LAB: Filtering (UFW)

Lab Environment

The workshop WiFi:

- SSID: `workshop`
- PASS: `iij/2497`

Hosts - Virtual machines (Ubuntu 18.04LTS/LXC)

- Hostname: `nsXX.workshop`
- IPv6: `fd00:2497:1::X`
- IPv4: `10.0.0.X`

Where `X` and `XX` is yoru group ID. For group 1, hostname is `ns01.workshop`, IPv6 address is `fd00:2497:1::1`, and IPv4 is `10.0.0.1`.

Exercise 0: Install Apache

Before starting this hands on, let's install Apache web server to open local port on your virtual machine

```
$ sudo apt-get install apache2
```

Check if your web server is running before moving forward.

Exercise 1: Check UFW status

UFW (Uncomplicated Firewall) is filtering management package bundled with recent Linux distributions.

Check the current status of UFW using the `ufw` command.

```
$ sudo ufw status
```

By default, UFW is inactivated.

Exercise 2: Activate UFW

The default policy of UFW is deny all the incoming connections. Before activating UFW, we need to setup a ssh filtering rule, otherwise, we will lose ssh access from outside just after activating UFW.

```
$ sudo ufw allow ssh
```

And activate UFW.

```
$ sudo ufw enable
```

Check the status.

```
$ sudo ufw status
```

Exercise 3: Add a rule for SSH

Configure your UFW to accept SSH connections.

Exercise 4: Check the filter

Now the only ssh port is accessible. Check what happens if you use other ports like,

- HTTP
- DNS

Check which ports are open using `nmap`.

Exercise 5: Open ports

Now the only ssh port is available on your virtual server. You cannot access your web server anymore.

Add a new rule to allow HTTP and DNS access.

```
$ sudo ufw allow http
$ sudo ufw allow domain
```

Exercise 6: Close ports

You have two different filter rules to close a port.

```
$ sudo ufw deny http
```

```
$ sudo ufw reject http
```

What are the difference?

Exercise 6: Delete rules

If you don't need some rules anymore, you can delete rules. First, check the index of each rule.

```
$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 22/tcp                     ALLOW IN    Anywhere
[ 2] 80/tcp                     ALLOW IN    Anywhere
[ 3] 53                         ALLOW IN    Anywhere
[ 4] 22/tcp (v6)                ALLOW IN    Anywhere (v6)
[ 5] 80/tcp (v6)                ALLOW IN    Anywhere (v6)
[ 6] 53 (v6)                    ALLOW IN    Anywhere (v6)
```

Remove a rule by specifing the rule index.

```
$ ufw delete 2
```

You need to check the index each time whenever you remove multiple rules, since the indices are renumbered after deletion.

Exercise 7: Change default

By default, ufw deny all the incoming connections, but allow all the outgoing connections. In some cases, you may want to limit only specific services.

Let's try to configure to deny all the outgoing connections, and allow some selected servies only on yoru virtual server.

> Note: Check the man page to change the default behavior.

Exercise 8: Fine grained rulesets

If you want to access to your web server from a specific client, you can specify the rule as below, for example.

```
$ sudo ufw allow proto tcp from 10.0.0.2 to any port 80
```

Try to configure UFW to accept only from your PC.

Advanced Exercise: Deny ICMP Echo Request

ICMP/ICMPv6 rules are not handled by UFW. Find the way to control such traffic.