

# BÀI THỰC HÀNH SỐ 2

## TRIỂN KHAI HỆ THỐNG TƯỜNG LỬA

### 1. MỤC ĐÍCH VÀ NỘI DUNG

#### Mục đích

- Sinh viên hiểu được vai trò của hệ thống tường lửa trong các giải pháp phòng chống tấn công mạng
- Làm quen với một số hệ thống tường lửa cơ bản (Cisco ASA 5500, iptables)
- Rèn luyện kỹ năng cấu hình tường lửa theo yêu cầu

#### Môi trường thực hành

- Hệ điều hành: Windows/Linux
- Phần mềm: GNS3, Cisco IOS, Virtualbox, Wireshark

#### Yêu cầu chuẩn bị:

- Nắm vững kiến trúc và các mô hình triển khai tường lửa bộ lọc gói
- Cài đặt và tìm hiểu sử dụng các công cụ trong bài thực hành: GNS3, Virtualbox, Wireshark, iptables
- Môi trường thực hành: Windows và Linux
- **Đọc tài liệu thực hành. Hoàn thành các yêu cầu luyện tập tại nhà trong mục số 3 trước khi đến thực hành.**

### 2. GIỚI THIỆU MỘT SỐ SẢN PHẨM TƯỜNG LỬA

#### 2.1. Cisco ACL

Danh sách điều khiển truy cập (ACL – Access Control List) là giải pháp lọc gói cơ bản được cung cấp trên hầu hết các thiết bị của Cisco. Danh sách này sẽ chỉ ra các gói tin được phép chuyển tiếp hoặc bị loại bỏ dựa trên các thông tin trên phần tiêu đề, hoặc thông tin của phiên làm việc. Mỗi danh sách có thể có nhiều luật. Khi so khớp, các luật sẽ được sử dụng theo thứ tự ưu tiên từ trên xuống dưới. Gói tin phù hợp với luật nào sẽ được xử lý theo chỉ thị tại luật đó và bỏ qua tất cả các luật còn lại. Có 2 loại ACL chính được sử dụng:

- Standard ACLs: lọc gói theo địa chỉ trên gói tin
- Extended ACLs: lọc gói theo giao thức, địa chỉ, số hiệu cổng

Ngoài ra, Cisco cung cấp các Extended ACL đặc biệt sau:

- Reflexive ACLs: lọc gói theo trạng thái của liên kết
- Dynamic ACLs: ACL yêu cầu xác thực

Để hỗ trợ hoạt động của ACL, Cisco đưa ra một số cơ chế bổ sung tùy theo phiên bản IOS:

- Context-Based Access Control
- Turbo ACLs
- Distributed Time-Based ACLs
- Receive ACLs
- Infrastructure Protection ACLs
- Transit ACLs

### 2.1.1. Standard ACLs

Standard ACLs có thể định nghĩa bằng số hiệu hoặc bằng tên

- Định nghĩa bằng số hiệu:

```
(config)# access-list ACL-number {permit|deny} {host/source source-wildcard|any}
```

- Định nghĩa bằng tên

```
(config)# ip access-list standard name
```

```
(config-std-nacl)# {permit|deny} {host/source source-wildcard|any}
```

ACL-number	Từ 0 đến 99, từ 1300 đến 1699
name	Xâu không chứa dấu cách trắng(có phân biệt chữ hoa, chữ thường)
<b>deny   permit</b>	Chỉ thị xử lý ( <b>deny</b> : loại bỏ, <b>permit</b> : cho phép)
source	Địa chỉ IP tham chiếu
source-wildcard	Mặt nạ kiểm tra
<b>host</b>	Áp dụng luật với 1 địa chỉ IP chỉ định
<b>any</b>	Áp dụng luật với mọi địa chỉ IP

Mặt nạ kiểm tra: được sử dụng để kiểm tra sự so khớp giữa địa chỉ IP nguồn của gói tin và địa chỉ IP tham chiếu:

- Bit 0 trong mặt nạ: kiểm tra bit trong địa chỉ IP có vị trí tương ứng
- Bit 1 trong mặt nạ: không kiểm tra bit trong địa chỉ IP có vị trí tương ứng

Ví dụ:

128	64	32	16	8	4	2	1	Vị trí các bit trong byte và giá trị địa chỉ của nó
0	0	0	0	0	0	0	0	Mặt nạ kiểm tra tất cả các bit địa chỉ
0	0	1	1	1	1	1	1	Mặt nạ không kiểm tra 6 bits cuối cùng của địa chỉ
0	0	0	0	1	1	1	1	Mặt nạ không kiểm tra 4 bits cuối cùng của địa chỉ
1	1	1	1	1	1	0	0	Mặt nạ kiểm tra 2 bits cuối cùng của địa chỉ
1	1	1	1	1	1	1	1	Mặt nạ không kiểm tra địa chỉ

Ví dụ, để kiểm tra địa chỉ IP nguồn của gói tin có nằm trong mạng 192.168.1.32/28. Để thấy mạng 192.168.1.32 /28 cung cấp các địa chỉ IP có biểu diễn nhị phân như sau(màu đỏ là các bit NetworkID):

1100 0000 1010 1000 0000 0001 0010 0000 đến

1100 0000 1010 1000 0000 0001 0010 1111

Do đó, mặt nạ kiểm tra sẽ yêu cầu kiểm tra giá trị của 28 bit đầu tiên phải so khớp với chuỗi bit NetworkID là 1100 0000 1010 1000 0000 0001 0010

Vì vậy ta có:

- Địa chỉ IP tham chiếu: 192.168.1.32 (1100 0000 1010 1000 0000 0001 0010 0000)
- Mặt nạ kiểm tra: 0.0.0.15 (0000 0000 0000 0000 0000 0000 0000 1111)

### 2.1.2. Extended ACLs

Standard ACLs có thể định nghĩa bằng số hiệu hoặc bằng tên:

- Định nghĩa bằng số hiệu:

```
(config)# access-list ACL-number {deny | permit} protocol {host/source source-wildcard|any} [operator src-port] {host/destination destination-wildcard|any}[operator dst-port] [established]
```

- Định nghĩa bằng tên:

```
(config)# ip access-list extended name
```

```
(config-ext-nacl)# {deny | permit} protocol {host/source source-wildcard|any}
[operator src-port] {host/destination destination-wildcard|any}[operator dst-port]
[established]
```

ACL-number	Từ 100 đến 199, từ 2000 đến 2699
name	Xâu không chứa dấu cách trắng(có phân biệt chữ hoa, chữ thường)
<b>deny   permit</b>	Chỉ thị xử lý ( <b>deny</b> : loại bỏ, <b>permit</b> : cho phép)
source	Địa chỉ IP tham chiếu cho địa chỉ nguồn
source-wildcard	Mặt nạ kiểm tra địa chỉ nguồn
destination	Địa chỉ IP tham chiếu cho địa chỉ đích
destination-wildcard	Mặt nạ kiểm tra địa chỉ đích
<b>host</b>	Áp dụng luật với 1 địa chỉ IP chỉ định
<b>any</b>	Áp dụng luật với mọi địa chỉ IP
operator	Toán tử kiểm tra số hiệu cổng: <b>lt</b> (nhỏ hơn), <b>gr</b> (lớn hơn), <b>eq</b> (bằng), <b>neq</b> (khác), <b>range</b> (nằm trong khoảng)
src-port	Giá trị số hiệu cổng nguồn được so khớp
dst-port	Giá trị số hiệu cổng đích được so khớp
<b>established</b>	Áp dụng luật với các gói tin trên liên kết đã được thiết lập, chỉ sử dụng với giao thức TCP

### 2.1.3. Reflexive ACLs

Cách thức định nghĩa ACL ở trên sẽ cho phép thiết bị lọc gói theo dạng stateless. Trên thực tế, có nhiều trường hợp việc lọc gói dạng stateful yêu cầu thêm yếu tố là trạng thái của phiên. Reflexive ACL là dạng ACL cho phép lọc gói theo yêu cầu như vậy. Để sử dụng Reflexive ACLs cần định nghĩa 2 Extended ACL theo tên:

- Một Reflexive ACL định nghĩa các luật cho phép (permit) luồng dữ liệu inbound/outbound(đi vào/đi ra) trên một cổng mạng
- Một Nested ACL để chứa các luật tạm thời được sinh ra từ Reflexive ACL. Cứ mỗi một luồng dữ liệu được Reflexive ACL cho phép đi qua, có một luật được sinh ra trên Nested ACL cho phép luồng dữ liệu theo chiều ngược lại trên cổng mạng đó.

Các bước cấu hình Reflexive ACLs như sau:

- Bước 1: Định nghĩa Reflexive ACL đối với luồng dữ liệu inbound/outbound

```
(config)# ip access-list extended ref-ACL-name
```

```
(config-ext-nacl)# permit protocol {host/source source-wildcard|any} [operator src-port] {host/destination destination-wildcard|any}[operator dst-port] reflect reflect-name [timeout sec]
```

- Bước 2: Định nghĩa Nested ACL để nhúng các luật của Reflexive ACL

```
(config)# ip access-list extended nest-ACL-name
```

```
(config-ext-nacl)# evaluate reflect-name
```

Lưu ý: Trên Reflexive ACL và Nested ACL vẫn có thể định nghĩa các luật thông thường khác như Extended ACL

#### 2.1.4. Áp đặt ACL trên giao tiếp mạng

Quy tắc: Chỉ áp đặt được 1 ACL cho luồng inbound và 1 ACL cho luồng outbound trên mỗi cổng giao tiếp mạng

```
(config-if)#ip access-group {ACL-name / ACL-number} {in | out}
```

Để xem danh sách các ACL đã được định nghĩa:

```
#show access-list [ACL-name / ACL-number]
```

Lưu ý:

- Nếu có thể, nên áp đặt lên luồng inbound để bỏ qua quá trình chuyển tiếp dữ liệu nếu gói tin bị cấm. Điều này giúp cho hiệu năng hoạt động của thiết bị tốt hơn
- ACL áp dụng cho luồng inbound kiểm tra các gói tin trước khi thực hiện chuyển đổi địa chỉ bởi NAT nếu có
- ACL áp dụng cho luồng outbound kiểm tra các gói tin sau khi thực hiện chuyển đổi địa chỉ bởi NAT nếu có

## 2.2. Cisco ASA 55xx

Cisco ASA 55xx là dòng sản phẩm thiết bị tường lửa chuyên dụng, cung cấp các tính năng phong phú, cho phép kiểm soát các luồng dữ liệu tốt hơn so với giải pháp ACL. Cisco ASA sử dụng bộ lọc gói kiểm soát trạng thái (stateful inspector) dựa trên mức an ninh của các cổng giao tiếp mạng. Cụ thể như sau:

- Luồng dữ liệu chuyển tiếp từ cổng giao tiếp có **mức an ninh cao** sang cổng giao tiếp có **mức an ninh thấp**: mặc định cho phép. Có thể sử dụng ACL bổ sung để định nghĩa các luồng bị cấm.
- Luồng dữ liệu chuyển tiếp từ cổng giao tiếp có **mức an ninh thấp** sang cổng giao tiếp có **mức an ninh cao**: mặc định cấm. Có thể sử dụng ACL bổ sung để định nghĩa các luồng được phép.
- Luồng dữ liệu chuyển tiếp giữa các cổng giao tiếp có **mức an ninh như nhau**: mặc định cấm. Có thể chuyển sang chế độ mặc định cho phép hoặc sử dụng ACL bổ sung.

### 2.2.1. Cấu hình cổng giao tiếp mạng

```
Asa(config)# interface interface-ID
Asa(config-if)# ip address ip-addr netmask
Asa(config-if)# security-level level
Asa(config-if)# nameif interface-name
Asa(config-if)# no shutdown
```

Mặc định, luồng dữ liệu chuyển tiếp giữa các cổng giao tiếp có **mức an ninh như nhau** bị cấm. Để cho phép các luồng này, sử dụng câu lệnh sau:

```
Asa(config)# same-security-traffic permit inter-interface
Asa(config)# same-security-traffic permit intra-interface
```

### 2.2.2. Cấu hình ACL

ACL được sử dụng trên ASA với mục đích tương tự như trên các thiết bị khác, nhưng có một số khác biệt trong cách định nghĩa.

#### Cấu hình Standard ACLs

```
Asa(config)# access-list [line line] ACL-name standard {deny | permit}
                {any | host host-address | network-address mask}
```

#### Cấu hình Extended ACLs

Asa(config)# **access-list** [*line line*] *ACL-name* **extended** {*deny* | *permit*} *protocol* *source-address-args* *dest-address-args*

ACL-name	Xâu không chứa dấu cách trắng(có phân biệt chữ hoa, chữ thường)
<b>deny</b>   <b>permit</b>	Chỉ thị xử lý ( <b>deny</b> : loại bỏ, <b>permit</b> : cho phép)
protocol	Giao thức
source-address-args	Tham số so khớp nguồn của lưu lượng
dest-address-args	Tham số so khớp đích của lưu lượng
operator	Toán tử kiểm tra số hiệu cổng: <b>lt</b> (nhỏ hơn), <b>gr</b> (lớn hơn), <b>eq</b> (bằng), <b>neq</b> (khác), <b>range</b> (nằm trong khoảng)

Tham số để so khớp nguồn và đích có thể sử dụng kết hợp các thông tin sau:

- **host** *host-address*: Địa chỉ IP cụ thể của 1 nút mạng
- *net-address netmask*: Địa chỉ IP của một mạng và mặt nạ
- *any*: địa chỉ IP bất kỳ
- *operator port*: Số hiệu cổng với các toán tử kiểm tra operator là **eq**, **neq**, **lt**, **gt**, **range**
- **object** *object-name*, **object-group** *group-name*: các đối tượng mạng định nghĩa nút mạng/nhóm nút mạng, dịch vụ/nhóm dịch vụ

Định nghĩa các đối tượng mạng:

- Nút mạng: định nghĩa đối tượng

Asa(config)# **object network** *object-name*

Asa (config-network-object) # { **host** *host-address* |

**subnet** *net-address netmask*

**range** *start-address end-address* }

- Nhóm nút mạng: định nghĩa một nhóm đối tượng

Asa(config)# **object-group network** *group-name*

Các lệnh sau là tùy chọn và có thể lặp lại với số lần tùy ý

Asa (config-network-object-group)# **network-object host** *host-address*

Asa(config-network-object-group)# **network-object net-address** *netmask*

Asa(config-network-object-group)# **network-object object** *exist-object*

Asa(config-network-object-group)# **group-object** *exist-group*

- Dịch vụ: định nghĩa 1 dịch vụ

Asa(config)# **object service** *service-name*

Asa (config-service-object)# { **service protocol** |

**service icmp** [*icmp-type*] [*icmp-code*] |

**service [tcp | udp]** [**source operator port**]

[**destination operator port**]}

- Nhóm dịch vụ:

Asa(config)# **object-group service** *group-name*

Các lệnh sau là tùy chọn và có thể lặp lại với số lần tùy ý

Asa(config-service-object-group)# **service-object protocol**

Asa(config-service-object-group)# **service-object icmp** [*icmp-type*] [*icmp-code*]

Asa(config-service-object-group)# **service-object [tcp | udp]**

[**source operator port**][**destination operator port**]}

Asa(config-service-object-group)# **service-object object** *exist-object*

Asa(config-service-object-group)# **group-object** *exist-group*

### Cấu hình Webtype ACLs

Webtype ACLs là dạng ACL để kiểm soát lưu lượng Web mức độ đơn giản dựa trên địa chỉ URL

Asa(config)# **access-list [line line] ACL-name webtype {deny | permit}**

**url** [*url-string* | **any**]

*url-string*: Địa chỉ URL, trong đó có thể sử dụng ký tự đại diện

\*: Đại diện cho một chuỗi bất kỳ

?: Đại diện cho 1 ký tự bất kỳ

[]: Đại diện cho dãy ký tự



### Áp đặt ACL lên cổng giao tiếp mạng

Asa(config)#**access-group** *ACL-name* {**in** | **out**} **interface** *interface-name*

Lưu ý: Khi áp đặt lên luồng nào trên cổng, luồng đó sẽ được điều khiển bởi ACL thay cho cơ chế điều khiển bởi mức an ninh.

### 2.2.3. Giám sát lưu lượng tầng ứng dụng

ASA cho phép thực hiện các kiểm soát sâu trên các lưu lượng tầng ứng dụng. Các bước thực hiện cấu hình giám sát lưu lượng tầng ứng dụng như sau:

- Bước 1: Tạo class-map xác định đặc điểm tại tầng mạng, giao vận(L3/L4) của lưu lượng

Asa(config)# **class-map** *l4-class-name*

Asa(config-cmap)# **match access-list** *ACL-name*

Asa(config-cmap)# **match port** {**tcp** | **udp**} {**eq** *port-num* | **range** *start-port end-port*}

Asa(config-cmap)# **match default-inspection-traffic**

- Bước 2: Tạo class-map định nghĩa đặc điểm tại tầng ứng dụng của lưu lượng

Asa(config)# **class-map type inspect** *protocol app-class-name*

- Bước 3: Tạo policy-map định nghĩa chính sách xử lý lưu lượng ứng dụng

Asa(config)# **policy-map type inspect** *protocol protocol-policy-name*

Asa(config-pmap)# **class** *app-class-name*

Asa(config-pmap-c)# {**drop** | **drop-connection** | **reset** | **log**}

- Bước 4: Tạo policy-map định nghĩa cách thức xử lý lưu lượng phù hợp với class-map

Asa(config)# **policy-map** *policy-map-name*

Asa(config-pmap)# **class** *l4-class-name*

Asa(config-pmap-c)# **inspect** *protocol [protocol-map-name]*

- Bước 5: Áp đặt policy-map

Asa(config)# **service-policy** *policy-map-name* {**global** | **interface** *interface-name*}

### 2.3.2.1. Định nghĩa regex

ASA cho phép giám sát nội dung lưu lượng dựa trên việc so khớp với biểu thức chính quy regex (Xem thêm quy tắc viết regex ở phần phụ lục cuối tài liệu). Regex có thể được định nghĩa đơn lẻ hoặc sử dụng class-map để định nghĩa tập nhiều regex.

Trước khi định nghĩa regex, có thể sử dụng lệnh sau để kiểm tra regex đã được định nghĩa đúng theo ý muốn (lệnh này có thể sử dụng ở mọi chế độ cấu hình):

```
#test regex input_text regular_expression
```

*Input\_text*: Nội dung cần kiểm tra

*Regular\_expression*: Biểu thức regex sẽ sử dụng

- Định nghĩa regex:

```
Asa(config)# regex regex_name regular_expression
```

*name*: Tên regex

*regular\_expression*: Biểu thức regex (có tối đa 100 ký tự)

- Định nghĩa class-map:

```
Asa(config)# class-map type regex match-any class_map_name
```

```
Asa(config-cmap)# match regex regex-name
```

### 2.2.3.2. Giám sát lưu lượng DNS

- Bước 1: Tạo class-map

```
Asa(config)# class-map type inspect dns [match-all | match-any] class_map_name
```

Các câu lệnh xác định các đặc điểm của lưu lượng sau đây là tùy chọn:

```
Asa(config-cmap)# match [not] header-flag [eq] {f_name [f_name...]}
```

```
Asa(config-cmap)# match [not] dns-type eq t_name
```

```
Asa(config-cmap)# match [not] {question | resource-record {answer | authority | additional}}
```

```
Asa(config-cmap)# match [not] domain-name regex {regex_name | class class_name}
```

```
Asa(config-cmap)# exit
```

Từ khóa **not** chỉ định cho các lưu lượng không mang đặc trưng đã chỉ ra

- Bước 2: Tạo policy-map

```
Asa(config)# policy-map type inspect dns policy_map_name
```

Để chỉ ra các lưu lượng sẽ được áp đặt policy-map, có thể sử dụng một hoặc đồng thời hai cách sau để chỉ ra đặc điểm của lưu lượng đó:

- Sử dụng class-map

```
Asa(config-pmap)# class class-map-name
```

```
Asa(config-pmap-c)# {drop [log]} | {drop-connection [log]} | log
```

- Sử dụng lệnh **match** tương tự khi định nghĩa class-map cho DNS

```
Asa(config-pmap)# match something
```

```
Asa(config-pmap-c)# {drop [log]} | {drop-connection [log]}
```

Có thể liệt kê nhiều đặc điểm trong 1 policy-map. Nếu lưu lượng khớp với nhiều đặc điểm đã chỉ ra trong policy-map, nó sẽ được xử lý bởi tất cả các cách thức tương ứng cho tới khi bị **drop**. Tất cả các xử lý sau khi lưu lượng bị drop sẽ không có hiệu lực

- Bước 3: Đặt thông số cho bộ giám sát. Các lệnh sau đây là tùy chọn

```
Asa(config-pmap)# parameters
```

```
Asa(config-pmap-p)# dns-guard
```

```
Asa(config-pmap-p)# message-length maximum {length | client length | server length}
```

```
Asa(config-pmap-p)# nat-rewrite
```

```
Asa(config-pmap-p)# id-randomization
```

```
Asa(config-pmap-p)# protocol-enforcement
```

```
Asa(config-pmap-p)# id-mismatch count number duration time action log
```

### 2.2.3.3. Giám sát lưu lượng Web

- Bước 1: Tạo class-map

```
Asa(config)# class-map type inspect http [match-all | match-any] class_map_name
```

Các câu lệnh xác định các đặc điểm của lưu lượng sau đây là tùy chọn:

```
Asa(config-cmap)# match [not] req-resp content-type mismatch
```

```
Asa(config-cmap)# match [not] request args regex {regex_name | class class_name}
```

```
Asa(config-cmap)# match [not] request body {regex {regex_name | class class_name}  
| length gt N-bytes}
```

```
Asa(config-cmap)# match [not] request header {field | regex regex_name} regex  
{regex_name | class class_name}
```

```
Asa(config-cmap)# match [not] request header {field | regex {regex_name | class class_name}} {length gt N-bytes | count gt number}
```

```
Asa(config-cmap)# match [not] request header {length gt N-bytes | count gt number | non-ascii}
```

```
Asa(config-cmap)# match [not] request method {method | regex {regex_name | class class_name}}
```

```
Asa(config-cmap)# match [not] request uri {regex {regex_name | class class_name} | length gt bytes}
```

```
Asa(config-cmap)# match [not] response body {active-x | java-applet | regex {regex_name | class class_name}}
```

```
Asa(config-cmap)# match [not] response body length gt N-bytes
```

```
Asa(config-cmap)# match [not] response header {field | regex regex_name} regex {regex_name | class class_name}
```

```
Asa(config-cmap)# match [not] response header {field | regex {regex_name | class class_name}} {length gt bytes | count gt number}
```

```
Asa(config-cmap)# match [not] response header {length gt bytes | count gt number | non-ascii}
```

```
Asa(config-cmap)# match [not] response status-line regex {regex_name | class class_name}
```

- Bước 2: Tạo policy-map

```
Asa(config)# policy-map type inspect http policy_map_name
```

Để chỉ ra các lưu lượng sẽ được áp đặt policy-map, có thể sử dụng một hoặc đồng thời hai cách sau để chỉ ra đặc điểm của lưu lượng đó:

➤ Sử dụng class-map

```
Asa(config-pmap)# class class-map-name
```

```
Asa(config-pmap-c)# {drop-connection [log]} | {reset [log]} | {log}
```

➤ Sử dụng lệnh **match** tương tự khi định nghĩa class-map cho HTTP

```
Asa(config-pmap)# match something
```

```
Asa(config-pmap-c)# {drop-connection [log]} | {reset [log]} | {log}
```

- Bước 3: Đặt thông số cho bộ giám sát. Các lệnh sau đây là tùy chọn

```
Asa(config-pmap)# parameters
```

Asa(config-pmap-p)# **body-match-maximum** *number*

Asa(config-pmap-p)# **protocol-violation action** {**drop-connection** [log] | **reset** [log] | **log**}

Asa(config-pmap-p)# **spoof-server** *string*

**Xem thêm cách thức cấu hình giám sát các lưu lượng khác của tầng ứng dụng tại địa chỉ sau:**

<http://www.cisco.com/c/en/us/td/docs/security/asa/asa95/configuration/firewall/asa-95-firewall-config/inspect-basic.html#ID-2092-00000007>

### 2.3. Tường lửa iptables

iptables là phần mềm tường lửa bộ lọc gói phát triển từ framework Netfilter hoạt động trên nền tảng các HĐH Linux. iptables có thể triển khai theo cả hai dạng là host-based firewall và network-based firewall. Tuy nhiên, mô hình host-based firewall thường được ưa chuộng hơn.

iptables sử dụng 5 tập luật (chain):

- INPUT: tập luật áp dụng cho lưu lượng đi vào nút mạng
- OUTPUT: tập luật áp dụng cho lưu lượng đi ra khỏi nút mạng
- FORWARD: tập luật áp dụng cho lưu lượng được chuyển tiếp qua nút mạng (khi sử dụng iptables ở mô hình network-based firewall)
- PREROUTING và POSTROUTING: tập luật được sử dụng trong bảng nat để thay đổi thông tin tiêu đề cho gói tin

iptables bao gồm các bảng:

- filter table: lọc lưu lượng
- mangle table: thiết lập các thông số điều khiển lưu lượng
- nat table: chuyển đổi địa chỉ cho lưu lượng
- raw table: gán trạng thái cho gói tin
  - NEW: gói tin đầu tiên khởi tạo phiên
  - ESTABLISHED: gói tin trong phiên đã được thiết lập

- RELATED: gói tin trên phiên phát sinh từ một phiên ESTABLISHED
- INVALID: gói tin không hợp lệ
- UNTRACKED: gói tin được đánh dấu NOTRACK

Trong phần này, chúng ta chỉ quan tâm đến cấu hình tập luật cho bảng filter để lọc gói tin. iptables cung cấp các cách thức(TARGET) xử lý gói tin cho bảng filter như sau:

- ACCEPT: chấp nhận
- DROP: hủy
- LOG: ghi nhận vào syslog
- REJECT: hủy gói tin và gửi thông báo lỗi cho nút nguồn
- RETURN: gói tin được xử lý bởi default policy

Để so khớp thông tin trên gói tin và luật, iptables cung cấp các tùy chọn

- source (-s): so khớp địa chỉ nguồn
- destination (-d): so khớp địa chỉ đích
- protocol (-p): so khớp giao thức
- in-interface (-i): so khớp cổng vào của gói tin
- out-interface (-o): so khớp cổng ra của gói tin
- state: so khớp trạng thái của gói tin
- string: so khớp chuỗi bytes trong dữ liệu tầng ứng dụng mà gói tin mang theo

Các câu lệnh thay đổi tập luật của iptables:

- Thêm một luật vào cuối tập luật:

**iptables -A chain [match-options] -j target**

*match-options* là các tùy chọn để so khớp

**-p** *protocol*

**-s** *source-address*

**-d** *dest-address*

**-i** *input-interface*

**--sport** *source-port*

**--dport** *dest-port*

**-o** *output-interface*

**-m state --state** *packet-state*

- Chèn một luật:

**iptables -I** *chain rule-line [match-options] -j target*

- Xóa một luật

**iptables -D** *chain rule-line*

- Thiết lập luật mặc định

**iptables -P** *chain-name target*

- Hiển thị tập luật: **iptables -L** [*chain-name*] **--line-numbers**

- Lưu tập luật: **iptables save**

### 3. LUYỆN TẬP Ở NHÀ

#### 3.1. Chuẩn bị môi trường thực hành

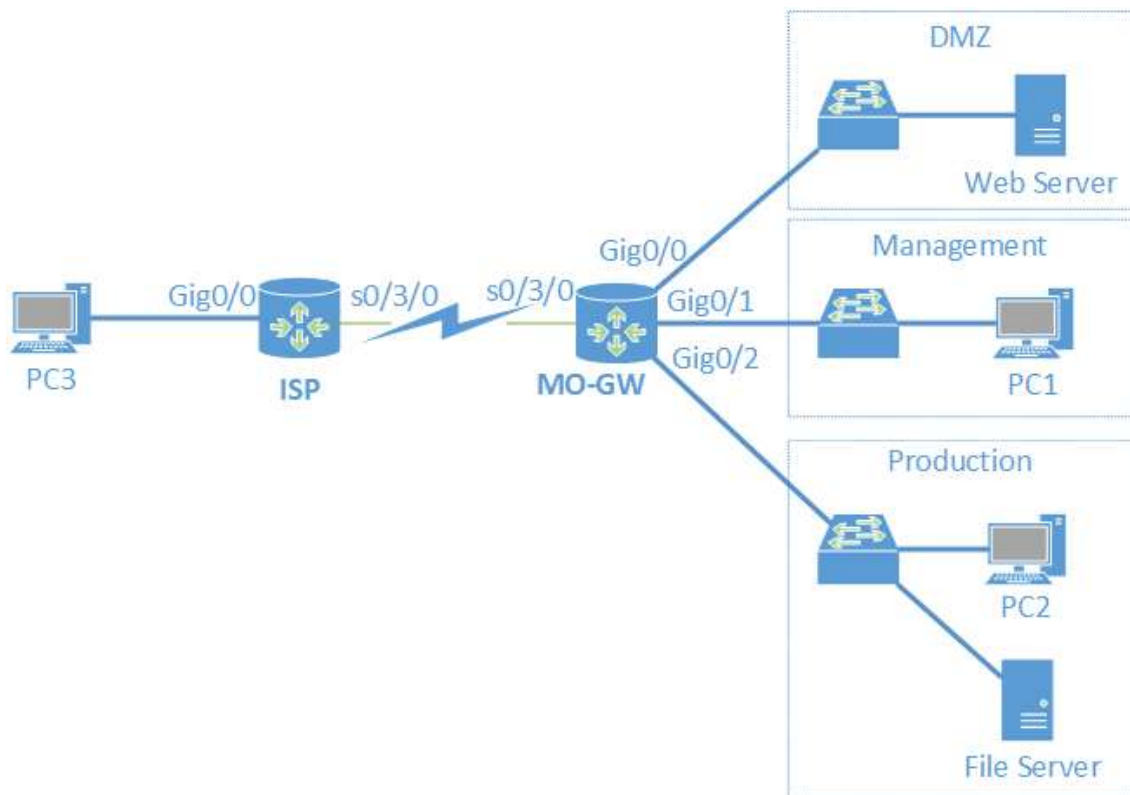
- Nội dung thực hành cài đặt và cấu hình tường lửa sử dụng ACL và Cisco ASA được thực hiện trên phần mềm giả lập Packet Tracer

- Thực hành cấu hình tường lửa iptables trên máy ảo cài đặt hệ điều hành Ubuntu16.4

<https://drive.google.com/file/d/1e2084qM9y9HbtbzWF1z3GedBR5qCJVpV>

#### 3.2. Cấu hình ACL trên Cisco router

Trên Packet Tracer tạo sơ đồ giả lập như sau, với X là số thứ tự của sinh viên:



- Mạng nội bộ có 3 phân vùng mạng kết nối với router MO-GW:

➤ Router MO-GW(Sử dụng router 2911):

- s0/3/0: 201.10.X.2 /29
- Gig0/0: 222.1.X.1 /24
- Gig0/1: 192.168.X.1 /24
- Gig0/2: 10.X.0.1 /16

➤ Phân vùng DMZ: 222.1.X.0 /24

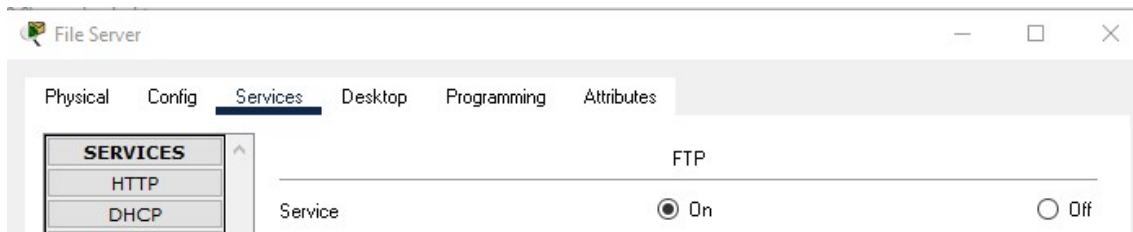
- Web Server: 222.1.X.10 /24. Để mở dịch vụ Web trên máy chủ này, ta chọn thẻ Services→HTTP và tích vào mục On cho HTTP và HTTPS.



➤ Phân vùng Management: 192.168.X.0 /24



- PC1: 192.168.X.100 /24
- Phân vùng Production: 10.X.0.0 /16
  - PC2: 10.X.0.100 /16
  - File Server: 10.X.255.20 /16. Để mở dịch vụ FTP trên máy chủ này, ta chọn thẻ Services→FTP và tích vào mục On.



- Mạng Internet được minh họa gồm có:
  - Router ISP (Sử dụng router 2911)
    - s0/3/0: 201.10.X.1 /29
    - Gig0/0: 202.20.X.1 /24
  - PC3: 202.20.X.2 /24

### 3.2.1. Cấu hình cơ bản cho các router

#### - Bước 1: Cấu hình router ISP

```
#config terminal
#interface s0/3/0
#ip address 201.10.X.1 255.255.255.248
#no shutdown
#interface Gig0/0
#ip address 202.20.X.1 255.255.255.0
#no shutdown
#exit
#ip route 222.1.X.0 255.255.255.0 201.10.X.2
#
#copy running-config startup-config
```

#### - Bước 2: Cấu hình router MO-GW

```

#config terminal
#interface s0/3/0
#ip address 201.10.X.2 255.255.255.248
#no shutdown
#interface Gig0/0
#ip address 222.1.X.1 255.255.255.0
#no shut
#interface Gig0/1
#ip address 192.168.X.1 255.255.255.0
#no shut
#interface Gig0/2
#ip address 10.X.0.1 255.255.0.0
#no shut
#exit
#ip route 0.0.0.0 0.0.0.0 201.10.X.1
#exit
#
#copy running-config startup-config

```

**Chúng ta kiểm tra kết nối cho thấy giữa các phân vùng mạng sau là thông suốt**

- Từ PC3 trên Internet tới Web Server trên phân vùng DMZ: Thành công
- Từ PC1 trên phân vùng Management tới Web Server trên phân vùng DMZ
- Từ PC2 trên phân vùng Production tới Web Server trên phân vùng DMZ
- Từ PC1 trên phân vùng Management tới PC2 trên phân vùng Production

Mặt khác, kết nối từ các phân vùng Management và Production tới Internet là không thông suốt.

Lưu lại file trên với tên là **Section3.2.pkt**

### **3.2.1. Cấu hình Standard ACL**

Trong phần này chúng ta sẽ sử dụng Standard ACLs với hai mục đích:

- Định nghĩa các lưu lượng cho dịch vụ NAT trên router: Thực hiện NAT cho phân vùng Production. Phân vùng này có địa chỉ 10.X.0.0 /16 nên dễ dàng xác định được mặt nạ kiểm tra (wild-card) là 0.0.255.255
- Lọc lưu lượng theo địa chỉ nguồn: Chỉ cho các nút từ phân vùng DMZ gửi dữ liệu tới phân vùng Management. Phân vùng này có địa chỉ 222.1.X.0 /24 nên dễ dàng xác định được mặt nạ kiểm tra (wild-card) là 0.0.0.255

**Tạo file bản sao của file Section3.2.pkt với tên là Section3.2.1.pkt**

### **Cấu hình và sử dụng Standard ACLs trên MO-GW**

```
#config terminal
#ip access-list standard NAT
#permit 10.X.0.0 0.0.255.255
#exit
#ip access-list standard toManagement
#permit 222.1.X.0 0.0.0.255
#exit
#ip nat inside source list NAT interface s0/3/0 overload
#
#interface s0/3/0
#ip nat outside
#interface Gig0/2
#ip nat inside
#
#interface Gig0/1
#ip access-group toManagement out
#
#copy running-config startup-config
```

Kiểm tra kết nối cho thấy phân vùng Production đã truy cập được mạng Internet. Đó là do chúng ta đã thực hiện cấu hình NAT cho phân vùng này bằng các câu lệnh:

```
#ip access-list standard NAT
#permit 10.X.0.0 0.0.255.255
```

```
#ip nat inside source list NAT interface s0/3/0 overload
```

Mặt khác, ta thấy từ phân vùng Production không thể truy cập vào phân vùng Management được nữa. Đó là do ta đã cấu hình ACL để chỉ cho phép lưu lượng từ phân vùng DMZ đi ra khỏi cổng Gig0/1 để vào phân vùng DMZ.

```
#ip access-list standard toManagement
#permit 222.1.X.0 0.0.0.255

#interface Gig0/1
#ip access-group toManagement out
```

*Cần lưu ý rằng, quy tắc của ACL là những lưu lượng không được liệt kê sẽ bị mặc định cấm.*

### 3.2.2. Cấu hình và sử dụng Extended ACL

Trong phần này chúng ta sẽ sử dụng Extended ACL để thực hiện các kiểm soát truy cập sau:

- (1) Chỉ cho phép các nút mạng từ mạng Internet truy cập vào dịch vụ Web trên máy chủ Web Server.
- (2) Chỉ không cho phép lưu lượng ICMP trao đổi giữa phân vùng Production và phân vùng DMZ.

Theo mục 2.1.4, ta nên sử dụng các ACL áp dụng cho luồng vào (inbound) trên các cổng. Do đó chúng ta định nghĩa 3 ACL để kiểm soát lưu lượng đi vào các cổng, hay nói cách khác là đi ra từ các vùng mạng tương ứng.

- 1 ACL áp dụng cho luồng vào (inbound) của cổng s0/3/0, là cổng kết nối với vùng mạng Internet, để đáp ứng yêu cầu (1)

- 1 ACL áp dụng cho luồng vào (inbound) của cổng Gig0/0, là cổng kết nối với vùng mạng DMZ. ACL này chỉ không cho phép gửi lưu lượng ICMP sang phân vùng Production.
- 1 ACL áp dụng cho luồng vào (inbound) của cổng Gig0/2, là cổng kết nối với vùng mạng Production. ACL này chỉ không cho phép gửi lưu lượng ICMP sang phân vùng DMZ.

***Tạo file bản sao của file Section3.2.pkt với tên là Section3.2.2.pkt***

#### **Cấu hình Extended ACLs trên MO-GW**

```
#config terminal
#ip access-list extended FromInternet
#permit tcp any host 222.1.X.10 eq 80
#permit tcp any host 222.1.X.10 eq 443
#exit
#
#interface s0/3/0
#ip access-group FromInternet in
#exit
#
#ip access-list extended FromDMZ
#deny icmp any 10.X.0.0 0.0.255.255
#permit any any
#exit
#
#interface Gig0/0
#ip access-group FromDMZ in
#exit
#
#ip access-list extended FromProduction
#deny icmp any 222.1.X.0 0.0.0.255
```

```
#permit any any
#exit
#
#interface Gig0/2
#ip access-group FromProduction in
#exit
#
#exit
#copy running-config startup-config
```

Thực hiện các thao tác kiểm tra sau để cho thấy các ACL đã hoạt động:

- Từ PC3, truy cập Website 222.1.X.10 thành công. Từ PC3, thực hiện lệnh ping tới máy chủ Web server có kết quả thất bại. Các kết quả này là do ta đã định nghĩa ACL FromInternet và áp dụng lên luồng vào (inbound) của cổng s0/3/0. Với ACL này, ta thấy rằng chỉ có lưu lượng truy cập tới dịch vụ Web trên máy có địa chỉ 222.1.X.10 mới được chấp nhận. Mặt khác, lưu lượng Web trả lời từ máy chủ cho vùng Internet là được chấp nhận vào cổng Gig0/0 do ACL trên cổng này chỉ cấm lưu lượng ICMP đi tới phân vùng Production.

Lưu lượng ICMP do lệnh ping không được liệt kê trong ACL FromInternet nên không được chấp nhận, do quy tắc mặc định cấm của ACL.

- Từ PC2, truy cập Website 222.1.X.10 thành công. Từ PC2, thực hiện lệnh ping tới máy chủ Web server có kết quả thất bại. Các kết quả này là do ta đã định nghĩa ACL FromProduction và áp dụng lên luồng vào (inbound) của cổng Gig0/2. Với ACL này, ta thấy lưu lượng ICMP tới phân vùng DMZ bị cấm còn mọi lưu lượng khác được cho phép. Mặt khác, lưu lượng Web trả lời từ máy chủ Web cho vùng Production là được chấp nhận vào cổng Gig0/0 do ACL trên cổng này chỉ cấm lưu lượng ICMP đi tới phân vùng Production.



- Từ Web Server, truy cập dịch vụ FTP trên File Server thành công.

```
C:\>ftp 10.100.255.20
Trying to connect...10.100.255.20
Connected to 10.100.255.20
220- Welcome to PT Ftp server
Username:
```

Từ Web Server, thực hiện lệnh ping tới PC2 hoặc File Server có kết quả thất bại. Các kết quả này là do ta đã định nghĩa ACL FromDMZ và áp dụng lên luồng vào (inbound) của cổng Gig0/0. Với ACL này, ta thấy lưu lượng ICMP tới phân vùng Production bị cấm còn mọi lưu lượng khác được cho phép. Mặt khác, lưu lượng FTP trả lời từ máy chủ File Server cho vùng DMZ là được chấp nhận vào cổng Gig0/2 do ACL FromProduction trên cổng này chỉ cấm lưu lượng ICMP đi tới phân vùng DMZ.

### 3.2.3. Cấu hình và sử dụng Reflexive ACLs (Không bắt buộc luyện tập)

Trong phần này chúng ta sẽ thực hiện cấu hình ACL trên MO-GW để cho phép các máy trạm từ mạng 192.168.X.0 /24 có thể sử dụng lệnh ping để kiểm tra kết nối tới các máy trạm trên mạng 10.X.0.0 /16 nhưng cấm theo chiều ngược lại.

#### 3.2.3.1. Sử dụng Extended ACLs

**Tạo file bản sao của file Section3.2.pkt với tên là Section3.2.3.1.pkt**

- **Bước 1:** Cấu hình sử dụng ACL để cho phép gói tin ICMP từ 192.168.X.0 /24 tới 10.X.0.0 /16

```
#config terminal
#ip access-list extended toLAN20
#permit icmp 192.168.X.0 0.0.0.255 172.16.X.0 0.0.255.255
#exit
#interface Gig0/1
```

```
#ip access-group toLAN20 in
#exit
#
#copy running-config startup-config
```

- **Bước 2:** Cấu hình sử dụng ACL để cấm các gói tin ICMP từ 10.X.0.0 /16 tới 192.168.X.0 /24

```
#config terminal
#ip access-list extended toLAN10
#deny icmp 172.16.X.0 0.0.255.255 192.168.X.0 0.0.0.255
#exit
#interface Gig0/0
#ip access-group toLAN10 in
#exit
#
#copy running-config startup-config
```

- **Bước 3:** Thực hiện lệnh ping từ PC1 sang Server ta thấy kết quả là thông báo lỗi. Lý do là ACL toLAN20 sử dụng trên cổng Gig0/0 đã chặn các gói tin ICMP Reply mà Server trả lời cho PC1.

Để giải quyết giải quyết yêu cầu đặt ra, ta phải cấu hình các ACL để chỉ cho phép các gói tin ICMP từ mạng 10.X.0.0 /16 tới mạng 192.168.X.0 /24 nếu trước đó đã có các gói tin ICMP gửi từ mạng 192.168.X.0 /24 sang mạng 10.X.0.0 /16. Giải pháp là sử dụng Reflexive ACLs.

### 3.2.3.2. Sử dụng Reflexive ACLs

*Tạo file bản sao của file Section3.2.pkt với tên là Section3.2.3.2pkt*

- **Bước 1:** Cấu hình Reflexive ACLs trên MO-GW

```
#config terminal
#ip access-list extended toLAN20
```

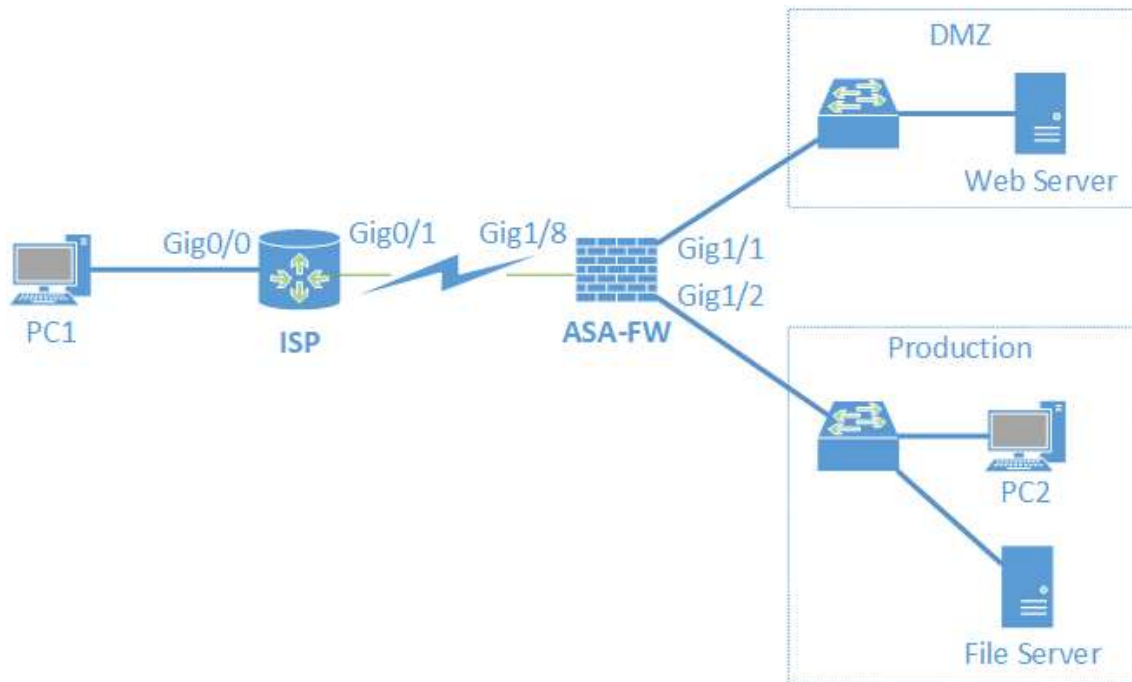


```
#permit icmp 192.168.X.0 0.0.0.255 172.16.X.0 0.0.255.255 reflect ICMPtraffic
#exit
#
#ip access-list extended toLAN10
#evaluate ICMPtraffic
#exit
#
#interface Gig0/1
#ip access-group toLAN20 in
#interface Gig0/0
#ip access-group toLAN10 in
#exit
#
#copy running-config startup-config
```

- **Bước 2:** Kiểm tra nội dung các ACL đã định nghĩa và ghi nhớ vào báo cáo.
- **Bước 3:** Thực hiện lệnh ping từ PC1 sang Server.

### 3.3. Cấu hình tường lửa Cisco ASA

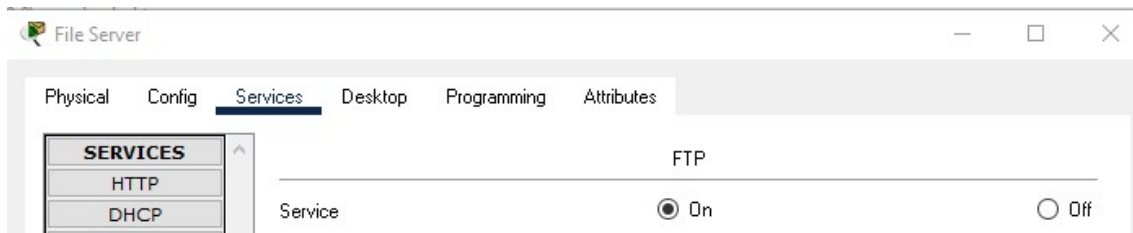
Tạo sơ đồ giả lập như hình dưới đây, trong đó X là số thứ tự của sinh viên.



- Mạng nội bộ có 3 phân vùng mạng kết nối với tường lửa ASA-FW
  - Tường lửa ASA-FW(Sử dụng tường lửa Cisco ASA 5506)
    - Gig1/8: 201.10.X.2 /24
    - Gig1/1: 222.1.X.1 /24
    - Gig1/2: 10.X.0.1 /16
  - Phân vùng DMZ: 222.1.X.0 /24
    - Web Server: 222.1.X.10 /24. Để mở dịch vụ Web trên máy chủ này, ta chọn thẻ Services→HTTP và tích vào mục On cho HTTP và HTTPS.



- Phân vùng Production: 10.X.0.0 /16
  - PC2: 10.X.0.100 /16
  - File Server: 10.X.255.20 /16. Để mở dịch vụ FTP trên máy chủ này, ta chọn thẻ Services→FTP và tích vào mục On.



- Mạng Internet được minh họa gồm có:

➤ Router ISP (Sử dụng router 2911)

- Gig0/1: 201.10.X.1 /29
- Gig0/0: 202.20.X.1 /24

➤ PC1: 202.20.X.2 /24

### Cấu hình router ISP

```
#config terminal
#interface Gig0/0
#ip address 202.20.X.1 255.255.255.0
#no shutdown
#interface Gig0/1
#ip address 201.10.X.1 255.255.255.248
#no shutdown
#exit
#ip route 222.1.X.0 255.255.255.0 201.10.X.2
#exit
#
#copy running-config startup-config
```

#### 3.3.1. Cấu hình kiểm soát lưu lượng theo mức an ninh

Chúng ta sẽ thực hiện một số cấu hình cơ bản trên tường lửa Cisco ASA, trong đó thực hiện giám sát lưu lượng dựa trên mức an ninh để thực hiện các yêu cầu sau:

(1) Cho phép lưu lượng yêu cầu từ phần vùng DMZ và Production ra Internet và lưu lượng đáp ứng ngược lại.

(2) Cho phép lưu lượng yêu cầu từ phần vùng Production tới DMZ và lưu lượng đáp ứng ngược lại.

Với 2 yêu cầu như trên, ta sẽ cấu hình mức an ninh theo thứ tự tăng dần cho các cổng trên firewall kết nối lần lượt với các vùng mạng Internet, DMZ và Production.

**- Bước 2: Thực hiện cấu hình ASA như sau:**

```
>enable
Password: (Bỏ trống)
#config terminal
#
#interface Gig1/8
#nameif Internet
#ip address 201.10.X.2 255.255.255.0
#no shutdown
#
#interface Gig1/1
#nameif DMZ
#security-level 30
#ip address 222.1.X.1 255.255.255.0
#no shutdown
#
#interface Gig1/2
#nameif Production
#security-level 80
#ip address 10.X.0.1 255.255.0.0
#no shutdown
#exit
#
#route Internet 0.0.0.0 0.0.0.0 201.10.X.1
#
```

```

#object network ProductionSubnet
#subnet 10.X.0.0 255.255.0.0
#nat (Production,Internet) dynamic interface
#exit
#
#class-map ICMPclass
#match default-inspection-traffic
#exit
#policy-map ICMPpolicy
#class ICMPclass
#inspect icmp
#exit
#service-policy ICMPpolicy interface Production
#service-policy ICMPpolicy interface DMZ
#exit
#
#write memory

```

- Từ PC1, kiểm tra kết nối tới Web Server bằng lệnh ping và truy cập vào dịch vụ Web trên Web Server đều có kết quả thất bại. PC1 nằm trên vùng mạng Internet, kết nối với cổng Gig1/8 có mức an ninh là 0, thấp hơn mức an ninh của cổng Gig1/1, cổng kết nối với phân vùng DMZ của Web Server, là 30. Vì vậy, lưu lượng yêu cầu từ vùng Internet không được đi sang phân vùng DMZ.

- Từ Web Server, kiểm tra kết nối tới File Server và truy cập vào dịch vụ FTP trên File Server đều có kết quả thất bại. Web Server nằm trên phân vùng DMZ, kết nối với cổng Gig1/1 có mức an ninh là 30, thấp hơn mức an ninh của cổng Gig1/2, cổng kết nối với phân vùng Production của File Server, là 80. Vì vậy, lưu lượng yêu cầu từ vùng DMZ không được đi sang phân vùng Production.

- Từ PC2, kiểm tra kết nối tới Web Server bằng lệnh ping và truy cập vào dịch vụ Web trên Web Server cho kết quả thành công. PC2 nằm trên phân vùng Production kết nối với cổng Gig1/2 có mức an ninh cao hơn cổng Gig1/1 là cổng kết nối với phân vùng DMZ. Do đó, lưu lượng yêu cầu từ phân vùng Production được phép đi sang phân vùng DMZ, và lưu lượng đáp ứng theo chiều ngược lại cũng được phép đi qua.

- Từ PC2, kiểm tra kết nối tới PC1 bằng lệnh ping cho kết quả thành công. PC2 nằm trên phân vùng Production kết nối với cổng Gig1/2 có mức an ninh cao hơn cổng Gig1/8 là cổng kết nối với vùng Internet. Do đó, lưu lượng yêu cầu từ phân vùng Production được phép đi sang vùng Internet, và lưu lượng đáp ứng theo chiều ngược lại cũng được phép đi qua. Mặt khác, trên tường lửa ta cũng đã cấu hình dịch vụ NAT cho phân vùng Production.

### 3.3.2. Cấu hình ACL

Trên tường lửa Cisco ASA, theo mặc định, lưu lượng yêu cầu không được đi từ phần vùng có mức an ninh thấp sang phân vùng có mức an ninh cao. Đó là lý do tại sao từ PC1 trên vùng Internet không thể truy cập tới Web Server trên phân vùng DMZ. Trong phần này, chúng ta sẽ thực hiện cấu hình ACL để cho phép PC1 có thể truy cập dịch vụ của Web Server

**Thực hiện cấu hình trên tường lửa ASA-FW như sau**

```
#config terminal
#
#access-list FromInternetFilter extended permit tcp any host 222.1.X.10 eq 80
#access-list FromInternetFilter extended permit tcp any host 222.1.X.10 eq 443
#access-group FromInternetFilter in interface Internet
#
#exit
#write memory
```

Sau khi thực hiện cấu hình trên, ta kiểm tra lại thấy từ PC1 đã truy cập được vào website trên máy chủ Web Server.

Lưu lại file đã giả lập với tên là **Section3.3.pkt**

## 3.4. Cấu hình tường lửa iptables

### 3.4.1. Chuẩn bị môi trường

#### Cài đặt và cấu hình Virtualbox

- **Bước 1:** Download phần mềm Virtualbox tại địa chỉ sau và cài đặt như một phần mềm thông thường trên Windows:

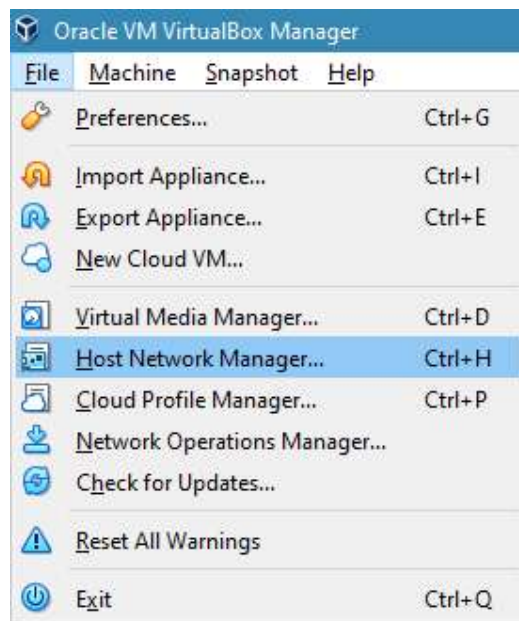
<https://download.virtualbox.org/virtualbox/6.1.12/VirtualBox-6.1.12-139181-Win.exe>

- **Bước 2:** Download gói mở rộng cho Virtualbox từ địa chỉ sau:

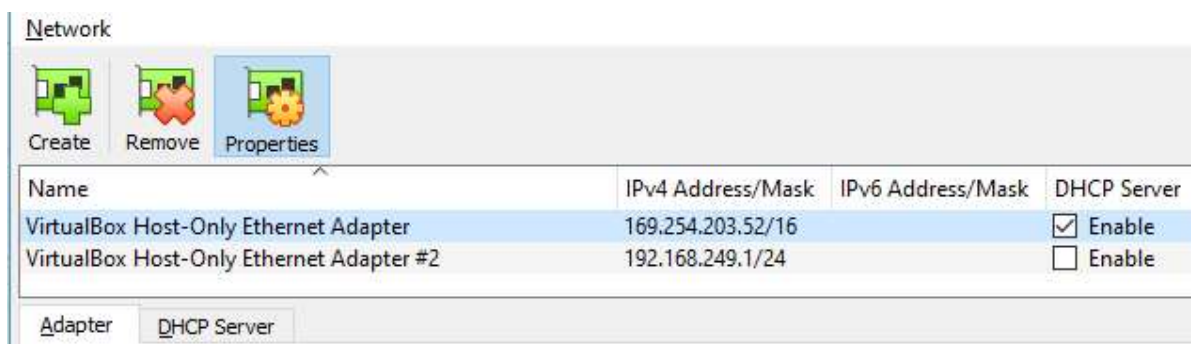
[https://download.virtualbox.org/virtualbox/6.1.12/Oracle\\_VM\\_VirtualBox\\_Extension\\_Pack-6.1.12.vbox-extpack](https://download.virtualbox.org/virtualbox/6.1.12/Oracle_VM_VirtualBox_Extension_Pack-6.1.12.vbox-extpack)

Sau khi download xong, nhấp đúp chuột vào file để cài đặt.

- **Bước 3:** Khởi động phần mềm Virtualbox
- **Bước 4:** Trên giao diện của Virtualbox, chọn File → Host Network Manager...



- **Bước 5:** Chọn các mạng ảo VirtualBox Host-Only Ethernet Adapter. Chọn



- **Bước 6:** Chọn thẻ Adapter và lựa chọn Configure Adapter Automatically.
- **Bước 7:** Chọn thẻ DHCP Server và thiết lập các thông số như sau:

Server Address: 192.168.X.2

Server Mask: 255.255.255.0

Lower Address Bound: 192.168.X.3

Upper Address Bound: 192.168.X.254

Trong đó X là số thứ tự của sinh viên trong danh sách. Hình ảnh sau minh họa với X = 117.

DHCP Server	
<input checked="" type="checkbox"/> Enable Server	
Server Address:	192.168.117.2
Server Mask:	255.255.255.0
Lower Address Bound:	192.168.117.3
Upper Address Bound:	192.168.117.254

- **Bước 8:** Nhấp nút Apply và Close để hoàn tất.

### Triển khai máy ảo Server

- **Bước 1:** Download máy ảo từ địa chỉ sau và giải nén

[https://drive.google.com/drive/folders/1NPW\\_zza6xLlecaLvmUXGQVlvmcRvsZYx?usp=sharing](https://drive.google.com/drive/folders/1NPW_zza6xLlecaLvmUXGQVlvmcRvsZYx?usp=sharing)

- **Bước 2:** Trên cửa sổ chính của Virtualbox, chọn Machine → New...



- **Bước 3:** Trên cửa sổ tạo máy ảo, đặt các thông số như sau. Sau đó nhấn **Next**.
  - **Name:** Tên máy ảo
  - **Machine Folder:** Thư mục chứa máy ảo
  - **Type:** Linux
  - **Version:** Ubuntu (32-bit)




## Name and operating system

Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name:

Machine Folder:

Type:  

Version:

- **Bước 4:** Chọn dung lượng bộ nhớ RAM cho máy ảo là 2048 MB. Nhấn Next để tiếp tục.

## Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **1024 MB**.

MB

4 MB 8192 MB

- **Bước 5:** Trong cửa sổ Hard disk tạo ổ cứng máy ảo, chọn mục **Use an existing virtual hard disk file**. Sau đó bấm nút **Choose a virtual hard disk file...**

**Hard disk**

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

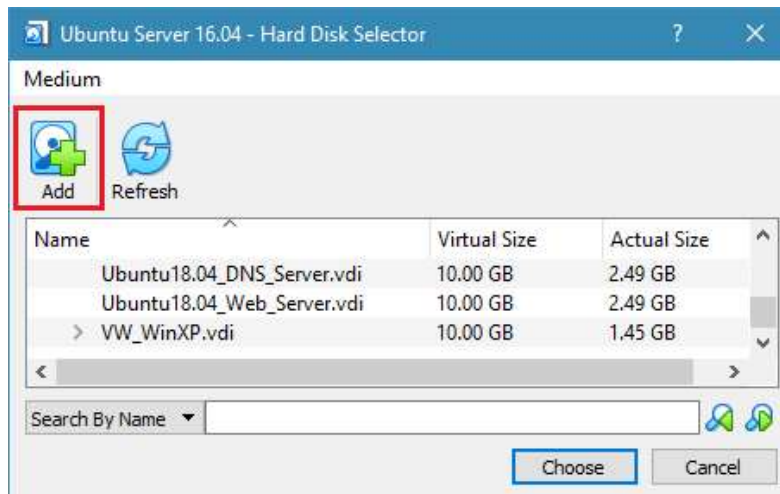
The recommended size of the hard disk is **10.00 GB**.

☐ Do not add a virtual hard disk

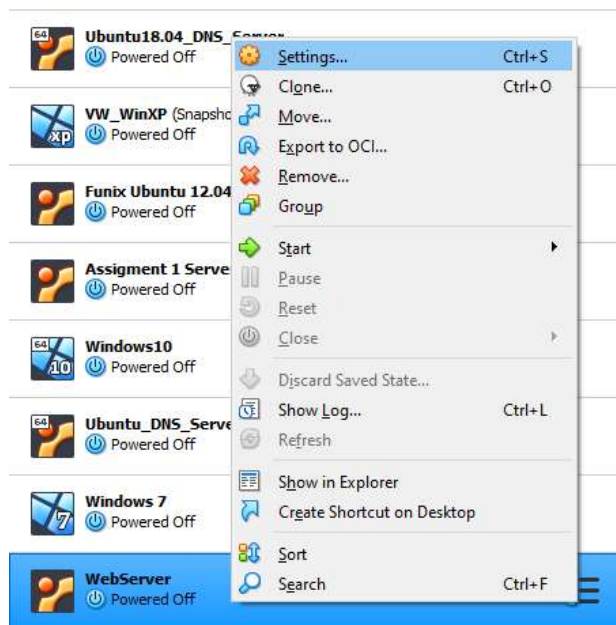
☐ Create a virtual hard disk now

☒ Use an existing virtual hard disk file

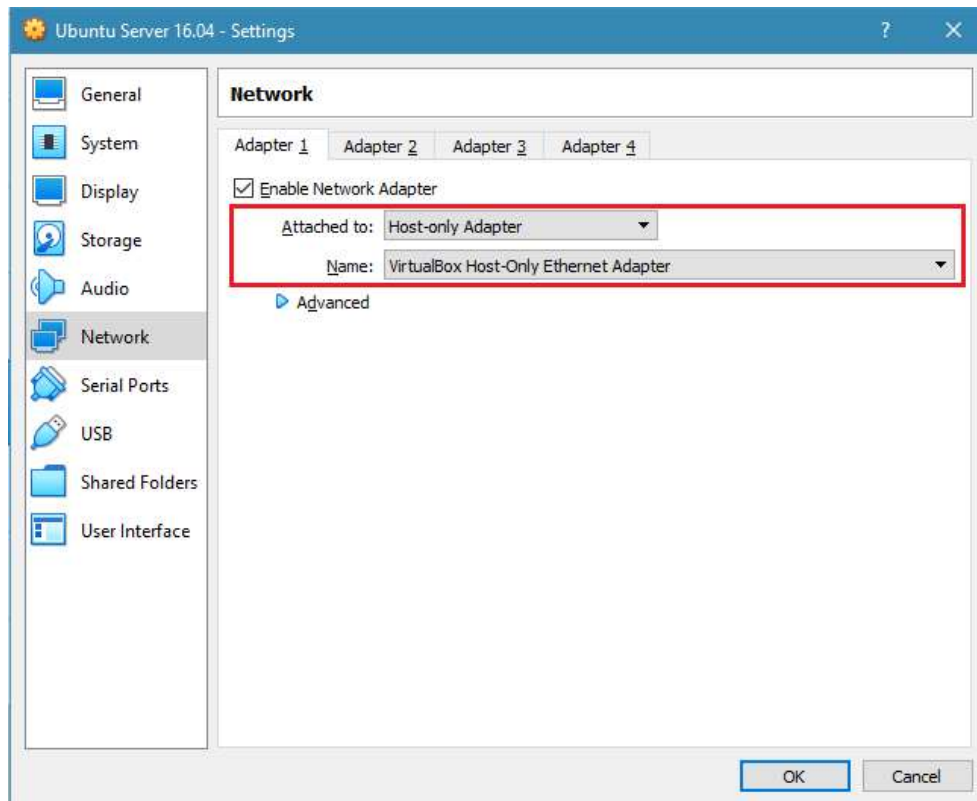
- **Bước 6:** Trên cửa sổ Hard Disk Selector, nhấn Add và chọn file Server.vdi đã download ở bước 1 để thêm vào danh sách



- **Bước 7:** Chọn file Server.vdi vừa được thêm vào trong danh sách ổ cứng ảo. Nhấn Choose để lựa chọn và đóng cửa sổ.
- **Bước 8:** Trên cửa sổ Hard disk, nhấn Create để tạo ổ cứng ảo.
- **Bước 9:** Trên cửa sổ chính của Virtualbox, chọn máy ảo vừa tạo và nhấp chuột phải. Chọn **Settings...**



- **Bước 10:** Chọn Network → Adapter 1. Thiết lập các thông số như sau:
  - **Attached to:** Host-only Adapter
  - **Name:** VirtualBox Host-Only Ethernet Adapter (hoặc còn gọi là VirtualBox Host-Only Network)



Sau khi máy ảo khởi động xong, đăng nhập bằng tài khoản sau:

- Username: bkcs
- Password: bkcs

- **Bước 11:** Trên Server mở cửa sổ Terminal và thực hiện lệnh `ifconfig`. Trong hình ảnh minh họa sau, địa chỉ của Server là 192.168.117.13

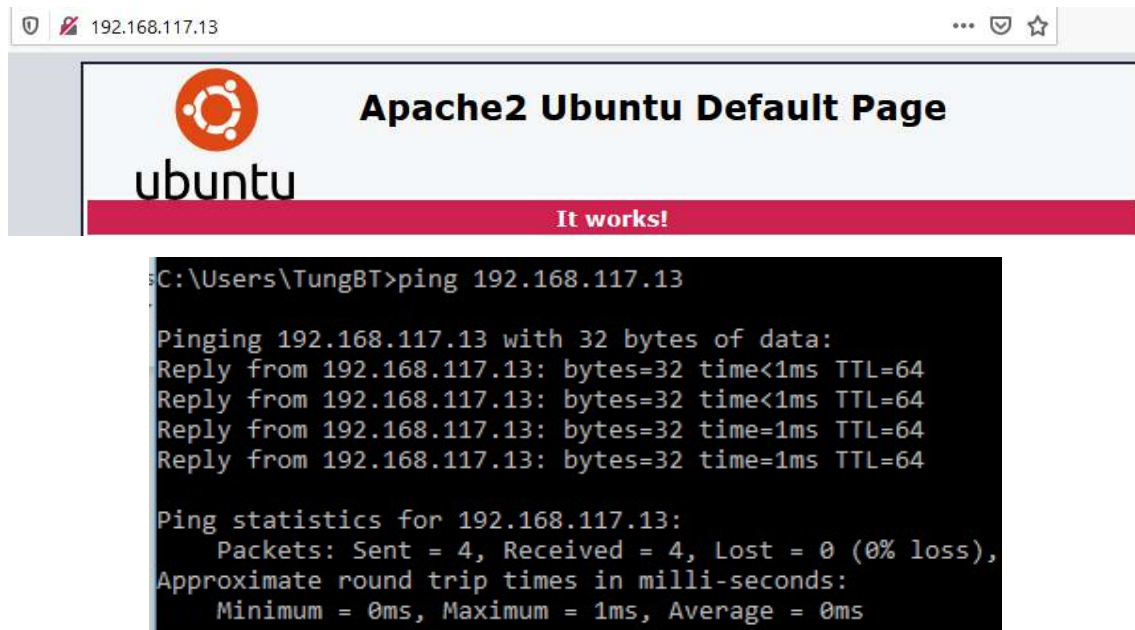
```
bkcs@ubuntu:~$ ifconfig
enp0s8  Link encap:Ethernet  HWaddr 08:00:27:44:38:b0
        inet addr:192.168.117.13  Bcast:192.168.117.255  Mask:255.255.255.0
        inet6 addr: fe80::a00:27ff:fe44:38b0/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:24 errors:0 dropped:0 overruns:0 frame:0
        TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:3339 (3.3 KB)  TX bytes:648 (648.0 B)
```

- **Bước 12:** Cấu hình địa chỉ IP tĩnh cho máy vật lý với địa chỉ là **192.168.X.100/24** trong đó X là STT của sinh viên trong danh sách.

### 3.4.2. Cấu hình tường lửa

Trong phần này, ta sẽ thực hiện các lệnh để thêm luật cho iptables để kiểm soát lưu lượng tới máy ảo Server.

- **Bước 1:** Trên máy vật lý, kiểm tra kết nối bằng lệnh `ping` và truy cập vào dịch vụ Web trên Server. Kết quả kiểm tra là thành công.



- **Bước 2:** Khởi động máy ảo Ubuntu. Mở cửa sổ Terminal và thực hiện lệnh  
**#sudo iptables -L --line-numbers**

```
bkcs@ubuntu:~$ sudo iptables -L --line-numbers
Chain INPUT (policy ACCEPT)
num target      prot opt source      destination
Chain FORWARD (policy ACCEPT)
num target      prot opt source      destination
Chain OUTPUT (policy ACCEPT)
num target      prot opt source      destination
```

Kết quả cho thấy chính sách mặc định trên tập luật INPUT là ACCEPT, tức là những lưu lượng không được liệt kê sẽ được chấp nhận. Việc sử dụng chính sách mặc định cho phép này là kém an toàn nên ta đổi sang chính sách mặc định cấm bằng lệnh sau:

**#sudo iptables -P INPUT DROP**

- **Bước 3:** Thực hiện các kiểm tra giống như bước 1 ta thấy kết quả là thất bại. Như vậy, lệnh thay đổi cấp hình trên đã có tác dụng.

- **Bước 4:** Trước khi thêm các luật kiểm soát truy cập tới dịch vụ, cần thêm một luật để chấp nhận mọi gói tin gửi tới máy chủ trên phiên truy cập đã được khởi tạo. Thực hiện lệnh sau để thêm luật

**sudo iptables -A INPUT -m state --state ESTABLISHED, RELATED -j ACCEPT**

- **Bước 5:** Thực hiện lệnh sau để cho phép các lưu lượng yêu cầu dịch vụ Web từ địa chỉ bất kỳ được chấp nhận.

**sudo iptables -A INPUT -p tcp --dport 80 -m state --state NEW -j ACCEPT**

**sudo iptables -A INPUT -p tcp --dport 443 -m state --state NEW -j ACCEPT**

- **Bước 6:** Từ máy vật lý, ta thấy đã truy cập được vào dịch vụ Web

- **Bước 7:** Dịch vụ SSH cung cấp kết nối điều khiển từ xa, chỉ nên cho phép truy cập từ các máy người dùng quản trị. Trong ví dụ này, giả sử máy của người dùng quản trị có địa chỉ 192.168.117.100.

**sudo iptable -A INPUT -p tcp -i enp0s8 -s 192.168.117.5 --dport 22 -m state --state NEW -j ACCEPT**

- **Bước 8:** Từ máy chủ vật lý, ta thấy kết nối tới dịch vụ SSH là thành công

- **Bước 9:** Thay đổi địa chỉ của máy chủ vật lý thành địa chỉ khác ta thấy không còn truy cập được tới dịch vụ SSH trên Server nữa. Đó là do ta đã cấu hình luật số 7.

- **Bước 10:** Thực hiện lệnh để xem danh sách các luật đã định nghĩa

**#sudo iptables -L --line-numbers**

```
bkcs@ubuntu:~$ sudo iptables -L --line-numbers
Chain INPUT (policy DROP)
num target      prot opt source                destination              state
1  ACCEPT        all  --  anywhere               anywhere                  state RELATED,ESTABLISHED
2  ACCEPT        tcp  --  anywhere               anywhere                  tcp dpt:http state NEW
3  ACCEPT        tcp  --  anywhere               anywhere                  tcp dpt:https state NEW
4  ACCEPT        tcp  --  192.168.117.100        anywhere                  tcp dpt:ssh state NEW

Chain FORWARD (policy ACCEPT)
num target      prot opt source                destination

Chain OUTPUT (policy ACCEPT)
num target      prot opt source                destination
```

- **Bước 11:** Thực hiện lệnh sau để lưu lại các luật của iptables

**#sudo netfilter-persistent save**

## 4. NỘI DUNG THỰC HÀNH

Cấu hình tường lửa theo yêu cầu của giáo viên hướng dẫn.