Sprawozdanie z listy 4. - Technologie sieciowe

Jakub Bachanek

25 maja 2022

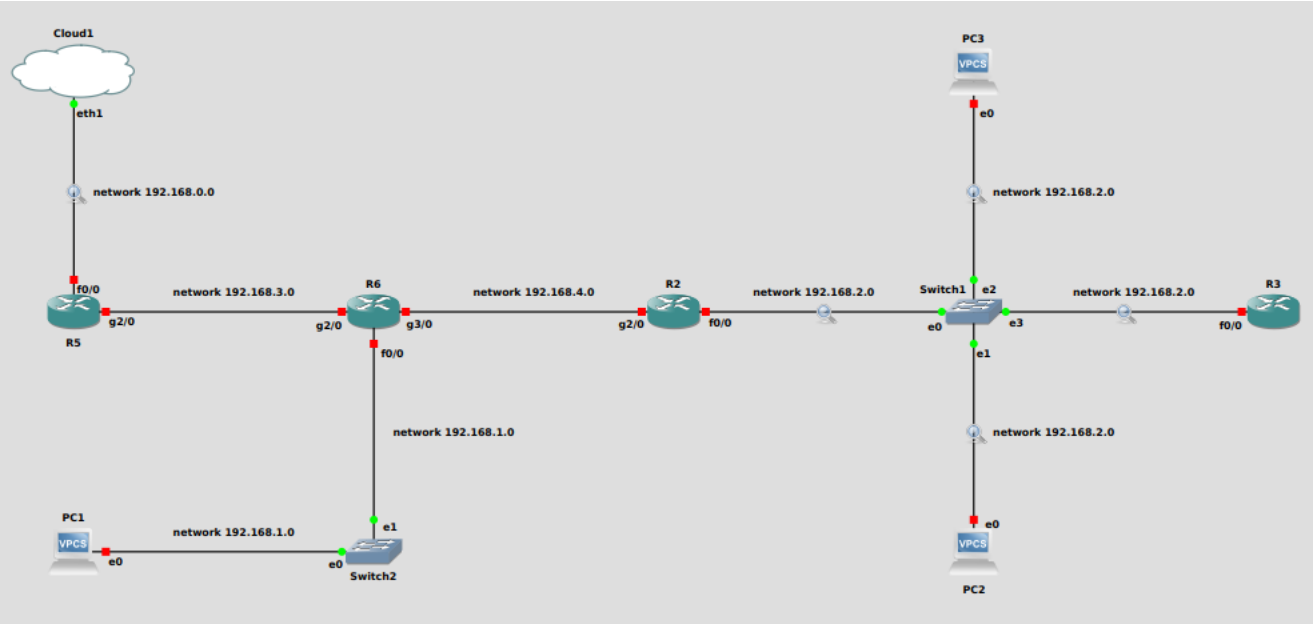
1 GNS3

1.1 Opis programu

GNS3 (Graphical Network Simulator) to emulator sieci, który pozwala na projektowanie i analizę działania topologii o różnej złożoności. W graficznym interfejsie mamy możliwość wyboru wirtualnych urządzeń, stworzenia połączeń oraz kompleksowej konfiguracji. Program pozwala na użycie obrazów prawdziwych systemów operacyjnych routerów, przez co zachowanie sieci jest wiernie odtworzone.

1.2 Model sieci

W zadaniu utworzony został następujący model sieci:



Składa się on z czterech routerów *Cisco 7200*, dwóch switchy, trzech komputerów PC oraz chmury. Router *R5* uzyskuje dynamiczny adres IP z sieci *Cloud*, a pozostałe urządzenia posiadają statyczne adresy IP w swoich sieciach.

1.3 Konfiguracja sieci

Każde urządzenie możemy dokładnie skonfigurować poprzez konsolę, do której mamy pełny dostęp.

1.3.1 Dynamiczny adres IP

W celu uzyskiwania dynamicznego adresu IP z chmury, w *R5* ustawiamy:

```
int fa0/0
ip address dhcp
ip nat outside
no shut
```

1.3.2 Statyczne adresy IP

Przydzielanie statycznych adresow IP odbywa się ręcznie:

Na przykład dla *R2*:

```
int fa0/0
ip address 192.168.2.1 255.255.255.0
no shut

int g2/0
ip address 192.168.4.2 255.255.255.0
no shut
```

1.3.3 Wyszukiwanie DNS

Aby możliwe było użycie polecenia *ping* z nazwą hosta należy poustawiać:

Dla *R6*:

```
ip domain lookup source-interface g2/0
ip name-server 8.8.8.8
```

Dla innych:

```
ip domain-lookup
ip name-server 8.8.8.8
```

1.3.4 Protokół routingu RIP

W celu włączenia procesu routingu *RIP* i powiązania z nim sieci:

Dla *R5*:

```
router rip
version 2
no auto-summary
network 192.168.0.0
network 192.168.3.0
default-information originate
```

1.3.5 Reguły filtrujące

Żeby ustawić reguły filtrujące ruch z sieci, dla *R5* należy:

```
int g2/0
ip nat inside
ip nat inside source list 10 interface fa0/0 overload
access-list 10 permit 192.168.1.0 0.0.254.255
access-list 10 permit 192.168.2.0 0.0.253.255
access-list 10 permit 192.168.3.0 0.0.252.255
access-list 10 permit 192.168.4.0 0.0.251.255
```

1.4 Ping dwóch urządzeń

Każdą parę urządzeń w sieci można pingować. Poniżej przykłady:

PC2 z *PC1*:

```
PC2> ping 192.168.1.2

84 bytes from 192.168.1.2 icmp_seq=1 ttl=62 time=67.388 ms
84 bytes from 192.168.1.2 icmp_seq=2 ttl=62 time=21.719 ms
84 bytes from 192.168.1.2 icmp_seq=3 ttl=62 time=26.964 ms
```

PC2 z routerem *R6*:

```
PC2> ping 192.168.4.1

84 bytes from 192.168.4.1 icmp_seq=1 ttl=254 time=20.238 ms
84 bytes from 192.168.4.1 icmp_seq=2 ttl=254 time=18.322 ms
84 bytes from 192.168.4.1 icmp_seq=3 ttl=254 time=19.488 ms
```

router *R5* z routerem *R2*:

```
R5# ping 192.168.4.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.4.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms
```

1.5 Ping ze światem zewnętrznym

Z każdego urządzenia możliwe jest wysłanie komunikatów ping do zewnętrznych serwerów. Poniżej przykłady:

ping google.com z *PC2*:

```
PC2> ping google.com
google.com resolved to 142.250.179.142

84 bytes from 142.250.179.142 icmp_seq=1 ttl=57 time=59.586 ms
84 bytes from 142.250.179.142 icmp_seq=2 ttl=57 time=55.037 ms
84 bytes from 142.250.179.142 icmp_seq=3 ttl=57 time=54.871 ms
```

ping cs.pwr.edu.pl z routera *R2*:

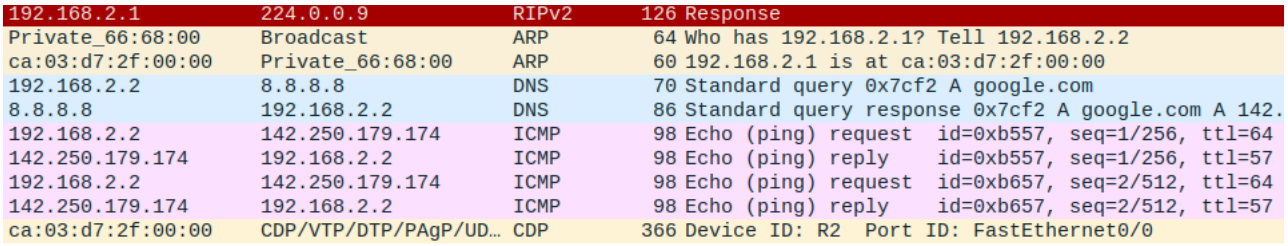
```
R2# ping cs.pwr.edu.pl
Translating "cs.pwr.edu.pl"...domain server (8.8.8.8) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 156.17.7.22, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/30/32 ms
```

1.6 Przechwytywanie pakietów programem Wireshark

W programie *GNS3* mamy również możliwość ustawienia przechwytywania ruchu sieciowego na wybranych łączach. Realizowane jest to za pomocą *Wiresharka*. W naszej konfiguracji nasłuchiwanie włączone jest dla sieci: 192.168.0.0, 192.168.2.0, 192.168.3.0.

Będziemy wysyłać komunikaty *ping google.com* z *PC2* i analizować uzyskany ruch.

Poniższy zrzut przedstawia komunikację w 192.168.2.0:

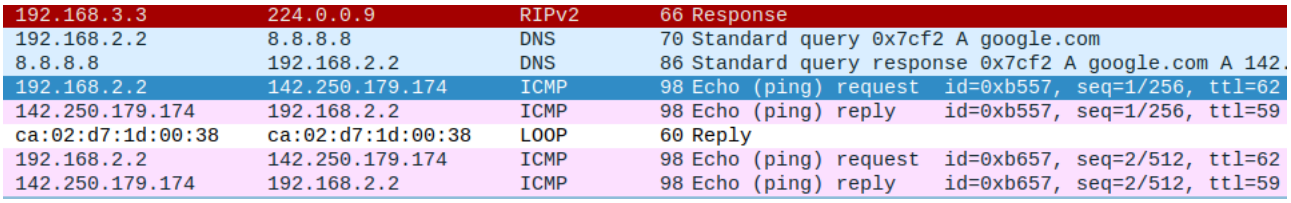


192.168.2.1	224.0.0.9	RIPv2	126 Response
Private_66:68:00	Broadcast	ARP	64 Who has 192.168.2.1? Tell 192.168.2.2
ca:03:d7:2f:00:00	Private_66:68:00	ARP	60 192.168.2.1 is at ca:03:d7:2f:00:00
192.168.2.2	8.8.8.8	DNS	70 Standard query 0x7cf2 A google.com
8.8.8.8	192.168.2.2	DNS	86 Standard query response 0x7cf2 A google.com A 142.
192.168.2.2	142.250.179.174	ICMP	98 Echo (ping) request id=0xb557, seq=1/256, ttl=64
142.250.179.174	192.168.2.2	ICMP	98 Echo (ping) reply id=0xb557, seq=1/256, ttl=57
192.168.2.2	142.250.179.174	ICMP	98 Echo (ping) request id=0xb657, seq=2/512, ttl=64
142.250.179.174	192.168.2.2	ICMP	98 Echo (ping) reply id=0xb657, seq=2/512, ttl=57
ca:03:d7:2f:00:00	CDP/VTP/DTP/PAgP/UD…	CDP	366 Device ID: R2 Port ID: FastEthernet0/0

Widać tutaj kilka różnych protokołów:

- *RIPv2* (Routing Information Protocol) - służy do określania ścieżek
- *ARP* (Address Resolution Protocol) - służy do odwzorowania adresu logicznego na fizyczny
- DNS (Domain Name System) - służy do tłumaczenia nazw hostów
- *ICMP* (Internet Control Message Protocol) - służy do kontroli transmisji danych
- *CDP* (Cisco Discovery Protocol) - służy do wykrywania urządzeń z sąsiedztwa

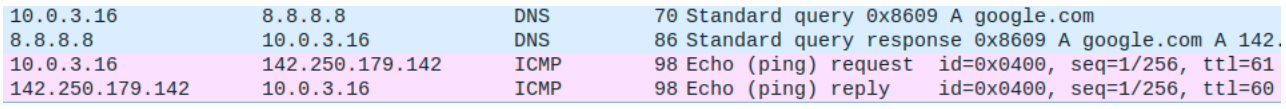
Poniższy zrzut przedstawia komunikację w 192.168.3.0:



192.168.3.3	224.0.0.9	RIPv2	66 Response
192.168.2.2	8.8.8.8	DNS	70 Standard query 0x7cf2 A google.com
8.8.8.8	192.168.2.2	DNS	86 Standard query response 0x7cf2 A google.com A 142.
192.168.2.2	142.250.179.174	ICMP	98 Echo (ping) request id=0xb557, seq=1/256, ttl=62
142.250.179.174	192.168.2.2	ICMP	98 Echo (ping) reply id=0xb557, seq=1/256, ttl=59
ca:02:d7:1d:00:38	ca:02:d7:1d:00:38	LOOP	60 Reply
192.168.2.2	142.250.179.174	ICMP	98 Echo (ping) request id=0xb657, seq=2/512, ttl=62
142.250.179.174	192.168.2.2	ICMP	98 Echo (ping) reply id=0xb657, seq=2/512, ttl=59

Widać, że pakiety przemieszczają się tą ścieżką, ale zostały przepakowane przy przejściach przez routery. Wartość pola TTL zmniejszyła się.

Poniższy zrzut przedstawia komunikację w 192.168.0.0:



10.0.3.16	8.8.8.8	DNS	70 Standard query 0x8609 A google.com
8.8.8.8	10.0.3.16	DNS	86 Standard query response 0x8609 A google.com A 142.
10.0.3.16	142.250.179.142	ICMP	98 Echo (ping) request id=0x0400, seq=1/256, ttl=61
142.250.179.142	10.0.3.16	ICMP	98 Echo (ping) reply id=0x0400, seq=1/256, ttl=60

Router *R5*, który jest połączony z *Cloud*, dokonuje translacji i wysyła pakiet dalej, do sieci zewnętrznej. Po otrzymaniu odpowiedzi, przesyła ją dalej, w głąb naszej sieci.

1.7 Wnioski

Używając *GNS3* można bliżej przyjrzeć się temu, jak należy skonfigurować sieć, aby poprawnie działała. Program przydaje się do diagnostyki i testowania różnych topologii. Bardzo ważną funkcjonalnością jest możliwość używania różnych obrazów systemów operacyjnych routerów bez konieczności posiadania prawdziwego sprzętu. W rezulatcie otrzymujemy środowisko, które wiernie odwzorowuje rzeczywiste zachowania w sieciach komputerowych.