thể tồn tại hai mối liên kết:

Liên kết tĩnh : trong suốt thời gian tồn tại của tiến trình trong hệ thống , tiến trình chỉ hoạt động trong một miền bảo vệ . Trong trường hợp tiến trình trên những tập tài nguyên khác nhau. Như vậy trong liên kết tĩnh , miền bảo vệ phải xác định ngay từ đầu các quyền truy cập cho tiến trình trong tất các giai đoạn xử lý. Điều này khiến cho tiến trình sẽ được dư thừa quyền trong một giai đoạn xử lý nào đó và vi phạm nguyên lý need- to- know. Để đảm bảo được nguyên lý này phải có khả năng cập nhật nội dung miền bảo vệ qua các giai đoạn xử lý khác nhau để đảm bảo các quyền tối thiểu của tiến trình trong miền bảo vệ tại một thời điểm.

Liên kết động : Cơ chế này cho phép tiến hành chuyển đổi từ miền bảo vệ này sang miền bảo vệ khác trong suốt thời gian tồn tại trong hệ thống của nó. Để tuân thủ nguyên lý need-to-know, thay vì sửa đổi nội dung miền bảo vệ, hệ thống có thể tạo ra các miền bảo vệ mới với nội dung thay đổi tùy theo từng giai đoạn xử lý của tiến trình và chuyển tiến trình sang hoạt động tại các miền bảo vệ phù hợp với từng thời điểm.

### 3. Ma trận quyền truy nhập

Sử dụng một bảng toàn cục bao gồm các bộ ba thành phần <miền bảo vệ, khách thể, quyền truy nhập>. Mỗi khi thực hiện thao tác M trên khách thể Oj trong miền bảo vệ Di, cần tìm trong bảng toàn cục một bộ ba <Di, Oj, Rk> mà M thuộc Rk (tập các quyền truy nhập). Nếu tìm thấy, thao tác M được phép thi hành, ngược lại xảy ra lỗi truy nhập.

Object Subject	F1	F2	F3	Printer
D1	Đọc/ghi	Xử lý	Đọc/ghi	
D2		Ghi		In
D3	Xử lý		Đọc	In

Bảng 1: Ma trận quyền truy nhập

#### Danh sách quyền truy nhập (Access Control List - ACL):

Mỗi cột trong ma trận quyền truy nhập được xem như một danh sách các quyền truy nhập tới một khách thể bao gồm <miền bảo vệ, các quyền truy nhập>, xác định các quyền truy nhập được quy định trong từng miền bảo vệ có thể tác động trên khách thể.

#### Danh sách khả năng (Capability List):

Mỗi dòng trong ma trận quyền truy nhập tương ứng với một miền bảo vệ sẽ được tổ chức thành một danh sách khả năng bao gồm các khách thể và các thao tác được phép thực hiện trên khách thể khi chủ thể hoạt động trong miền bảo vệ.

---

Chủ thể chỉ có thể thực hiện thao tác  $M$  trên khách thể  $O_j$  trong miền bảo vệ  $D_i$  nếu trong danh sách khả năng của  $D_i$  có chứa khả năng tương ứng của  $O_j$  và trong danh sách quyền truy nhập của khách thể  $O_j$  một bộ hai  $\langle D_i, R_k \rangle$  mà  $M$  thuộc  $R_k$ .

### **Cơ chế khoá và chìa (A Lock/ Key Mechanism):**

Phương pháp này thực chất là sự kết hợp giữa danh sách quyền truy nhập và danh sách khả năng.

Mỗi khách thể sở hữu một danh sách các mã nhị phân được gọi là khoá (lock); Tương ứng mỗi miền bảo vệ sẽ sở hữu một danh sách mã nhị phân gọi là chìa (key). Một chủ thể hoạt động trong miền bảo vệ chỉ có thể truy nhập tới một khách thể nếu miền bảo vệ sở hữu một chìa tương ứng với một khoá trong danh sách của khách thể.

## **4. Virus máy tính**

### **4.1. Khái niệm virus**

Virus máy tính là một chương trình có khả năng gián tiếp tự kích hoạt, tự lan truyền trong môi trường của hệ thống tính toán và làm thay đổi môi trường hệ thống hoặc cách thực hiện chương trình.

Virus tự kích hoạt là lan truyền trong môi trường làm việc của hệ thống mà người sử dụng không hề hay biết. thông thường, virus nào cũng mang tính chất phá hoại, nó gây ra lỗi khi thực hiện chương trình, điều này có thể dẫn đến việc chương trình hoặc dữ liệu bị hỏng, không khôi phục được, thậm chí chúng có thể bị xóa. Như vậy, virus là chương trình thông minh, mang yếu tố tự thích nghi, lan truyền xa và do đó khả năng phá hoại là rất lớn.

Một số biểu hiện của máy tính bị nhiễm virus:

- Hệ thống hoạt động không ổn định.
- Các chương trình ứng dụng có thể không hoạt động hoặc hoạt động sai chức năng.
- Dữ liệu bị sai lệch.
- Kích thước các file tăng.
- Xuất hiện các file lạ trên đĩa.

....

### **4.2. Phân loại virus**

---

Dựa vào cơ chế lan của virus, người ta có thể phân thành một số loại như sau:

- Boot virus ( B – virus) : là những virus chỉ lây lan vào các boot sector hoặc master boot record của các ổ đĩa.
- File virus ( F – virus) : là những virus lây lan vào các file chương trình của người sử dụng (các file COM hoặc EXE).
- Virus lưỡng tính (B/F –virus) : là những virus vừa có thể lây lan vào các boot sector hoặc master boot record, vừa có thể lây lan vào các file chương trình.
- Macro virus : là những virus được viết bằng các lệnh macro, chúng thường lây nhiễm vào các file văn bản hoặc bảng tính...
- Trojan virus (Trojan Horse) : là những virus nằm tiềm ẩn trong hệ thống máy tính dưới dạng các chương trình ứng dụng nhưng trên thực tế, khi chương trình này được kích hoạt, các lệnh phá hoại sẽ hoạt động.
- Worm (sâu) : sâu được di chuyển trong hệ thống mạng từ máy tính này sang máy tính khác. Nhiệm vụ chính của chúng là thu thập các thông tin cá nhân người sử dụng để chuyển về một địa chỉ xác định cho người điều khiển.
- Virus máy tính Hijacker ( Chuyên tấn công trình duyệt ) : là những virus khiến máy tính gặp phải hiện tượng xuất hiện nhiều dạng trang web khác nhau khi thực hiện một thao tác tìm kiếm bất kỳ trên trình duyệt của mình. Dạng Virus này được xem như là thường gặp phải vì nó thường ẩn trong những file dữ liệu được người dùng tải xuống dưới dạng miễn phí.

#### **4.3. Cơ chế hoạt động virus**

Về cơ chế hoạt động của virus, chúng ta có thể hình dung quá trình như sau:

Khi đọc một đĩa hoặc thi hành một chương trình bị nhiễm virus, nó sẽ tạo ra một bản sao đoạn mã của mình và nằm thường trú trong bộ nhớ của máy tính. Khi đọc một đĩa hoặc thực hiện một chương trình, đoạn mã virus nằm trong bộ nhớ sẽ kiểm tra đĩa/file đó đã tồn tại đoạn mã chưa? Nếu chưa thì tạo một bản sao khác lây nhiễm nào đĩa/file.

---

Ví dụ về cơ chế chiếm quyền điều khiển của B- virus: khi máy tính bắt đầu khởi động, mọi thanh ghi CPU sẽ bị xóa. Các thanh ghi đoạn được gán giá trị 0FFFh còn tất cả các thanh ghi còn lại sẽ bị xóa về 0, ngay lúc này cặp CS:IP trỏ đến địa chỉ 0FFFh. Tại địa chỉ này, một lệnh JMP FAR chuyển quyền điều khiển đến một đoạn chương trình định sẵn trong ROM BOIS. Đoạn chương trình này sẽ thực hiện quá trình POST ( tự kiểm tra khi khởi động ).

## **Chương III.Các bước ban đầu để thiết lập một hệ thống bảo vệ trong Linux**

### **1.Sử dụng chế độ bảo mật mặc định Kernel.**

Trước tiên ta cần đặt câu hỏi Kernel là gì? Khái niệm kernel ở đây nói đến những phần mềm, ứng dụng ở mức thấp (low-level) trong hệ thống, có khả năng thay đổi linh hoạt để phù hợp với phần cứng. Chúng tương tác với tất cả ứng dụng và hoạt động trong chế độ user mode, cho phép các quá trình khác – hay còn gọi là server, nhận thông tin từ các thành phần khác qua inter-process communication (IPC).

Trong Kernel của một số hệ thống Linux mới hiện giờ có cấu hình sẵn một vài Rules chuẩn với mục đích cung cấp những thông số căn bản nhất để cấu hình cho hệ thống dành cho những Admin không có nhiều kinh nghiệm về bảo mật hệ thống. Các File và thông số đó thường được chứa ở /proc/sys. Về căn bản giao thức IPV4, bên trong /proc/sys/net/ipv4 cung cấp các tính năng căn bản:

`icmp_echo_ignore_all`: Vô hiệu hoá tất cả các yêu cầu phản hồi ICMP ECHO. Sử dụng tùy chọn này nếu như bạn không muốn hệ thống của mình trả lời các yêu cầu Ping.

`icmp_echo_ignore_broadcasts`: Vô hiệu hoá tất cả các yêu cầu phản hồi ICMP ECHO trên Broadcast và Multicast. Tùy chọn này được sử dụng để ngăn chặn nguy cơ hệ thống của bạn có thể bị lợi dụng khai thác cho những cuộc tấn công DDOS.

`ip_forward`: Cho phép hay không cho phép sự chuyển tiếp IP giữa các giao diện mạng trong hệ thống của bạn. Tùy chọn này được sử dụng khi bạn muốn Server của mình hoạt động như Router.

`ip_masq_debug`: Kích hoạt hay vô hiệu hoá quá trình gỡ lỗi cho IP Masquerading

---

`tcp_syncookies`: Tùy chọn này được sử dụng để bảo vệ hệ thống của bạn chống các cuộc tấn công sử dụng kỹ thuật ngập SYN đã từng gây kinh hoàng một thời trên Internet.

`rp_filter`: Chứng thực và xác định địa chỉ IP nguồn hợp lệ. Tùy chọn này được sử dụng để bảo vệ hệ thống của bạn chống lại các cuộc tấn công giả mạo địa chỉ IP "IP Spoof".

`secure_redirects`: Chỉ chấp nhận chuyển tiếp những thông điệp ICMP cho những Gateway tin tưởng trong danh sách.

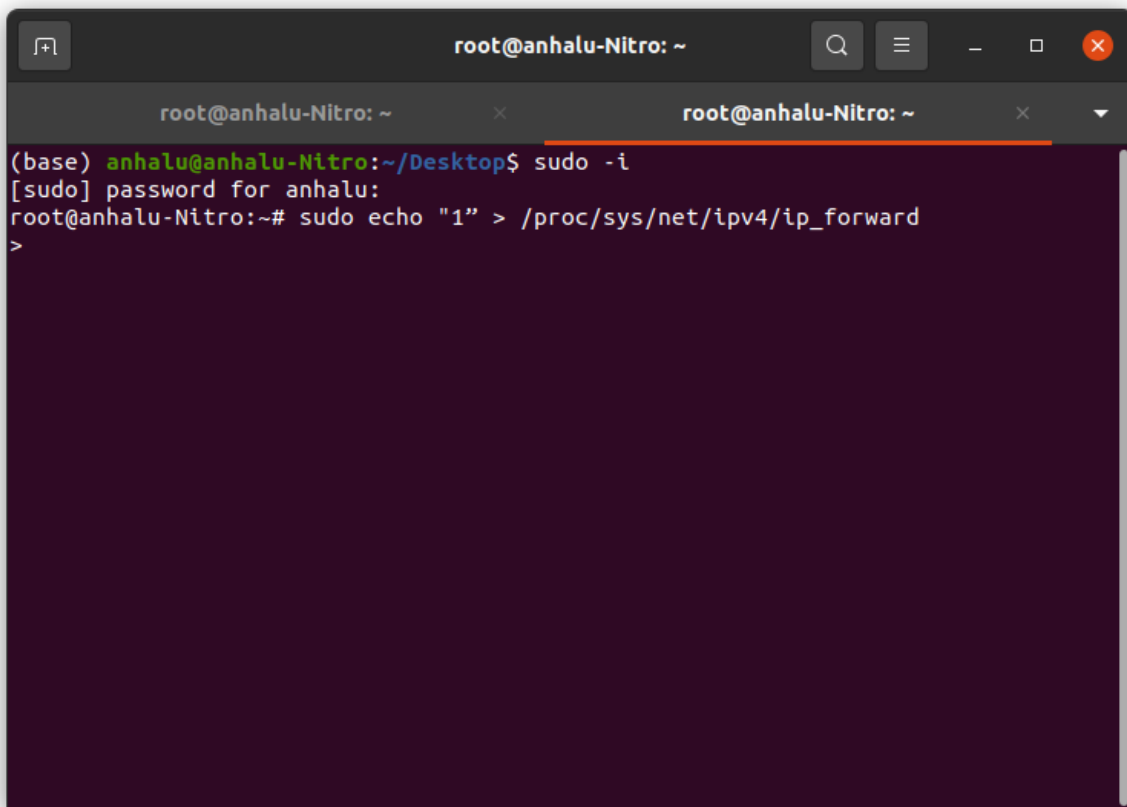
`log_martians`: Ghi lại những Packet không được xử lý bởi Kernel.

`accept_source_route`: Xác định xem liệu có phải những Source Routed Packet được chấp nhận hay từ chối. Để an toàn bạn lên vô hiệu hoá tính năng này.

Trong hệ thống Redhat, ở `/etc/sysctl.conf` chứa thông tin về những thiết bị mặc định được xử lý ngay khi khởi động hệ thống, những thông số đó được đọc, điều khiển và thực thi bởi `/usr/bin/sysctl`.

Nếu bạn muốn vô hiệu hoá tính năng `"ip_foward"` đơn giản bạn chỉ việc sử dụng lệnh: `root@localhost# echo "0" > /proc/sys/net/ipv4/ip_forward`

Tương tự để kích hoạt tính năng nào bạn chỉ việc thay giá trị `"0"` bằng `"1"`...



```
(base) anhalu@anhalu-Nitro:~/Desktop$ sudo -i
[sudo] password for anhalu:
root@anhalu-Nitro:~# sudo echo "1" > /proc/sys/net/ipv4/ip_forward
>
```

Hình 1: Minh họa vô hiệu hoá tính năng "ip\_foward"

## 2. Ngắt kết nối tới các mạng không mong muốn.

Bước đầu tiên trong việc bảo mật cho một hệ thống Linux là ngắt kết nối hay vô hiệu hóa tất cả các mạng ma và các dịch vụ mà bạn không cần. Một cách cơ bản, bất kì cổng mạng nào mà hệ thống đang chờ kết nối đều có thể nguy hiểm, bởi vì đó có thể là một sự khai thác bảo mật dựa vào một mạng ma sử dụng cổng đó. Cách nhanh nhất để tìm ra những cổng nào được mở là sử dụng netstat -an, như được chỉ ra dưới đây (tuy nhiên chúng ta sẽ bỏ đi một vài dòng): # netstat -an

Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address Foreign Address State

tcp 0 0 0.0.0.0:7120 0.0.0.0:\* LISTEN

tcp 0 0 0.0.0.0:6000 0.0.0.0:\* LISTEN

tcp 0 0 0.0.0.0:22 0.0.0.0:\* LISTEN

Ở đây chúng ta thấy rằng hệ thống này đang nghe ngóng cho những kết nối trên cổng 7120, 600 và 22. Nhìn vào /etc/services, hoặc sử dụng -p với lệnh netstat, có thể

---

thường tiết lộ mạng ma nào đang giao tiếp với những cổng đó. Trong trường hợp này nó là X font server, X Window System server và SSH.

Nếu bạn nhìn thấy rất nhiều những cổng khác mở - cho những thứ như telnetd, sendmail...hãy tự hỏi bạn xem liệu bạn có thực sự cần những deamons đó chạy không. Qua thời gian, những vấn đề bảo mật sẽ càng bộc lộ, và trừ phi bạn có nhiều kinh nghiệm trong việc theo dõi tất cả những cập nhật bảo mật, nếu không hệ thống của bạn có thể bị tổn thương từ những cuộc tấn công. Bởi vậy, telnetd, ftpd, và rshd tất cả bảo gồm gửi các mật khẩu thông qua mạng Internet cho việc chứng thực, một giải pháp tốt hơn là sử dụng sshd, nó mã hóa dữ liệu và sử dụng một cơ chế chứng thực mạnh hơn. Thậm chí nếu bạn chưa bao giờ sử dụng telnetd thì để nó chạy trên hệ thống của bạn không phải là một ý kiến hay trong trường hợp một ai đó cố tìm một cách phá vỡ nó.

Ngắt các dịch vụ thường phải chỉnh sửa các file cấu hình tương ứng cho bản phân phối của bạn và khởi động lại hệ thống. Trên các hệ thống Red Hat, ví dụ, nhiều deamons được bắt đầu bằng các kịch bản trong thư mục /etc/rc.d/init.d

Đổi tên hoặc gỡ bỏ những file kịch bản đó có thể ngăn chặn deamons tương ứng từ lúc khởi động. Những daemon khác được khởi động bởi inetd hoặc xinetd trong việc trả lời các kết nối mạng; sửa những cấu hình của những hệ thống đó có thể giới hạn tập hợp các daemon chạy trên hệ thống của bạn.

Nếu bạn thực sự cần một dịch vụ chạy trên máy của bạn (chẳng hạn như X server), hãy tìm những cách để ngăn chặn các kết nối tới dịch vụ đó từ những máy chủ không mong muốn, chẳng hạn, sẽ là an toàn nhất để cho phép kết nối ssh chỉ từ những máy chủ tin tưởng, chẳng hạn chỉ từ những máy trong mạng nội bộ của bạn. Trong trường hợp của X server và X font server, cái mà chạy trên nhiều máy Linux, thường không có lý do để cho phép các kết nối tới những daemon từ bất kì thứ gì ngoài chính mạng cục bộ. Lọc các kết nối tới những daemon có thể được thực hiện bởi TCP wrapper hoặc IP filtering, chúng ta sẽ mô tả trong phần sau.

### **3. Vô hiệu hóa các Service không sử dụng.**

Để tránh tình trạng có lỗ hổng bảo mật sau này bạn lên vô hiệu hoá và gỡ bỏ những chương trình, Service không dùng đến trên hệ thống của mình. Bạn có thể sử



---

dụng các công cụ quản lý để hiển thị danh sách những gói phần mềm nào đã được cài đặt để thực hiện việc này (Redhat Package Manager - Linux )

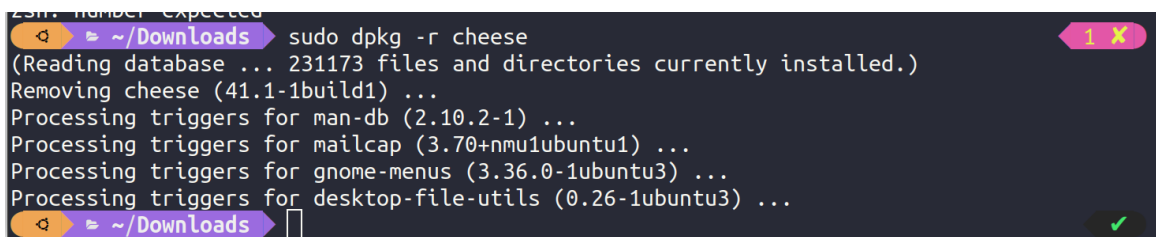
Về cơ bản! các Service được định nghĩa hoạt động bởi inetd (trên một số hệ thống Linux mới nó có thể là xinetd). Nội dung Service được định nghĩa hoạt động bởi inetd được chứa ở /etc/inetd.conf. Mỗi Service được định nghĩa bằng sau ký tự "#"...Bạn có thể vô hiệu hoá Service không sử dụng.

Thư mục /etc/rc\*.d và /etc/rc.d/rc\* là nơi chứa các Shell Script và các thông số để điều khiển sự thực hiện của Network và Service trong suốt thời gian nó hoạt động. Bạn có thể xoá bỏ hết những thứ liên quan đến những Service mà bạn không cần sử dụng. Đối với hệ thống Redhat, SuSE, Mandrake...bạn có thể sử dụng lệnh:

- root@localhost#chkconfig --list
- root@localhost#chkconfig --del <name>

### **Gỡ bỏ một gói phần mềm:**

- root@localhost# rpm -e <package-name>
- root@localhost# dpkg -r <package-name>



Hình 2: Minh họa gỡ bỏ một gói phần mềm

### **Liệt kê danh sách những gói đã được cài đặt với thông tin chi tiết cho mỗi gói:**

- root@localhost# rpm -qvia
- root@localhost# dpkg -l

```
~/Doc/HIT_Product on git develop_15_fix_bug +1 !25 ?17 sudo dpkg -l
[sudo] password for minhbien:
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
||/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name                               Version
+++=====
ii accountsservice                     22.07.5-2ubuntu1.3
ii acl                                 2.3.1-1
ii acpi-support                         0.144
ii acpid                               1:2.0.33-1ubuntu1
ii adduser                             3.118ubuntu5
ii adwaita-icon-theme                  41.0-1ubuntu1
ii aisleriot                           1:3.22.22-1
ii alsa-base                           1.0.25+dfsg-0ubuntu7
ii alsa-topology-conf                  1.2.5.1-2
ii alsa-ucm-conf                       1.2.6.3-1ubuntu1
ii alsa-utils                          1.2.6-1ubuntu1
ii amd64-microcode                     3.20191218.1ubuntu2
ii anacron                             2.3-31ubuntu2
ii apg                                 2.2.3.dfsg.1-5build2
ii apparmor                            3.0.4-2ubuntu2
```

Hình 3: Minh họa Liệt kê danh sách những gói đã được cài đặt với thông tin chi tiết cho mỗi gói.

### Liệt kê thông tin chính xác các File của gói đã được chỉ định:

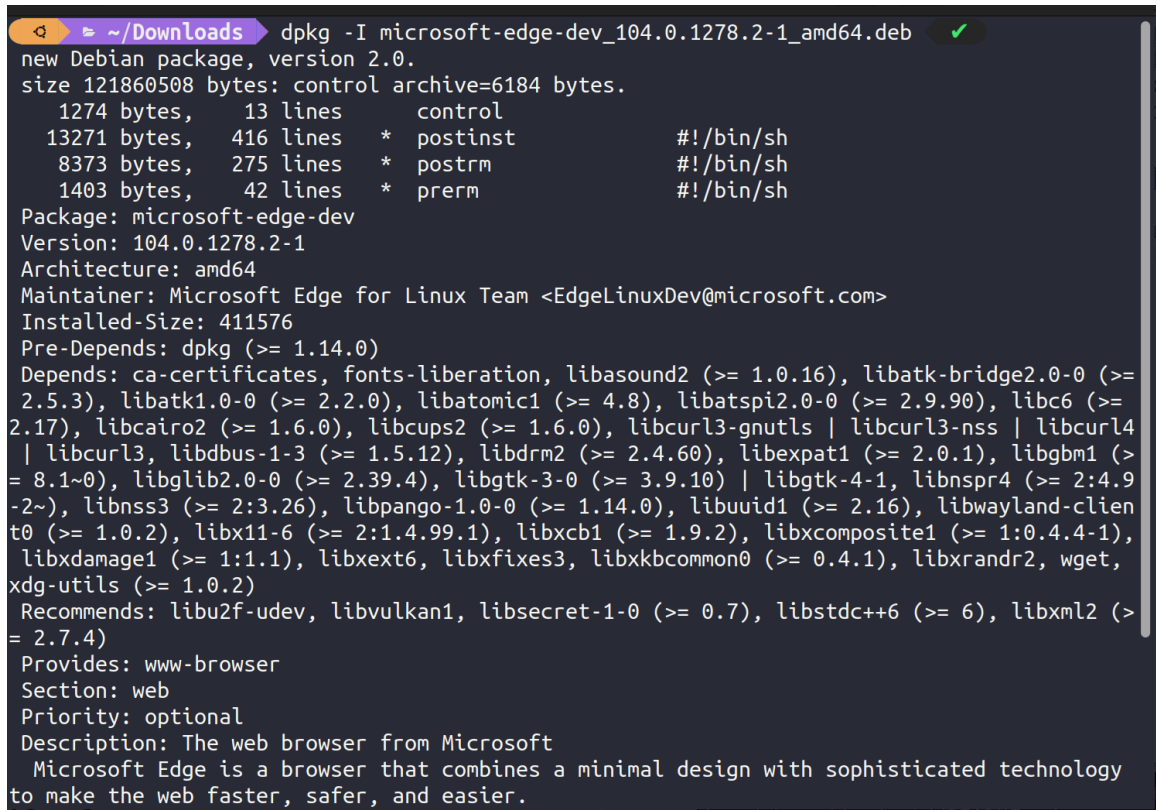
- root@localhost# rpm -qvp <package-name.rpm>
- root@localhost# dpkg -c <package-name.deb>

```
~/Downloads dpkg -c microsoft-edge-dev_104.0.1278.2-1_amd64.deb
drwxr-xr-x root/root      0 2022-06-07 07:32 ./
drwxr-xr-x root/root      0 2022-06-07 07:32 ./etc/
drwxr-xr-x root/root      0 2022-06-07 07:32 ./etc/cron.daily/
drwxr-xr-x root/root      0 2022-06-07 07:32 ./opt/
drwxr-xr-x root/root      0 2022-06-07 07:32 ./opt/microsoft/
drwxr-xr-x root/root      0 2022-06-07 07:32 ./opt/microsoft/msedge-dev/
drwxr-xr-x root/root      0 2022-06-07 07:32 ./opt/microsoft/msedge-dev/MEIPreload/
-rw-r--r-- root/root    228 2022-06-07 07:32 ./opt/microsoft/msedge-dev/MEIPreload/manif
est.json
-rw-r--r-- root/root   7682 2022-06-07 07:32 ./opt/microsoft/msedge-dev/MEIPreload/prelo
aded_data.pb
drwxr-xr-x root/root      0 2022-06-07 07:10 ./opt/microsoft/msedge-dev/WidevineCdm/
drwxr-xr-x root/root      0 2022-06-07 07:10 ./opt/microsoft/msedge-dev/WidevineCdm/_pla
tform_specific/
drwxr-xr-x root/root      0 2022-06-07 07:10 ./opt/microsoft/msedge-dev/WidevineCdm/_pla
tform_specific/linux_x64/
-rw-r--r-- root/root   9810712 2022-06-07 06:57 ./opt/microsoft/msedge-dev/WidevineCdm/_pla
tform_specific/linux_x64/libwidevinecdm.so
-rw-r--r-- root/root     721 2022-06-07 06:57 ./opt/microsoft/msedge-dev/WidevineCdm/mani
fest.json
drwxr-xr-x root/root      0 2022-06-07 07:32 ./opt/microsoft/msedge-dev/cron/
-rwxr-xr-x root/root    7923 2022-06-07 07:32 ./opt/microsoft/msedge-dev/cron/microsoft-e
dge-dev
-rw-r--r-- root/root     542 2022-06-07 07:32 ./opt/microsoft/msedge-dev/default-app-bloc
k
-rw-r--r-- root/root 12246928 2022-06-07 07:32 ./opt/microsoft/msedge-dev/icudtl.dat
```

Hình 4: Minh họa Liệt kê thông tin chính xác các File của gói đã được chỉ định.

### Hiển thị thông tin về một gói phần mềm:

- root@localhost# rpm -qpi <package-name.rpm>
- root@localhost# dpkg -I <package-name.deb>



```

~/Downloads dpkg -I microsoft-edge-dev_104.0.1278.2-1_amd64.deb
new Debian package, version 2.0.
size 121860508 bytes: control archive=6184 bytes.
 1274 bytes, 13 lines control
13271 bytes, 416 lines * postinst      #!/bin/sh
 8373 bytes, 275 lines * postrm       #!/bin/sh
 1403 bytes, 42 lines * prerm         #!/bin/sh
Package: microsoft-edge-dev
Version: 104.0.1278.2-1
Architecture: amd64
Maintainer: Microsoft Edge for Linux Team <EdgeLinuxDev@microsoft.com>
Installed-Size: 411576
Pre-Depends: dpkg (>= 1.14.0)
Depends: ca-certificates, fonts-liberation, libasound2 (>= 1.0.16), libatk-bridge2.0-0 (>= 2.5.3), libatk1.0-0 (>= 2.2.0), libatomic1 (>= 4.8), libatspi2.0-0 (>= 2.9.90), libc6 (>= 2.17), libcairo2 (>= 1.6.0), libcups2 (>= 1.6.0), libcurl3-gnutls | libcurl3-nss | libcurl4 | libcurl3, libdbus-1-3 (>= 1.5.12), libdrm2 (>= 2.4.60), libexpat1 (>= 2.0.1), libgbm1 (>= 8.1~0), libglib2.0-0 (>= 2.39.4), libgtk-3-0 (>= 3.9.10) | libgtk-4-1, libnspr4 (>= 2:4.9-2~), libnss3 (>= 2:3.26), libpango-1.0-0 (>= 1.14.0), libuuid1 (>= 2.16), libwayland-client0 (>= 1.0.2), libx11-6 (>= 2:1.4.99.1), libxcb1 (>= 1.9.2), libxcomposite1 (>= 1:0.4.4-1), libxdamage1 (>= 1:1.1), libxext6, libxf86vm0, libxkbcommon0 (>= 0.4.1), libxrandr2, wget, xdg-utils (>= 1.0.2)
Recommends: libu2f-udev, libvulkan1, libsecret-1-0 (>= 0.7), libstdc++6 (>= 6), libxml2 (>= 2.7.4)
Provides: www-browser
Section: web
Priority: optional
Description: The web browser from Microsoft
 Microsoft Edge is a browser that combines a minimal design with sophisticated technology to make the web faster, safer, and easier.

```

Hình 5: Minh họa Hiển thị thông tin về một gói phần mềm

### Kiểm tra tính toàn vẹn cho một gói phần mềm:

- root@localhost# rpm -Va
- root@localhost# debsums -a

```

/usr/share/icons/Yaru-blue/16x16/actions/go-first.png OK
/usr/share/icons/Yaru-blue/16x16/actions/go-last.png OK
/usr/share/icons/Yaru-blue/16x16/actions/mail-reply-all.png OK
/usr/share/icons/Yaru-blue/16x16/apps/filemanager-app.png OK
/usr/share/icons/Yaru-blue/16x16/apps/software-updater.png OK
/usr/share/icons/Yaru-blue/16x16/apps/tweaks-app.png OK
/usr/share/icons/Yaru-blue/16x16/emblems/emblem-symbolic-link.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-documents.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-download.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-dropbox.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-music.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-pictures.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-publicshare.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-remote.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-templates.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder-videos.png OK
/usr/share/icons/Yaru-blue/16x16/places/folder.png OK
/usr/share/icons/Yaru-blue/16x16/places/insync-folder.png OK
/usr/share/icons/Yaru-blue/16x16/places/preferences-desktop-wallpaper.png OK
/usr/share/icons/Yaru-blue/16x16/places/user-desktop.png OK
/usr/share/icons/Yaru-blue/16x16/places/user-home.png OK
/usr/share/icons/Yaru-blue/16x16/status/folder-open.png OK
/usr/share/icons/Yaru-blue/16x16@2x/actions/edit-select-all.png OK
/usr/share/icons/Yaru-blue/16x16@2x/actions/folder-new.png OK
/usr/share/icons/Yaru-blue/16x16@2x/actions/go-first.png OK
/usr/share/icons/Yaru-blue/16x16@2x/actions/go-last.png OK
/usr/share/icons/Yaru-blue/16x16@2x/actions/mail-reply-all.png OK
/usr/share/icons/Yaru-blue/16x16@2x/apps/filemanager-app.png OK
/usr/share/icons/Yaru-blue/16x16@2x/apps/software-updater.png OK
/usr/share/icons/Yaru-blue/16x16@2x/apps/tweaks-app.png OK
/usr/share/icons/Yaru-blue/16x16@2x/emblems/emblem-symbolic-link.png OK

```

Hình 6: Minh họa Kiểm tra tính toàn vẹn cho một gói phần mềm

### Cài đặt một gói phần mềm mới:

- root@localhost# rpm -Uvh <package-name.rpm>
- root@localhost# dpkg -i <package-name.deb>

```

~ /Downloads dpkg -i microsoft-edge-dev_104.0.1278.2-1_amd64.deb ✓
dpkg: error: requested operation requires superuser privilege
~ /Downloads sudo dpkg -i microsoft-edge-dev_104.0.1278.2-1_amd64.deb
[sudo] password for minhbien:
(Reading database ... 231173 files and directories currently installed.)
Preparing to unpack microsoft-edge-dev_104.0.1278.2-1_amd64.deb ...
Unpacking microsoft-edge-dev (104.0.1278.2-1) over (104.0.1271.2-1) ...
Setting up microsoft-edge-dev (104.0.1278.2-1) ...
Processing triggers for mailcap (3.70+nmu1ubuntu1) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu3) ...
Processing triggers for desktop-file-utils (0.26-1ubuntu3) ...
Processing triggers for man-db (2.10.2-1) ...
~ /Downloads

```

Hình 7: Minh họa Cài đặt một gói phần mềm mới

## 4.Sử dụng TCP Wrapper.

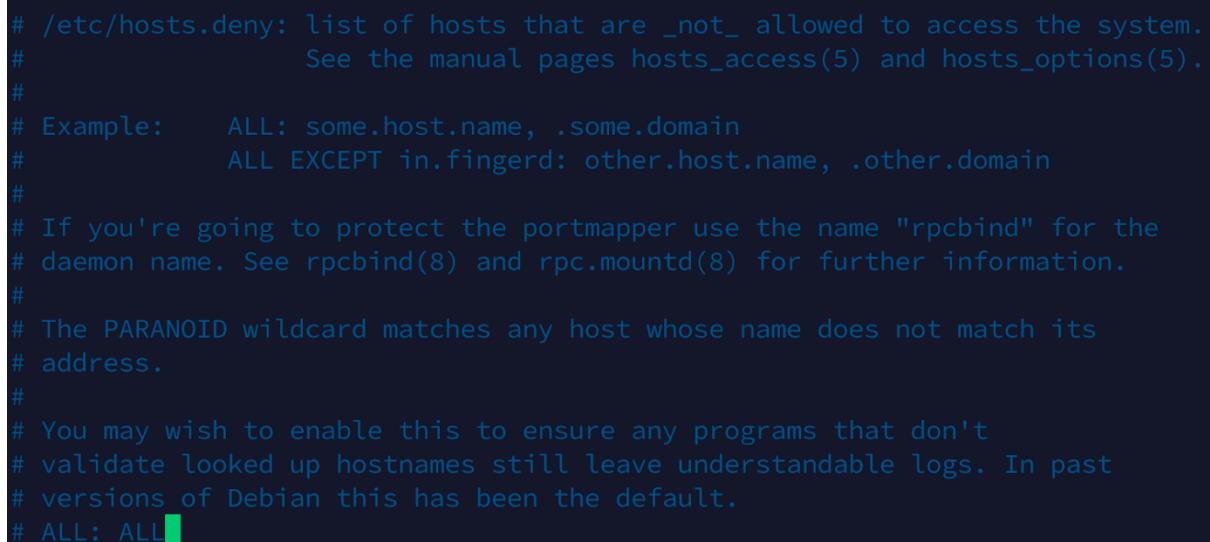
---

Đây là một phương pháp chặn truy cập các dịch vụ trên máy chủ Linux của bạn thông qua hạn chế IP. Bài viết này sẽ giúp bạn chặn truy cập SSH từ tất cả các IP ngoại trừ danh sách IP “được phép”. Cách thức này đạt được là thông qua hai tệp nằm trong thư mục / etc. Một tên là hosts.allow và các host.deny khác.

- /etc/hosts.allow : Tệp này chứa tên của các máy chủ được phép sử dụng các dịch vụ mạng.
  - /etc/hosts.deny : Tệp này chứa tên của máy chủ không thể sử dụng dịch vụ mạng.
- Cú pháp của các tệp này như sau:
- list\_of\_service : list\_of\_client [ : lệnh \_ shell ]
- Để bảo mật máy chủ Linux là chặn tất cả các kết nối đến và chỉ cho phép một vài máy chủ hoặc mạng cụ thể. Để làm như vậy, chỉnh sửa tệp /etc/hosts.deny :
- sudo vi /etc/hosts.deny

Thêm dòng sau. Dòng này từ chối kết nối với tất cả các dịch vụ và tất cả các mạng.

ALL: ALL



```
# /etc/hosts.deny: list of hosts that are _not_ allowed to access the system.
# See the manual pages hosts_access(5) and hosts_options(5).
#
# Example:      ALL: some.host.name, .some.domain
#              ALL EXCEPT in.fingerd: other.host.name, .other.domain
#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
#
# The PARANOID wildcard matches any host whose name does not match its
# address.
#
# You may wish to enable this to ensure any programs that don't
# validate looked up hostnames still leave understandable logs. In past
# versions of Debian this has been the default.
# ALL: ALL
```

Hình 8: Minh họa chặn tất cả các kết nối đến và chỉ cho phép một vài máy chủ hoặc mạng cụ thể

- Sau đó, chỉnh sửa tệp /etc/hosts.allow :
- sudo vi /etc/hosts.allow

Thêm địa chỉ mạng cụ thể muốn kết nối:

---

sshd : 192.168.184.140

```
# /etc/hosts.allow: list of hosts that are allowed to access the system.
# See the manual pages hosts_access(5) and hosts_options(5).
#
# Example:      ALL: LOCAL @some_netgroup
#              ALL: .foobar.edu EXCEPT terminalserver.foobar.edu
#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
# sshd: 192.168.184.140
```

Hình 9: Minh họa thêm địa chỉ mạng cụ thể muốn kết nối

Theo quy tắc trên, tất cả các kết nối đến sẽ bị từ chối cho tất cả các máy chủ ngoại trừ máy chủ 192.168.184.140

## 5. An toàn cho các giao dịch trên mạng

Có rất nhiều dịch vụ mạng truyền thông giao tiếp thông qua giao thức văn bản không mã hoá, như TELNET, FTP, RLOGIN, HTTP, POP3. Trong các giao dịch giữa người dùng với máy chủ, tất cả các thông tin dạng gói được truyền qua mạng dưới hình thức văn bản không được mã hoá. Việc giải mã các gói tin này rất dễ dàng, cho phép lấy được các thông tin như tên người dùng, mật khẩu và các thông tin quan trọng khác. Các kỹ thuật thông dụng hiện nay là IPSec, SSL, TLS, SASL và PKI. Quản trị từ xa là một tính năng hấp dẫn của các hệ thống UNIX. Người quản trị mạng có thể dễ dàng truy nhập vào hệ thống từ bất kỳ nơi nào trên mạng thông qua các giao thức thông dụng như telnet, rlogin. SSH.

---

## Chương IV: Cơ chế quản lý tài nguyên phân quyền

### 1. Quyền truy nhập thư mục và file

Mỗi file và thư mục trong Linux đều có một chủ sở hữu và một nhóm sở hữu, cũng như một tập hợp các quyền truy nhập. Cho phép thay đổi các quyền truy nhập và quyền sở hữu file và thư mục nhằm cung cấp truy nhập nhiều hơn hay ít hơn. Tính chất kiểm soát truy nhập hệ thống file Linux được thực hiện bằng cách sử dụng bộ dữ liệu, được bảo tồn riêng cho từng file này là dữ liệu chung gọi các chế độ truy nhập, hoặc gọi là các mod của file. Các chế độ là một phần của file, giữ lại những thông tin trong các hệ thống file đó. Mỗi file của chế độ kiểm soát truy cập có 3 lớp:

- User: những người dùng sở hữu file
- Group: Những nhóm sở hữu file
- Other: tất cả những người dùng trên hệ thống

Bản dưới đây liệt kê các kiểu file, thư mục trong Linux:

Chữ cái biểu diễn	Kiểu file, thư mục
d	Thư mục (directory)
b	File kiểu khối (block-type special file)
c	File kiểu kí tự (character-type special file)
l	Liên kết tượng trưng (symbolic link)
p	File đường ống (pipe)
s	Socket
-	File bình thường (regular file)

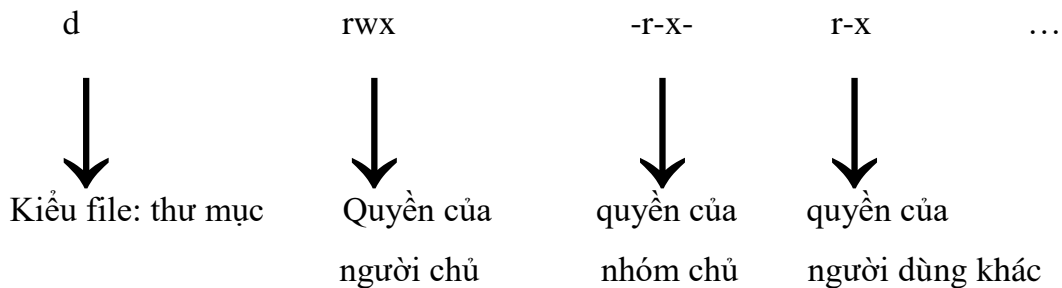


- Xem quyền thư mục với câu lệnh: `ls -l`

```
hungga@hungga ~$ ls -l
total 60
drwxrwxr-x  2 hungga hungga 4096 Thg 6   8 22:50 appImages
drwxrwxr-x 18 hungga hungga 4096 Thg 6  15 00:34 Code
-rw-rw-r--  1 hungga hungga  141 Thg 6   7 22:15 cv_debug.log
drwxr-xr-x  2 hungga hungga 4096 Thg 4  22 14:59 Desktop
drwxr-xr-x  3 hungga hungga 4096 Thg 6   8 22:55 Documents
drwxr-xr-x  3 hungga hungga 4096 Thg 6  15 20:54 Downloads
```

Hình 10: Minh họa quyền xem thư mục với câu lệnh: `ls-l`

9 kí tự tiếp theo trong chuỗi là quyền truy nhập được chia làm 3 nhóm tương ứng với quyền truy nhập của người sở hữu, nhóm sở hữu và người dùng khác.



Mỗi một file hay thư mục trong Linux đều có 3 quyền đọc, ghi, thực thi được xác định cho 3 chủ sở hữu ở trên.

- Đọc: Nếu là một file thì quyền này cho phép bạn mở file đó lên và đọc. Nếu là một thư mục thì nó cho phép bạn liệt kê danh sách file hay thư mục trong thư mục đó.
- Ghi: Quyền ghi cho phép bạn sửa đổi nội dung của file. Nếu là thư mục thì nó cho phép bạn có thể thêm, xóa và đổi tên các file trong thư mục đó.
- Thực thi: Với Windows bạn có thể chạy với một file có đuôi ".exe" một cách dễ dàng. Khác so với Windows, trong Linux bạn không thể chạy khi nó không được cấp quyền thực thi. Còn đối với thư mục thì bạn không thể truy cập(cd) nếu bạn không có quyền thực thi nó.

Như các chế độ, người sử dụng và quyền sở hữu tài sản của nhóm là một phần của Inode, và cả hai được chỉ định khi nào một file được tạo ra. Thông thường, chủ sở hữu là người tạo ra file này. Các file của nhóm này thường được thiết lập để



tạo sự mặc định của nhóm. Nhóm có quyền sở hữu cho biết quyền của các nhóm thành viên. Những người khác sử dụng là những người không phải thành viên các file của nhóm và không phải là file của người được trao quyền. Đối với mỗi lớp trong 3 lớp của người sử dụng, các chế độ truy cập xác định ba loại quyền khác nhau và áp dụng cho file và thư mục.

Quyền	Kí hiệu	Quyền với file	Quyền với thư mục
Đọc(read)	r	Kiểm tra nội dung của file	Danh sách nội dung thư mục
Ghi(write)	w	Ghi hoặc thay đổi file	Tạo, xóa file trong thư mục
Thực thi(execute)	x	Chạy file trong chương trình	Truy cập vào thư mục
Từ chối (Deny)	-	Không có quyền	Không có quyền

Các quyền này cũng có thể xác định bằng các con số tương ứng:

r (read) – được biểu diễn bằng số 4.

w (write) – được biểu diễn bằng số 2.

x (execute) – được biểu diễn bằng số 1.

– (Deny) – được biểu diễn bằng số 0

## 2. Chế độ truy nhập

Ba quyền đọc, ghi, thực thi cho phép áp dụng cho ba lớp khác nhau của người sử dụng: người sử dụng, nhóm và khác.

- SUID: Là thuộc tính cho các file thực thi duy nhất và không có hiệu lực trên các thư mục.

- SGID: Thuộc tính hoạt động theo cùng một cách như SUID cho các file thực thi, quá trình cài đặt nhóm chủ sở hữu vào file của nhóm.

- Sticky: Tại một thời gian, những bit, áp dụng cho các chương trình thực thi, cờ hệ thống, để giữ một hình ảnh của chương trình trong bộ nhớ sau khi hoàn tất chương trình đang chạy.

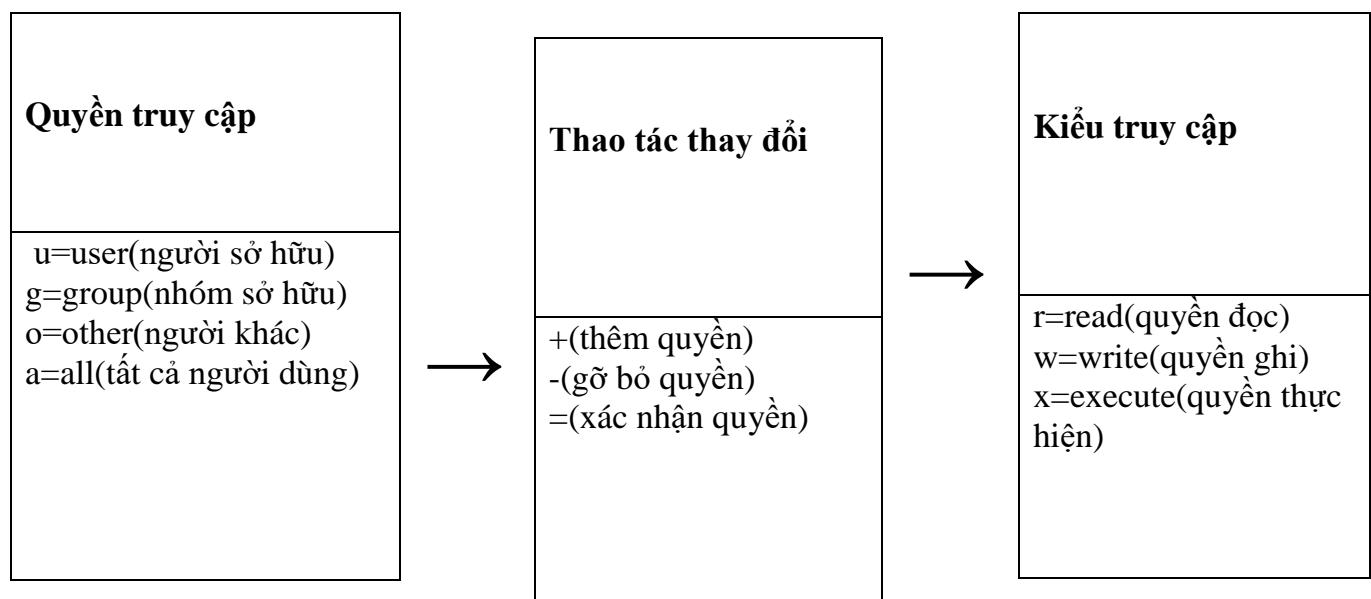
Tuy nhiên đối với thư mục thì chỉ có ba loại kí hiệu của các quyền truy nhập là: ---, r-x và rwx, vì nội dung của thư mục là danh sách của các file và thư mục con có

bên trong thư mục đó. Quyền đọc một thư mục là được xem nội dung của thư mục đó và quyền thực hiện đối với một thư mục là quyền tìm được file và thư mục con có trong thư mục.

Sự hạn chế trùng hợp về quyền truy nhập thư mục được giải thích. Giả sử chỉ có quyền đọc trên thư mục, khi đó sẽ xem được những file hay thư mục nào trong thư mục nhưng lại không thể xem nội dung cụ thể của file hay thư mục có trên thư mục đó vì không tìm được nó. Hoặc giả sử có quyền thực hiện – quyền này sẽ cho phép tìm được file có trên thư mục – nhưng lại không có quyền đọc với một thư mục, vậy thì khó có thể biết được trong thư mục có những file nào.

## 2.1. Cách xác lập tương đối

Cách xác lập tương đối là dễ nhớ theo ý nghĩa của nội dung các mod và chỉ những thay đổi thực sự mới biểu diễn trong lệnh. Hình sau đây sẽ mô tả:



Các chữ cái biểu diễn mod theo cách xác lập tương đối

Có thể kết hợp các mục từ hộp thứ nhất và hộp thứ ba với một mục từ hộp thứ hai để tạo ra một mod.

## 2.2. Cách xác lập tuyệt đối

Đối với người dùng hiểu sơ bộ về biểu diễn số trong hệ cơ số 8 thì cách xác lập tuy dễ dàng lại được ưa chuộng hơn.

Biểu diễn quyền truy nhập file thông qua dãy gồm 9 vị trí dưới dạng `rw-rw-rw-`, trong đó từng cụm 3 vị trí theo thứ tự tương ứng: với chủ sở hữu, nhóm sở hữu và người dùng khác. Thuộc tính quyền truy nhập của một số file có thể biểu diễn thành 9 bit nhị phân trong đó bit có giá trị 1 thì quyền đó được xác định, ngược lại thì quyền đó bị tháo bỏ. Như vậy chủ sở hữu tương ứng với 3 bit đầu tiên, nhóm sở hữu tương ứng với 3 bit giữa, người dùng khác tương ứng với 3 bit cuối.

Mỗi cụm 3 bit như vậy cho một chữ số hệ 8 (nhận giá trị từ 0 đến 7) và thuộc tính quyền truy nhập tương ứng với 3 chữ số hệ 8.

Đặt thuộc tính quyền truy nhập đối với file `memollà` `rw-r-xr-x`.

Để dễ xác lập 3 chữ số:

Quyền	Chữ số hệ 8	Quyền	Chữ số hệ 8
Chỉ đọc	4	Chỉ đọc và ghi	6
Chỉ ghi	2	Chỉ đọc và thực hiện	5
Chỉ thực hiện	1	Chỉ ghi và thực hiện	3
Không có quyền nào	0	Đọc ghi và thực hiện	7

Bảng 4: Hệ 8 áp dụng cách tính

Thiết lập Truy cập Modes

Khi tạo nhiều file có cùng một thuộc tính thì gán cho các file là mặc định.

Umask thiết lập một môi trường mà trong đó tác động đến quyền kiểm soát các file mới tạo. Giải thích Umask như sau: giá trị `mask` (mặt nạ) được thiết lập từ `umask` khi `mask` được thiết lập thì thông số này tác động đến quyền truy cập mặc định của file thư mục.

Khi được cung cấp với một số nguyên, `umask` đặt giá trị cho hệ vô gọi là giá trị `mask`.

Quyền truy cập cho phép = “quyền truy cập mặc định” and (not(`Umask`)).

Dạng thức	Áp dụng với file	Áp dụng với thư mục
Kí hiệu	Rw-rw-rw	Rwxrwxrwx
Nhị phân	110110110	111111111
Hệ 8	6 6 6	7 7 7

Bảng 5: Chế độ truy cập ban đầu

### 3. Một số lệnh thay đổi chế độ truy cập

#### 3.1. Thay đổi quyền sở hữu file với lệnh chown

Để thay đổi quyền sở hữu đối với một file. Nếu chỉ có tham số về chủ, thì người dùng sẽ có quyền sở hữu file và nhóm sở hữu không thay đổi. Nếu theo sau tên người chủ là dấu “.” Và tên của một nhóm thì nhóm đó sẽ sở hữu file. Nếu chỉ có dấu “.” và nhóm mà không có tên người chủ thì chỉ có quyền sở hữu nhóm của file thay đổi, lúc này, lệnh chown có tác dụng giống như lệnh chgrp.

Các tùy chọn của lệnh chown:

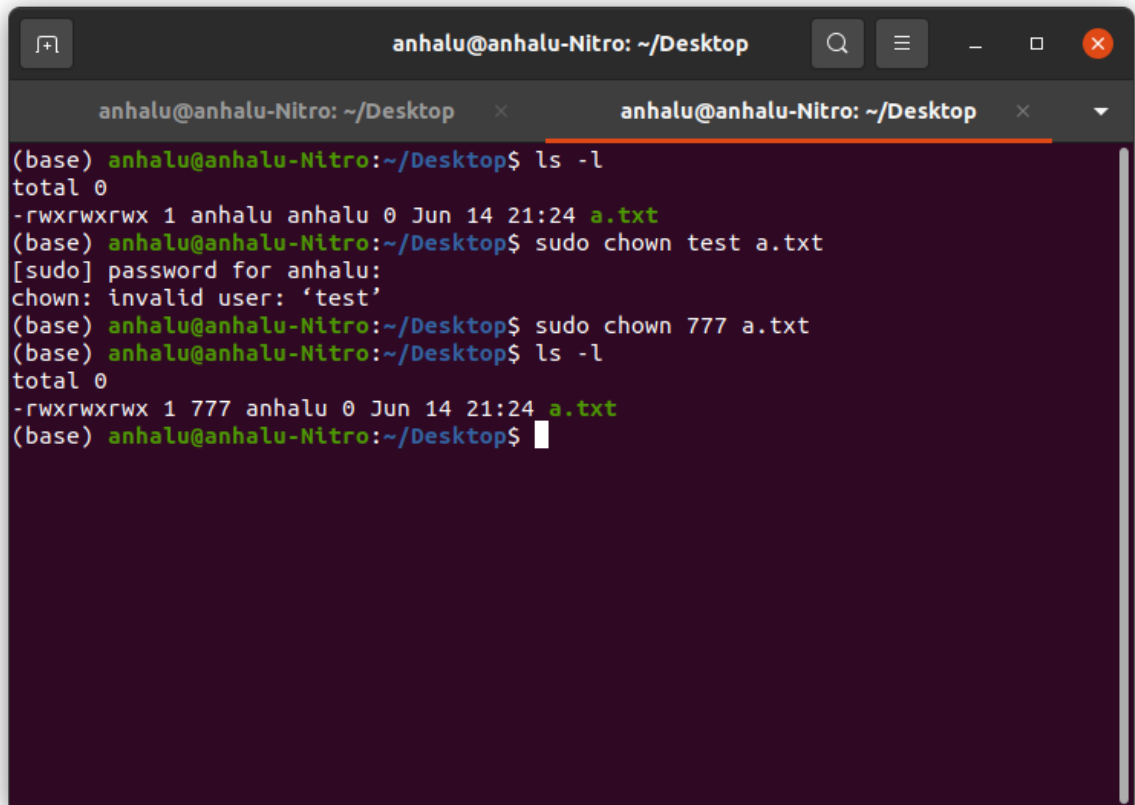
-c, --changes: hiển thị dòng thông báo chỉ với các file mà lệnh làm thay đổi sở hữu (số thông báo hiện ra có thể ít hơn trường hợp -v, -verbose).

-f, --silent, --quiet: bỏ qua hầu hết các thông báo lỗi.

-R, --recursive: thực hiện đổi quyền sở hữu đối với thư mục và file theo đệ quy.

-v, --verbose: hiển thị dòng thông báo với mọi file liên quan mà chown tác động tới (có hoặc không thay đổi sở hữu).

--help: đưa ra trang trợ giúp và thoát.



```
(base) anhalu@anhalu-Nitro: ~/Desktop$ ls -l
total 0
-rwxrwxrwx 1 anhalu anhalu 0 Jun 14 21:24 a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$ sudo chown test a.txt
[sudo] password for anhalu:
chown: invalid user: 'test'
(base) anhalu@anhalu-Nitro:~/Desktop$ sudo chown 777 a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$ ls -l
total 0
-rwxrwxrwx 1 777 anhalu 0 Jun 14 21:24 a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$
```

Hình 11: Minh họa thay đổi quyền sở hữu file với lệnh chown

### 3.2. Thay đổi quyền truy nhập file với lệnh chmod

Chế độ truy nhập có thể được thay đổi với chmod lệnh, tương đương chế độ truy cập tượng trưng hoặc truy cập vào chế độ kỹ thuật chi tiết.

Pemissi ons	r	Read pemission
	w	Write pemission
	x	Excute pemission
	X	Excute pemission for directories and files with excute pemission, but not plain files
	s	SUID or SGID pemissions
	t	Sticky bit

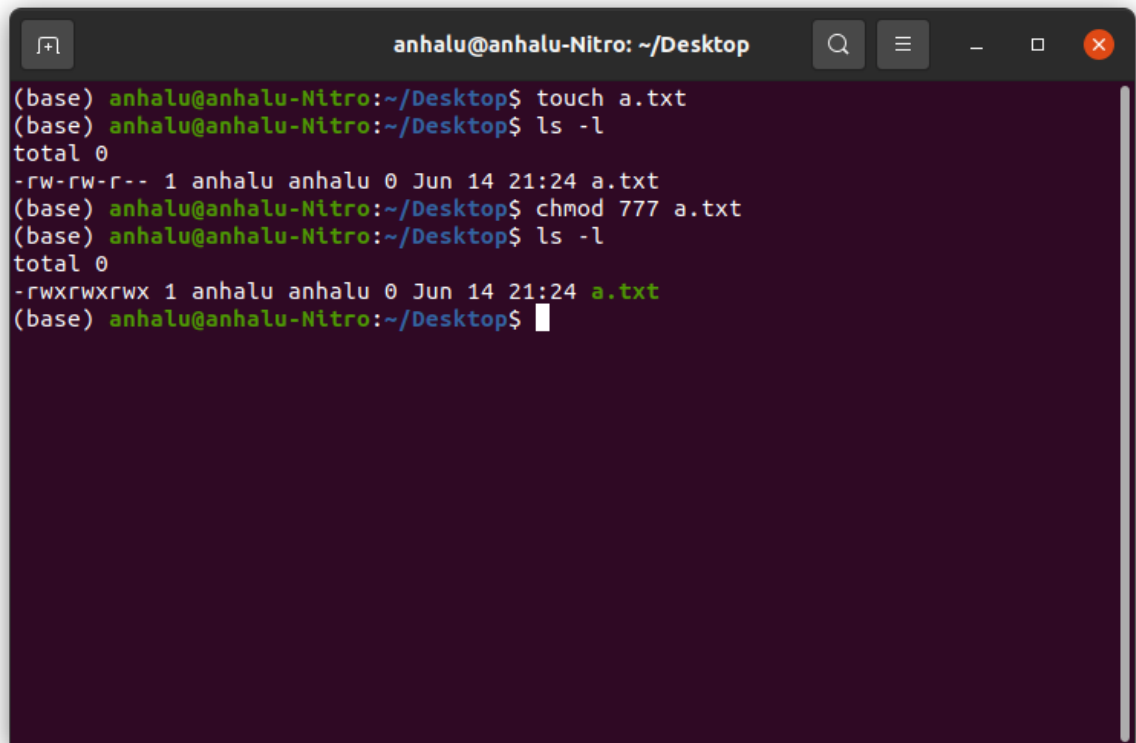
Bảng 6: Ký tự đặc biệt đối với lệnh chmod

Cú pháp lệnh chmod có ba dạng:

Chmod [tùy chọn] <mod [,mod]...><file...>

Chmod [tùy chọn] <mod-hệ 8><file...>

Chmod [tùy chọn] --reference=nhómR <file...>

A screenshot of a terminal window titled 'anhalu@anhalu-Nitro: ~/Desktop'. The terminal shows the following commands and output:

```
(base) anhalu@anhalu-Nitro:~/Desktop$ touch a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$ ls -l
total 0
-rw-rw-r-- 1 anhalu anhalu 0 Jun 14 21:24 a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$ chmod 777 a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$ ls -l
total 0
-rwxrwxrwx 1 anhalu anhalu 0 Jun 14 21:24 a.txt
(base) anhalu@anhalu-Nitro:~/Desktop$
```

Hình 12: Minh họa thay đổi quyền truy nhập file với lệnh chmod

Lệnh chmod cho phép xác lập quyền truy nhập theo kiểu (mode) trên file. Dạng đầu tiên là dạng xác lập tương đối, dạng thứ hai là dạng xác lập tuyệt đối và dạng cuối cùng là dạng gián tiếp chỉ dẫn theo quyền truy cập của file nhómR. Các tùy chọn của lệnh chmod được liệt kê như dưới đây và có nghĩa tương tự các tùy chọn tương ứng của các lệnh chown, chgrp:

-c,--changes

-f,--silent,--quiet

-v,--verbose

-R,—recursive

--help và tham số --reference=RFILE cũng có ý nghĩa gián tiếp như trong lệnh chgrp.

---

## Kết luận

Với đề tài : Nghiên cứu tìm hiểu hệ thống bảo vệ trong hệ điều hành Linux.

Chúng em tìm hiểu về 4 phần lớn đó là :

**Phần I** : An toàn hệ thống

**Phần II** : Bảo vệ hệ thống

**Phần III** : Các bước ban đầu để thiết lập một hệ thống bảo vệ trong Linux

**Phần IV** : Cơ chế quản lí tài nguyên phân quyền

Tổng kết lại các nội dung được trình bày trong tài liệu này, ta đã nắm được hệ thống bảo vệ trong hệ điều hành Linux. Từ đó, chúng ta có thể hiểu được một phần cách vận hành, hoạt động hệ thống bảo vệ của hệ điều hành. Tuy rằng tài liệu chỉ mang đến một phần kiến thức tuy nhỏ nhưng với chúng em thì cũng khá đầy đủ và căn bản để có thể tiếp tục nghiên cứu sâu hơn về hệ thống bảo vệ trong hệ điều hành Linux. Những nội dung chính như các vấn đề an toàn, cơ chế an toàn hệ thống, mục tiêu của bảo vệ hệ thống, cách hoạt động của miền bảo vệ, tổng quan và cơ chế hoạt động của virus, các bước ban đầu thiết lập hệ thống bảo vệ trong Linux, cơ chế quản lí tài nguyên phân quyền đều được trình bày trong tài liệu. Thông qua phần bài tập lớn này chúng em đã thấy được tầm quan trọng của hệ thống bảo vệ bên trong một hệ thống.

Qua đây chúng em xin gửi lời cảm ơn thầy đã tận tình giúp đỡ, hướng dẫn chúng em hoàn thành đề tài này. Tuy nhiên do trình độ và kiến thức của chúng em còn hạn chế nên không tránh khỏi những thiếu sót, chúng em rất mong nhận được những góp ý và bổ sung của thầy cô và các bạn để đề tài của chúng em được hoàn thiện hơn.

---

## **Tài liệu tham khảo**

- [1] Nguyễn Thanh Hải - Giáo Trình Nguyên Lý Hệ Điều Hành Đại học Công nghiệp Hà Nội, khoa công nghệ thông tin, Hà Nội 2016.
- [2] Nguyễn Kim Tuấn - Giáo Trình Lý Thuyết Hệ Điều Hành Đại học Huế, Trường Đại học khoa học, khoa công nghệ thông tin, Huế 06/2004.
- [3] Trần Hồ Thủy Tiên - Giáo Trình Nguyên Lý Hệ Điều Hành Đại học Đà Nẵng, trường Đại học Bách Khoa, Khoa Công Nghệ Thông Tin 01/04/2010.
- [4] Abraham Silberschatz, Galvin, Gagne, Operating System Concepts 8th edition. (tài liệu tham khảo từ nguồn nước ngoài).
- [5] Processes - IBM Documentation
- [6] Tài liệu trên internet