Лабораторно упражнение 6
FIREWALL

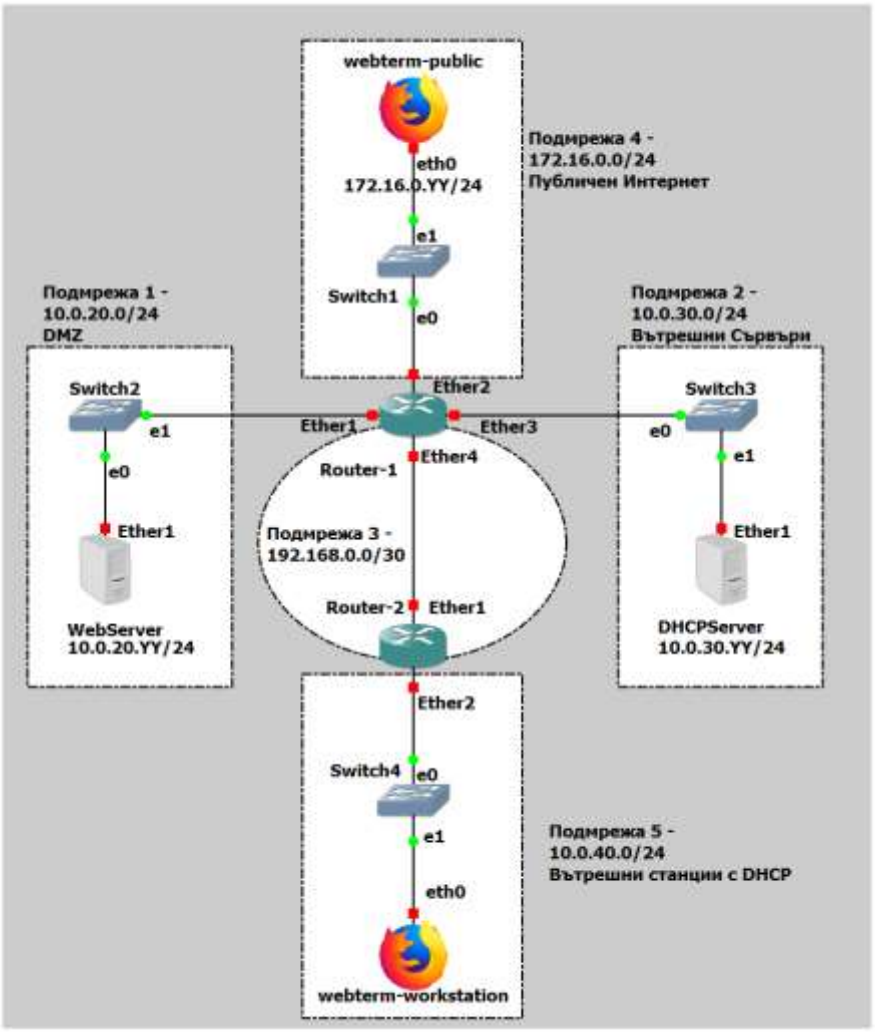
Обзор

В тази лабораторна работа ще конфигурирате виртуална GNS3 топология с интернет и ще конфигурирате елементарни firewall правила.

При използването на логически адреси `.YY` заместваме с последните 2 цифри от факултетния номер.

Създаване на мрежата

В GNS3 създайте мрежова топология, която съответства на тази:



Мрежова диаграма 1 Лабораторно 6 (Забележка: Етикетите на подмрежата и пунктираните граници са само за информация)

Тук се създават няколко подмрежи:

„**Публичен интернет**“ – Подмрежа, представляваща устройства, които са външни за вашата AS (организация)

"**DMZ**" (демилитаризирана зона) - Подмрежа, съдържаща сървъри, които трябва да бъдат публично достъпни

„**Вътрешни сървъри**“ – Подмрежа, съдържаща сървъри, които трябва да бъдат достъпни само от вътрешни устройства

„Вътрешни работни станции“ – Подмрежа, съдържаща компютри на крайни потребители (лаптопи, работни станции и т.н.), които трябва да бъдат достъпни само от вътрешни устройства

Обърнете внимание, че „сървърите“, показани на мрежовата диаграма, са просто MikroTik рутери, които са преназначени за нова роля. GNS3 е в състояние да поддържа пълноправни виртуални машини, използвани за сървър, като част от симулираната мрежа. Тук напълно достатъчно ни е Mikrotik OS за всички нужди.

Съвети:

- Процесът върви по-гладко, ако първо конфигурирате рутерите, а след това и компютрите във всяка подмрежа.
- Конкретният порт на комутатор няма значение
- Конкретният порт на рутера има значение. Конфигурацията на рутера в софтуера трябва да е в съответствие с начина, по който кабелите са свързани в хардуера.

Стъпки за конфигуриране:

1. Конфигурирайте имената на рутерите в GNS3, за да предотвратите объркване (чрез GUI).
2. Конфигурирайте имената на самия рутер, за да предотвратите объркване (чрез CLI).
3. Конфигурирайте IP адреси на всички интерфейси на рутера, които са свързани към подмрежи.
4. Деактивирайте DHCP клиента на всеки рутер. `ip dhcp-client print`, последван от `ip dhcp-client remove numbers=0`, ще премахне това.
5. Конфигурирайте динамично маршрутизиране (RIP) между подмрежи 1-5. След конфигуриране проверете с `routing rip route print`, че таблицата с маршрути е такава, каквато желаете.
    1) `routing rip interface add interface=etherX send=v2 receive=v2` # За интерфейса, който отива към други рутери
    2) `routing rip interface add interface=etherY passive=yes` # За интерфейса, който отива към клиентите
6. Конфигурирайте статичен IP адрес на webtern-public
7. Активирайте DHCP клиента на Webterm-workstation.

Фалшиви сървъри

MikroTik Router като уеб сървър

Вместо да създаваме друга виртуална машина на Linux и да инсталираме уеб сървър, просто ще използваме съществуващия рутер MikroTik като наш примерен уеб сървър за DMZ. В крайна сметка, той вече има уеб интерфейс, който работи.

1. Задайте на "сървъра" име на хоста (`WebServer`)

2. Задайте на "сървъра" IP адрес
3. Задайте на "сървъра" статичен маршрут по подразбиране, така че да изпраща целия трафик към `Router1`. Съвет: Разгледайте предишните лабораторни за това как да създадете статичен маршрут. За да го направите маршрут по подразбиране, адресът на дестинацията трябва да бъде 0.0.0.0/0 (което означава всички адреси), а шлюзът трябва да бъде IP адресът на `Router1`, който е част от същата подмрежа. Благодарение на Longest Prefix Match, адресите на местоназначение в локалната подмрежа все още ще бъдат достъпни директно, но всички други дестинации ще отидат до шлюза по подразбиране.

Тъй като това устройство е предназначено да бъде сървър, а не рутер, то не трябва да участва в RIP мрежата.

Можете да използвате опцията от менюто „Change Symbol“, ако искате мрежовата ви диаграма да показва символ на сървър вместо символ на рутер.

Mikrotik Router като DHCP сървър

Вместо да създаваме друга виртуална машина на Linux и да инсталираме DHCP сървър в нея, просто ще използваме съществуващия рутер MikroTik като наш примерен DHCP сървър за вътрешната подмрежа. В крайна сметка, той вече има инсталиран DHCP сървър.

1. Задайте на "сървъра" име на хоста (`DHPServer`)
2. Задайте на "сървъра" IP адрес
3. Задайте на "сървъра" статичен маршрут по подразбиране, така че да изпраща целия трафик към `Router1`

Тъй като това устройство е предназначено да бъде сървър, а не рутер, то не трябва да участва в RIP мрежата.

Можете да използвате опцията от менюто „Change Symbol“, ако искате мрежовата ви диаграма да показва символ на сървър вместо символ на рутер.

Dynamic Host Configuration Protocol (DHCP) Relay

В тази лабораторna, устройствата в подмрежа 5 трябва да могат да получат мрежовата си конфигурация чрез DHCP. DHCP сървърът обаче не е `Router2`, който е директно свързан към подмрежата. Вместо това DHCP сървърът се намира в подмрежа 2. Следователно е необходимо DHCP препредаване(**Relay**)

Първо, конфигурирайте `Router2` да функционира като DHCP предавател. Когато `Router2` получи DHCP заявка на посочения интерфейс, той ще я препрати към `DHCPServer`.

```
ip dhcp-relay add interface=ether1 dhcp-server=10.0.30.YY name=relay1 disabled=no
ip dhcp-relay print
```

Второ, конфигурирайте `DHCPServer` да присвоява адреси за подмрежа 5.

1. Създайте DHCP pool с набор от IP адреси, които да раздавате на клиентите. Изключите адреси, които се използват от самия рутер или всякакви други статично конфигурирани мрежови устройства в тази подмрежа. (`ip pool add`....)
2. Активирайте DHCP сървър на конкретен интерфейс, като използвате конкретен диапазон от IP адреси и предоставяте „поднаеми“ на IP адреси за определен период от време. Задайте `disabled=no`, за да активирате този сървър. (`ip dhcp-server add interface`...). Имайте предвид, че за тази команда трябва да добавите един допълнителен аргумент в сравнение с последната лабораторна работа. Аргументът relay трябва да се добави и да посочи IP адреса, откъдето са прихванати тези DHCP заявки. В нашия случай `10.0.40.???` (интерфейсът на `Router2`, който е свързан към подмрежа 5).

3. Конфигурирайте DHCP да изпраща информацията за подмрежата, желаните DNS сървъри (8.8.8.8, 8.8.4.4) и шлюза по подразбиране към клиентите.

Тестване на мрежата преди FireWall

За тестване:

1. Уверете се, че webterm-workstation е получила IP адрес чрез DHCP. (Изпълнете `ip dhcp-server lease print` на `DHCPServer`)
2. Уверете се, че webterm-public може успешно да осъществява ping до `WebServer`, `DHCPServer` и `webterm-workstation`.
3. Уверете се, че webterm-public може да зареди уеб страницата на уеб сървъра (http://10.0.20.1)

FireWall

Сега ще конфигурирате защитна стена на `Router1` за мрежата. Защитната стена работи посредством правилата на защитната стена. Всяко правило се състои от две части – съвпадение, което съпоставя трафика с дадени условия и действието, което дефинира какво да се прави със съвпадащия пакет, като например разрешаване или отказ.

Правилата за филтриране на защитната стена са групирани заедно във вериги. Това позволява на един пакет да се съпостави с един общ критерий в една верига и след това да се предаде за обработка спрямо други общи критерии към друга верига. Има пет предварително дефинирани вериги,:

1. **FORWARD** – обработва пакетите, преминаващи през възел, използван за шлюз (gateway), пристигащи на един мрежов интерфейс и излизащи веднага през друг;

2. **INPUT** – обработва мрежовите пакети точно преди да бъдат доставени на локалния процес, за който са предназначени; . С други думи, пакети, при които IP адресът на местоназначението е един от адресите на рутера (или конкретното устройство). Пакетите, преминаващи през рутера, не се обработват срещу правилата на входната верига.

3. **OUTPUT** – обработва мрежовите пакети веднага след генерирането им от някой локален процес. Пакетите, преминаващи през рутера, не се обработват срещу правилата на изходната верига.

4. **POSTROUTING** – обработва мрежовите пакети точно преди да излязат през някой мрежови интерфейс;

5. **PREROUTING** – обработва всички мрежови пакети веднага след пристигането им през някой от мрежовите интерфейси (след из-хвърлянето на всякакви пакети в резултат на това, че интерфейса работи в нефилтриращ режим и след проверка на контролните суми).

Изборът на верига се основава на това, в коя част от жизнения цикъл на пакетите искаме да приложим своите правила. Филтрирането на изходящите пакети се прави във веригата OUTPUT, тъй като веригата POSTROUTING не е асоциирана с таблицата за филтриране filter.

В днешната лаборатория ще използваме само Forward веригата.

Когато обработвате верига, правилата се вземат от веригата в реда, в който са изброени отгоре надолу. Ако даден пакет отговаря на критериите на правилото, тогава посоченото действие се изпълнява върху него и повече правила не се обработват в тази верига. Ако даден пакет не отговаря на никое правило във вградената

верига се изпълнява общата политика за тази верига(в повечето случаи се приема accepted. Политиката може да се променя).

Създайте правила за защитни стени на `Router1`, които изпълняват следните действия:

1. Входящите HTTP заявки от подмрежа 4 към уеб сървъра трябва да бъдат разрешени:
```
ip firewall filter add chain=forward action=accept protocol=tcp port=80 connection-state=new src-address=172.16.0.0/24 dst-address=10.0.20.YY comment="Permit Web Server in DMZ"
```
2. Пакетите, които са част от установените връзки (или свързаните), трябва да бъдат разрешени. Това позволява отговорът от уеб сървъра да се върне обратно към искащия клиент:
```
ip firewall filter add chain=forward action=accept connection-state=established,related comment="Accept established connections"
```
3. Всичко друго трябва да се забрани:
```
ip firewall filter add chain=forward action=drop comment="Default deny"
```

Вижте веригата на защитната си стена след конфигурирането с `ip firewall filter print chain=forward`. Всяко правило на защитната стена трябва да бъде в посочения по-горе ред. Първо разрешавате преминаването на конкретни нови връзки, след това разрешавате всички установени и свързани връзки да преминават и накрая забранявате всичко друго, което не е специално разрешено по-горе.

Тествайте вашата мрежа

1. Може ли вашият webterm-public да има достъп до уеб страницата на `WebServer`? (http://10.0.20.YY)
Трябва да успее - изрично разрешено от forward веригата
2. Може ли вашият webterm-public да има достъп до уеб страницата на `DHCPServer`? (http://10.0.30.YY)
Трябва да се провали - по подразбиране за отказване от forward веригата
3. Може ли вашият webterm-public да има достъп до уеб страницата на `Router1`? (http://172.16.0.???)
Трябва да успее - входната верига все още не е конфигурирана (и е разрешена по подразбиране)
(Можете също да получите достъп до рутера на IP 10.0.20.???, 10.0.30.??? и т.н.... и всичко ще работи, тъй като всички тези IP адреси на местоназначение ще бъдат обработени от input веригата)

Нека добавим няколко правила относно подмрежа 2 за вътрешните сървъри. Както е в момента, защитната стена блокира DHCP релето!

Създайте правило за защитна стена на `Router1`, което изпълнява следните действия:

1. Входящите DHCP заявки (UDP порт 67) от подмрежа 3 към DHCPServer трябва да бъдат разрешени:
```
ip firewall filter add chain=forward action=accept protocol=udp port=67 connection-state=new src-address=192.168.0.0/30 dst-address=10.0.30.YY comment="Permit DHCP Relay to Internal Server"
```
2. Изходящите DHCP отговори от DHCPServer към подмрежа 5 трябва да бъдат разрешени:
```
ip firewall filter add chain=forward action=accept protocol=udp port=67 connection-state=new src-address=10.0.30.YY dst-address=10.0.40.0/24 comment="Permit DHCP Response"
```

На вашата webterm-workstation, опитайте ръчно да стартирате отново DHCP клиента, за да проверите дали DHCP работи:

```
/tmp/gns3/bin/udhcpc -v
Sending discover...
Sending discover...
```

```
Sending discover...
(CTRL-C to exit)
```

Нещата **не** работят…. Вижте веригата на защитната си стена след конфигурирането с `ip firewall filter print chain=forward`. Забелязвате ли проблема?
Правилата на защитната стена, които току-що добавихте, са **след** правилото за отказ по подразбиране и по този начин никога няма да имат ефект. Преместете ги по-нагоре в списъка, над установените връзки и правилата за отказ по подразбиране: `ip firewall filter move numbers=3,4 destination=1`

Сега опитайте ръчно да стартирате отново DHCP клиента:

```
/tmp/gns3/bin/udhcpc -v
Sending discover...
Sending select for 10.0.40.???...
Lease of 10.0.40.??? obtained, lease time 86400
```