

Lista 4

Dominik Kaczmarek

Zad 1

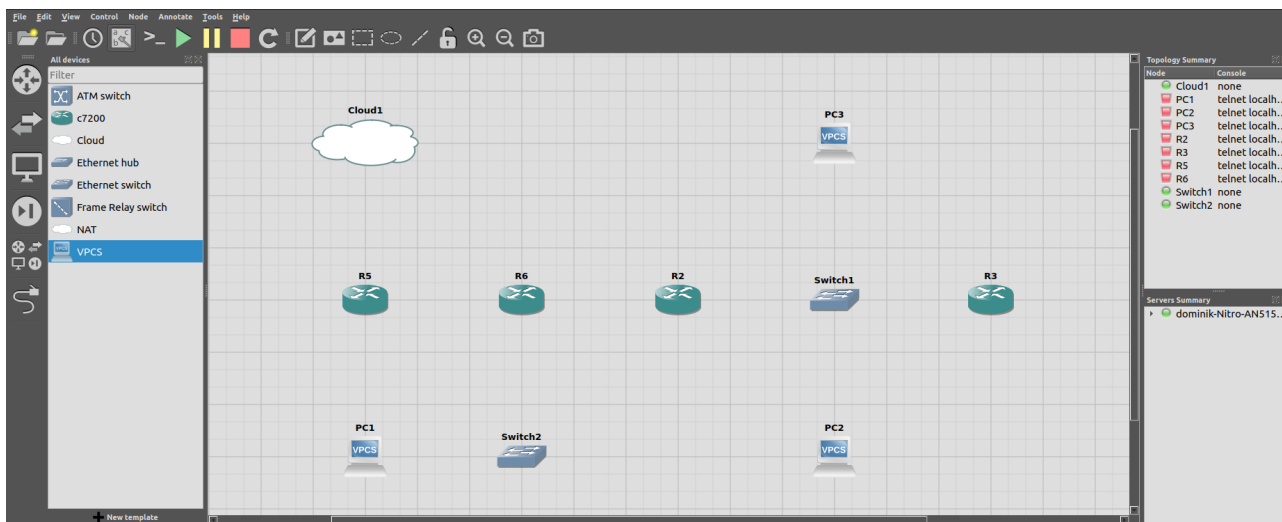
1. W symulatorze GNS3 skonfiguruj wirtualną sieć o podanej topologii, tak aby:

1. Wirtualna sieć była połączona z zewnętrzną ('fizyczną') siecią 'Cloud'.
2. Ruter R5 uzyskiwał dynamiczny adres IP z sieci 'Cloud'.
3. Pozostałe urządzenia posiadały statyczne adresy w swoich sieciach.
4. Możliwe było wysyłanie komunikatów "ping" pomiędzy dowolną parą urządzeń sieci wirtualnej.
5. Możliwe było wysyłanie komunikatów "ping" z dowolnego urządzenia w sieci wirtualnej na zewnętrzny adres, np. 'google.com'.

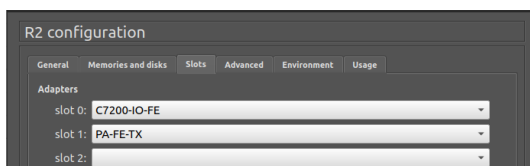
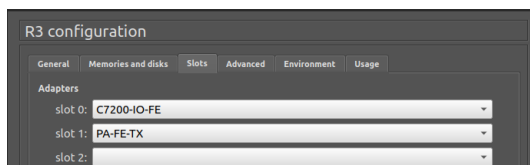
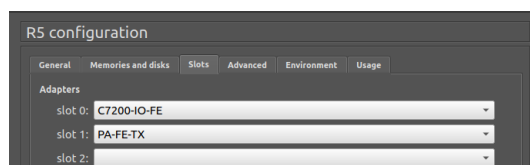
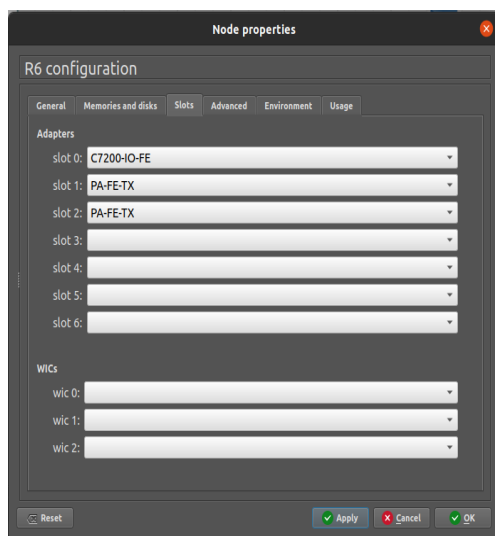
Po pobraniu programu gns3, dodatkowo należało ściągnąć z internetu obraz rutera. Ja pobrałem obraz rutera c7200 ze strony <https://www.sysnettechsolutions.com/en/cisco-ios-download-for-gns3/>.

| | | | | |
|-------|----|---------------------------------------|--------|----------|
| c7200 | 15 | c7200-adventerprisek9-mz.152-4.M7.bin | 512 MB | Download |
|-------|----|---------------------------------------|--------|----------|

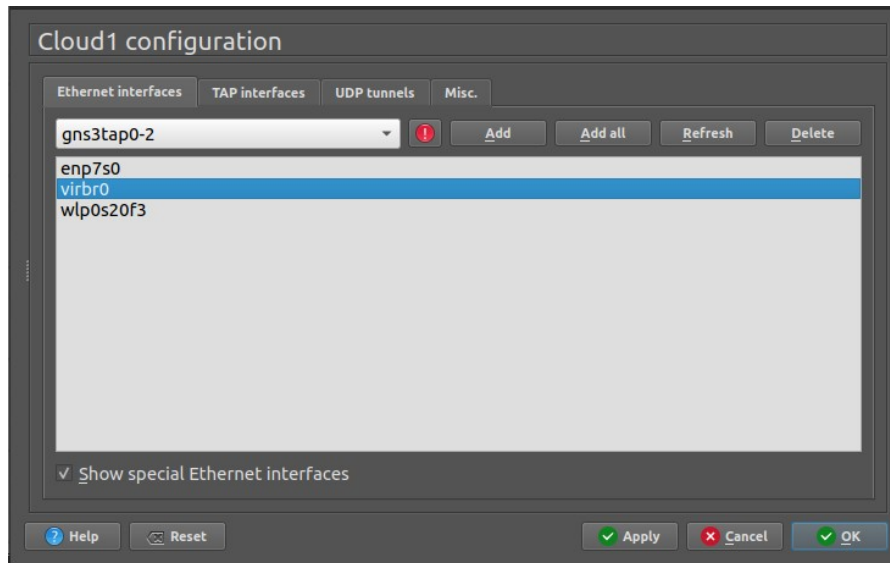
Mając ściągnięty ruter ustawiłem i ponazywałem wszystkie komponenty przykładowej sieci na wzór przykładu umieszczonego na stronie internetowej dr Krzywieckiego.



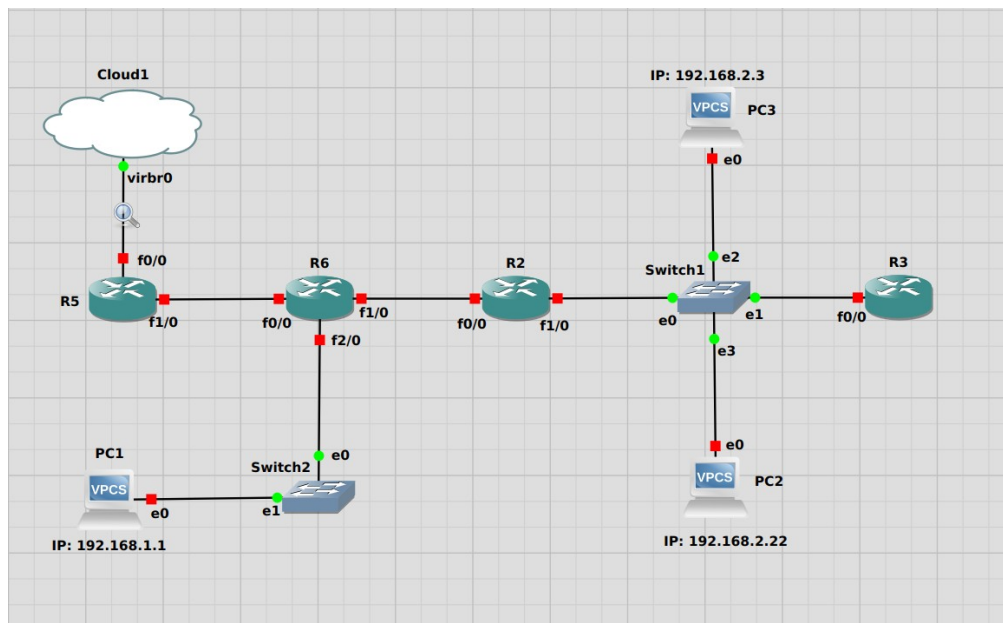
Kolejnym krokiem było skonfigurowanie wszystkich routerów. Najpierw ustawiłem ilość wejść/wyjść dla każdego z nich.



Laptop na którym przeprowadzałem symulacje był podpięty do mojej sieci domowej kablem internetowym. Przy konfiguracji clouda wybrałem interfejs **virbr0**.



Następnie podpiąłem wszystkie urządzenia kablami w następujący sposób.



Kiedy układ był już podłączony należało skonfigurować po kolei każde z urządzeń używając ich konsol. Żeby to zrobić najpierw odpaliłem cały układ przyciskiem „Start/Resume all devices”.



Następnie skonfigurowałem wszystkie routery. Na kolejnych stronach zamieściłem komendy których używałem do konfiguracji każdego z routerów, oraz zdjęcia terminala po wywołaniu tych komend.

Ruter R5

Komendy użyte do konfiguracji rutera

```
1 ~~~ Router R5 ~~~
2 conf t
3 int f0/0
4 ip address dhcp
5 ip nat outside
6 no shut
7 end
8
9 conf t
10 ip domain-lookup
11 ip name-server 8.8.8.8
12 end
13
14 conf t
15 int f1/0
16 ip add 192.168.3.6 255.255.255.0
17 ip nat inside
18 no shut
19 end
20
21 conf t
22 router rip
23 version 2
24 no auto-summary
25 network 192.168.42.0
26 network 192.168.3.0
27 default-information originate
28 end
29
30 conf t
31 access-list 10 permit 192.168.1.0 0.0.254.255
32 access-list 10 permit 192.168.2.0 0.0.253.255
33 access-list 10 permit 192.168.3.0 0.0.252.255
34 access-list 10 permit 192.168.4.0 0.0.251.255
35
36 ip nat inside source list 10 interface f0/0 overload
37 end
38
39 wr
40
```

Zrzut ekranu z terminala po wpisaniu wspomnianych komend

```
R5# conf t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#int f0/0
R5(config-if)#ip address dhcp
R5(config-if)#ip nat outside
R5(config-if)#no shut
R5(config-if)#end
R5#
R5#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#ip domain-lookup
R5(config)#ip name-server 8.8.8.8
R5(config)#end
R5#
R5#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#int f1/0
R5(config-if)#ip add 192.168.3.6 255.255.255.0
R5(config-if)#ip nat inside
R5(config-if)#no shut
R5(config-if)#end
R5#
R5#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#router rip
R5(config-router)#version 2
R5(config-router)#no auto-summary
R5(config-router)#network 192.168.42.0
R5(config-router)#network 192.168.3.0
R5(config-router)#default-information originate
R5(config-router)#end
R5#
R5#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#access-list 10 permit 192.168.1.0 0.0.254.255
R5(config)#access-list 10 permit 192.168.2.0 0.0.253.255
R5(config)#access-list 10 permit 192.168.3.0 0.0.252.255
R5(config)#access-list 10 permit 192.168.4.0 0.0.251.255
R5(config)#
R5(config)#ip nat inside source list 10 interface f0/0 overload
R5(config)#end
R5#
R5#wr
*May 31 14:20:42.231: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
*May 31 14:20:42.447: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:20:42.495: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:20:42.611: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:20:42.659: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:20:42.703: %SYS-5-CONFIG I: Configured from console by console
R5#wr
Building configuration...
[OK]
R5#
```

Ruter R6

Komendy użyte do konfiguracji rutera

```
1 ~~~ Router R6 ~~~
2 conf t
3 int f0/0
4 no shut
5 ip add 192.168.3.5 255.255.255.0
6 end
7
8 conf t
9 ip domain-lookup source-interface f0/0
10 ip name-server 8.8.8.8
11 end
12
13 conf t
14 int f1/0
15 no shut
16 ip add 192.168.4.2 255.255.255.0
17 end
18
19 conf t
20 int f2/0
21 no shut
22 ip add 192.168.1.11 255.255.255.0
23 end
24
25 conf t
26 router rip
27 version 2
28 no auto-summary
29 network 192.168.1.0
30 network 192.168.4.0
31 network 192.168.3.0
32 end
33
34 wr
35
```

Zrzut ekranu z terminala po wpisaniu wspomnianych komend

```
R6#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R6(config)#int f0/0
R6(config-if)#no shut
R6(config-if)#ip add 192.168.3.5 255.255.255.0
R6(config-if)#end
R6#
R6#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R6(config)#ip domain-lookup source-interface f0/0
R6(config)#ip name-server 8.8.8.8
R6(config)#end
R6#
R6#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R6(config)#int f1/0
R6(config-if)#no shut
R6(config-if)#ip add 192.168.4.2 255.255.255.0
R6(config-if)#end
R6#
R6#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R6(config)#int f2/0
R6(config-if)#no shut
R6(config-if)#ip add 192.168.1.11 255.255.255.0
R6(config-if)#end
R6#
R6#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R6(config)#router rip
R6(config-router)#version 2
R6(config-router)#no auto-summary
R6(config-router)#network 192.168.1.0
R6(config-router)#network 192.168.4.0
R6(config-router)#network 192.168.3.0
R6(config-router)#end
*May 31 14:44:50.939: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:44:50.959: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:44:50.967: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:44:50.975: %SYS-5-CONFIG I: Configured from console by console
R6(config-router)#end
R6#
*May 31 14:44:52.819: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*May 31 14:44:52.959: %LINK-3-UPDOWN: Interface FastEthernet1/0, changed state to up
*May 31 14:44:52.971: %LINK-3-UPDOWN: Interface FastEthernet2/0, changed state to up
R6#
*May 31 14:44:53.667: %SYS-5-CONFIG I: Configured from console by console
*May 31 14:44:53.819: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
*May 31 14:44:53.959: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to up
*May 31 14:44:53.971: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet2/0, changed state to up
R6#wr
Building configuration...
[OK]
```

Ruter R2

Komendy użyte do konfiguracji rutera

```
1 ~~~ Router R2 ~~~
2 conf t
3 int f1/0
4 ip add 192.168.2.1 255.255.255.0
5 no shut
6 int f0/0
7 ip add 192.168.4.6 255.255.255.0
8 no shut
9 end
10
11 conf t
12 router rip
13 version 2
14 no auto-summary
15 network 192.168.2.0
16 network 192.168.4.0
17 end
18
19 conf t
20 ip domain-lookup
21 ip name-server 8.8.8.8
22 end
23
24 wr
25
```

Zrzut ekranu z terminala po wpisaniu wspomnianych komend

```
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int f1/0
R2(config-if)#ip add 192.168.2.1 255.255.255.0
R2(config-if)#no shut
R2(config-if)#int f0/0
R2(config-if)#ip add 192.168.4.6 255.255.255.0
R2(config-if)#no shut
R2(config-if)#end
R2#
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.2.0
R2(config-router)#network 192.168.4.0
R2(config-router)#end
R2#
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip domain-lookup
R2(config)#ip name-server 8.8.8.8
R2(config)#end
*May 31 14:57:23.591: %SYS-5-CONFIG_I: Configured from console by console
*May 31 14:57:23.627: %SYS-5-CONFIG_I: Configured from console by console
R2(config)#end
R2#
*May 31 14:57:31.663: %SYS-5-CONFIG_I: Configured from console by console
R2#wr
Building configuration...
[OK]
R2#
```

Ruter R3

Komendy użyte do konfiguracji rutera

```
1 ~~~ Router R3 ~~~
2 conf t
3 int f0/0
4 ip add 192.168.2.2 255.255.255.0
5 no shut
6 end
7
8 conf t
9 router rip
10 version 2
11 no auto-summary
12 network 192.168.2.0
13 end
14
15 conf t
16 ip domain-lookup
17 ip name-server 8.8.8.8
18 end
19
20 wr
21
```

Zrzut ekranu z terminala po wpisaniu wspomnianych komend

```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#int f0/0
R3(config-if)#ip add 192.168.2.2 255.255.255.0
R3(config-if)#no shut
R3(config-if)#end
R3#
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto-summary
R3(config-router)#network 192.168.2.0
R3(config-router)#end
R3#
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#ip domain-lookup
R3(config)#ip name-server 8.8.8.8
R3(config)#end
*May 31 15:05:35.239: %SYS-5-CONFIG_I: Configured from console by console
*May 31 15:05:35.307: %SYS-5-CONFIG_I: Configured from console by console
R3(config)#end
*May 31 15:05:37.191: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*May 31 15:05:38.191: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R3(config)#end
R3#
*May 31 15:05:59.827: %SYS-5-CONFIG_I: Configured from console by console
R3#wr
Building configuration...
[OK]
```

Używając komendy „ping google.com” można sprawdzić czy ruter R3 został poprawnie podłączony do sieci.

```
R3#ping google.com

Translating "google.com"...domain server (8.8.8.8) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 142.250.186.206, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/40/48 ms
R3#
```


Ostatnim krokiem było ustawienie trzech komputerów: PC1, PC2 i PC3. Poniżej zamieściłem zdjęcia terminala z użytymi komendami i testem *ping google.com*.

PC1

```
PC1> ip 192.168.1.1/24 192.168.1.11
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.11

PC1> ip dns 8.8.8.8

PC1> wr
Saving startup configuration to startup.vpc
. done

PC1> ping google.com
google.com resolved to 216.58.215.78

84 bytes from 216.58.215.78 icmp_seq=1 ttl=116 time=29.058 ms
84 bytes from 216.58.215.78 icmp_seq=2 ttl=116 time=24.853 ms
84 bytes from 216.58.215.78 icmp_seq=3 ttl=116 time=24.998 ms
84 bytes from 216.58.215.78 icmp_seq=4 ttl=116 time=25.075 ms
84 bytes from 216.58.215.78 icmp_seq=5 ttl=116 time=25.030 ms

PC1> |
```

PC2

```
PC2> ip 192.168.2.22/24 192.168.2.1
Checking for duplicate address...
PC2 : 192.168.2.22 255.255.255.0 gateway 192.168.2.1

PC2> ip dns 8.8.8.8

PC2> wr
Saving startup configuration to startup.vpc
. done

PC2> ping google.com
google.com resolved to 216.58.215.78

84 bytes from 216.58.215.78 icmp_seq=1 ttl=115 time=39.232 ms
84 bytes from 216.58.215.78 icmp_seq=2 ttl=115 time=34.689 ms
84 bytes from 216.58.215.78 icmp_seq=3 ttl=115 time=35.427 ms
84 bytes from 216.58.215.78 icmp_seq=4 ttl=115 time=35.062 ms
84 bytes from 216.58.215.78 icmp_seq=5 ttl=115 time=34.449 ms

PC2> |
```

PC3

```
PC3> ip 192.168.2.3/24 192.168.2.1
Checking for duplicate address...
PC3 : 192.168.2.3 255.255.255.0 gateway 192.168.2.1

PC3> ip dns 8.8.8.8

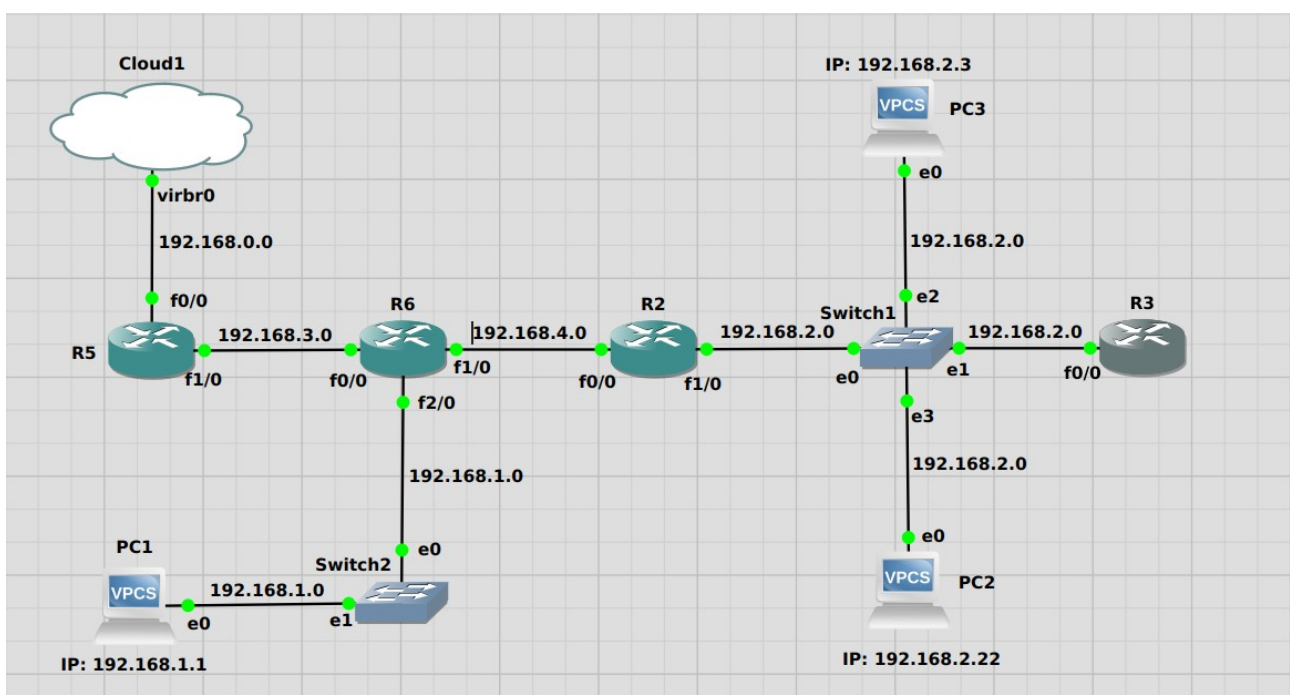
PC3> wr
Saving startup configuration to startup.vpc
. done

PC3> ping google.com
google.com resolved to 216.58.215.110

84 bytes from 216.58.215.110 icmp_seq=1 ttl=115 time=49.691 ms
84 bytes from 216.58.215.110 icmp_seq=2 ttl=115 time=44.586 ms
84 bytes from 216.58.215.110 icmp_seq=3 ttl=115 time=34.697 ms
84 bytes from 216.58.215.110 icmp_seq=4 ttl=115 time=65.464 ms
84 bytes from 216.58.215.110 icmp_seq=5 ttl=115 time=34.720 ms

PC3> |
```

Ostateczny schemat naszej sieci wygląda następująco



W zadaniu 1 pozostało mi jeszcze sprawdzenie czy powiedzie się przesłanie pakietów pomiędzy dwoma urządzeniami z naszej sieci.

1. W pierwszym teście korzystając z PC1 spróbuję „spingować” PC2 (IP: 192.168.2.22) i zobaczę czy pakiety dotrą z powrotem.

```
PC1> ping 192.168.2.22

84 bytes from 192.168.2.22 icmp_seq=1 ttl=62 time=48.808 ms
84 bytes from 192.168.2.22 icmp_seq=2 ttl=62 time=24.650 ms
84 bytes from 192.168.2.22 icmp_seq=3 ttl=62 time=24.653 ms
84 bytes from 192.168.2.22 icmp_seq=4 ttl=62 time=24.750 ms
84 bytes from 192.168.2.22 icmp_seq=5 ttl=62 time=24.839 ms

PC1> |
```

Jak widać wszystko poszło zgodnie z planem, ponieważ pakietom udało się dotrzeć w zamierzone miejsce. Ponadto ttl wynosi w tym przypadku 62 czyli (wykorzystując wiedzę z listy 1 oraz matematykę ($64-62=2$)) wychodzi na to, że pakiety przeszły po drodze przez dwa urządzenia. Spoglądając jeszcze raz na schemat naszej sieci, widać że wszystko się zgadza, ponieważ pakiety przeszły kolejno przez routery R6 i R2.

2. W drugim teście sprawdzę jak wygląda komunikacja między dwoma routerami. W tym celu spróbuję wysłać pakiet z R5 do R3. Jak widać na poniższym zrzucie ekranu test się powiódł.

```
R5#ping 192.168.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/28/32 ms
R5#|
```

Zad 2 i 3

2. Ustaw przechwytywanie komunikatów w sieciach: 192.168.0.0, 192.168.2.0, 192.168.3.0.

3. Przeanalizuj przechwycone komunikaty dla zapytania wysłanego z komputera PC2: 'ping google.com'.

192.168.0.0 (Cloud ↔ R5)

| * [Cloud1 virbr0 to R5 FastEthernet0/0] | | | | | |
|---|------------|-----------------|-----------------|----------|--|
| Plik Edytuj Widok Idź Przechwytyj Analizuj Statystyki Telefonia Bezprzewodowe Narzędzia Pomoc | | | | | |
| icmp or dns | | | | | |
| No. | Time | Source | Destination | Protocol | Length Info |
| 157 | 222.061477 | 192.168.122.104 | 8.8.8.8 | DNS | 70 Standard query 0x849a A google.com |
| 158 | 222.070165 | 8.8.8.8 | 192.168.122.104 | DNS | 86 Standard query response 0x849a A google.com A 216.58.215.78 |
| 159 | 222.101736 | 192.168.122.104 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa56c, seq=1/256, ttl=61 (reply in 160) |
| 160 | 222.112309 | 216.58.215.78 | 192.168.122.104 | ICMP | 98 Echo (ping) reply id=0xa56c, seq=1/256, ttl=118 (request in 159) |
| 162 | 223.147473 | 192.168.122.104 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa66c, seq=2/512, ttl=61 (reply in 163) |
| 163 | 223.156373 | 216.58.215.78 | 192.168.122.104 | ICMP | 98 Echo (ping) reply id=0xa66c, seq=2/512, ttl=118 (request in 162) |
| 165 | 224.183996 | 192.168.122.104 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa76c, seq=3/768, ttl=61 (reply in 166) |
| 166 | 224.191751 | 216.58.215.78 | 192.168.122.104 | ICMP | 98 Echo (ping) reply id=0xa76c, seq=3/768, ttl=118 (request in 165) |
| 167 | 225.218710 | 192.168.122.104 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa86c, seq=4/1024, ttl=61 (reply in 168) |
| 168 | 225.226138 | 216.58.215.78 | 192.168.122.104 | ICMP | 98 Echo (ping) reply id=0xa86c, seq=4/1024, ttl=118 (request in 167) |
| 170 | 226.254386 | 192.168.122.104 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xaa6c, seq=5/1280, ttl=61 (reply in 171) |
| 171 | 226.261758 | 216.58.215.78 | 192.168.122.104 | ICMP | 98 Echo (ping) reply id=0xaa6c, seq=5/1280, ttl=118 (request in 170) |

192.168.3.0 (R5 ↔ R6)

| * [R5 FastEthernet1/0 to R6 FastEthernet0/0] | | | | | |
|---|------------|---------------|---------------|----------|---|
| Plik Edytuj Widok Idź Przechwytyj Analizuj Statystyki Telefonia Bezprzewodowe Narzędzia Pomoc | | | | | |
| icmp or dns | | | | | |
| No. | Time | Source | Destination | Protocol | Length Info |
| 36 | 109.439861 | 192.168.2.22 | 8.8.8.8 | DNS | 70 Standard query 0x849a A google.com |
| 37 | 109.458376 | 8.8.8.8 | 192.168.2.22 | DNS | 86 Standard query response 0x849a A google.com A 216.58.215.78 |
| 38 | 109.480111 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa56c, seq=1/256, ttl=62 (reply in 39) |
| 39 | 109.508690 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa56c, seq=1/256, ttl=117 (request in 38) |
| 42 | 110.525724 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa66c, seq=2/512, ttl=62 (reply in 43) |
| 43 | 110.544369 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa66c, seq=2/512, ttl=117 (request in 42) |
| 45 | 111.562393 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa76c, seq=3/768, ttl=62 (reply in 46) |
| 46 | 111.580069 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa76c, seq=3/768, ttl=117 (request in 45) |
| 47 | 112.597990 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa86c, seq=4/1024, ttl=62 (reply in 48) |
| 48 | 112.615599 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa86c, seq=4/1024, ttl=117 (request in 47) |
| 49 | 113.635508 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xaa6c, seq=5/1280, ttl=62 (reply in 50) |
| 50 | 113.654284 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xaa6c, seq=5/1280, ttl=117 (request in 49) |

Frame 48: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0

Ethernet II, Src: ca:03:31:f7:00:1c (ca:03:31:f7:00:1c), Dst: ca:04:32:07:00:00 (ca:04:32:07:00:00)

Internet Protocol Version 4, Src: 216.58.215.78, Dst: 192.168.2.22

Internet Control Message Protocol

192.168.2.0 (R2 ↔ Switch1)

| *- [R2 FastEthernet1/0 to Switch1 Ethernet0] | | | | | |
|---|------------|---------------|---------------|----------|---|
| Plik Edytuj Widok Idź Przechwytyj Analizuj Statystyki Telefonia Bezprzewodowe Narzędzia Pomoc | | | | | |
| icmp or dns | | | | | |
| No. | Time | Source | Destination | Protocol | Length Info |
| 31 | 160.987140 | 192.168.2.22 | 8.8.8.8 | DNS | 70 Standard query 0x849a A google.com |
| 32 | 161.027117 | 8.8.8.8 | 192.168.2.22 | DNS | 86 Standard query response 0x849a A google.com A 216.58.215.78 |
| 33 | 161.028197 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa56c, seq=1/256, ttl=64 (reply in 34) |
| 34 | 161.077413 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa56c, seq=1/256, ttl=115 (request in 33) |
| 36 | 162.077847 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa66c, seq=2/512, ttl=64 (reply in 37) |
| 37 | 162.112930 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa66c, seq=2/512, ttl=115 (request in 36) |
| 38 | 163.113409 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa76c, seq=3/768, ttl=64 (reply in 39) |
| 39 | 163.148613 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa76c, seq=3/768, ttl=115 (request in 38) |
| 40 | 164.149315 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xa86c, seq=4/1024, ttl=64 (reply in 41) |
| 41 | 164.184034 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xa86c, seq=4/1024, ttl=115 (request in 40) |
| 42 | 165.185208 | 192.168.2.22 | 216.58.215.78 | ICMP | 98 Echo (ping) request id=0xaa6c, seq=5/1280, ttl=64 (reply in 43) |
| 43 | 165.219486 | 216.58.215.78 | 192.168.2.22 | ICMP | 98 Echo (ping) reply id=0xaa6c, seq=5/1280, ttl=115 (request in 42) |

Wnioski:

- Przez każdą z badanych sieci tj. 192.168.0.0, 192.168.2.0, 192.168.3.0 przeszło w sumie 10 „pingów”, 5 wysłanych „requestów” oraz 5 wiadomości zwrotnych „replayów”, czyli wszystko poszło zgodnie z planem.
- W każdej sieci zarówno *Source* jak i *Destiantion* miały te same adresy ip. (Nic dziwnego)
- Z ciekawszych obserwacji, możemy zauważyć, że im bliżej sprawdzana sieć znajduje się komputera PC2 tym większa wartość ttl, przy czym dla sieci 192.168.2.0 wartość ttl wynosi 64, co oznacza że między tą siecią a PC2 nie znajdował się ani PC ani router. Dla porównania spoglądając na informacje o pakietach w sieci 192.168.0.0 wartość ttl wynosi 61. Równanie $64 - 61 = x$ daje nam wynik $x = 3$, co wskazuje że pakiety podróżując od PC2 w kierunku Clouda napotkały na swojej drodze 3 rutery/PC. Spoglądając na schemat faktycznie tak było, gdyż pomiędzy Cloudem a PC2 znajdują się rutery R5, R6 i R2.
- Przeprowadzając podobną analizę jak we wniosku 3 widać, że im bliżej sieć znajduje się Clouda tym wartość ttl przy pakietach powracających (*reply*) jest większa, zatem droga przebyta od serwera *google.com* do obecnej sieci jest *mniej*sza.
- Do ttl nie są zliczane urządzenia typu *Switch*.