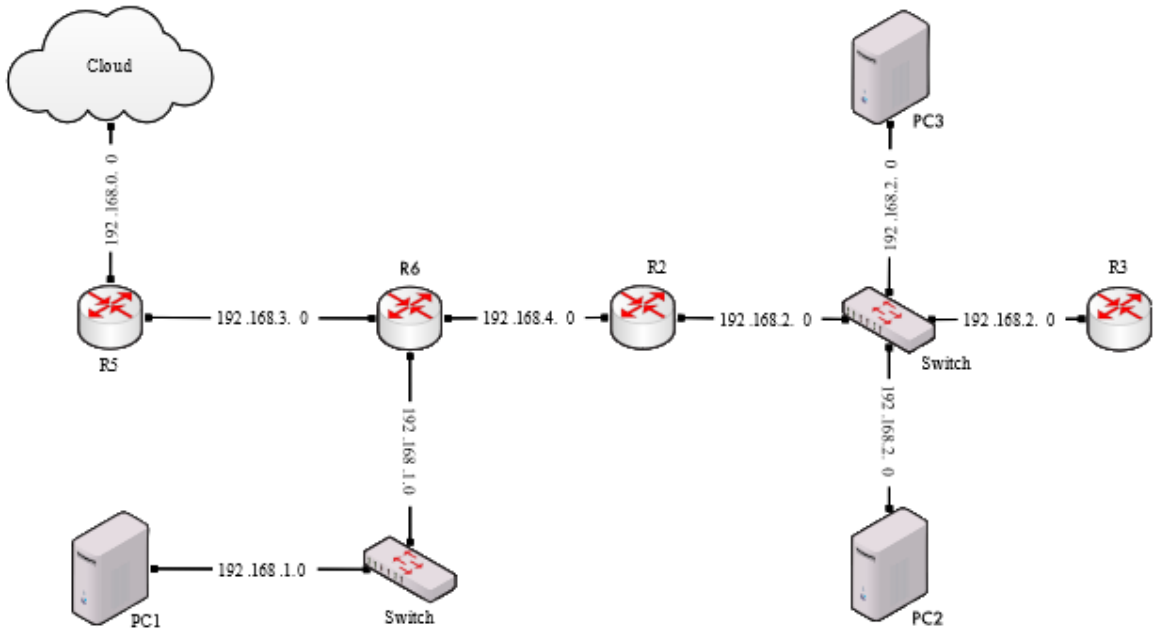
SPRAWOZDANIE 4

TECHNOLOGIE SIECIOWE

Wojciech Wróblewski

Opis doświadczenia

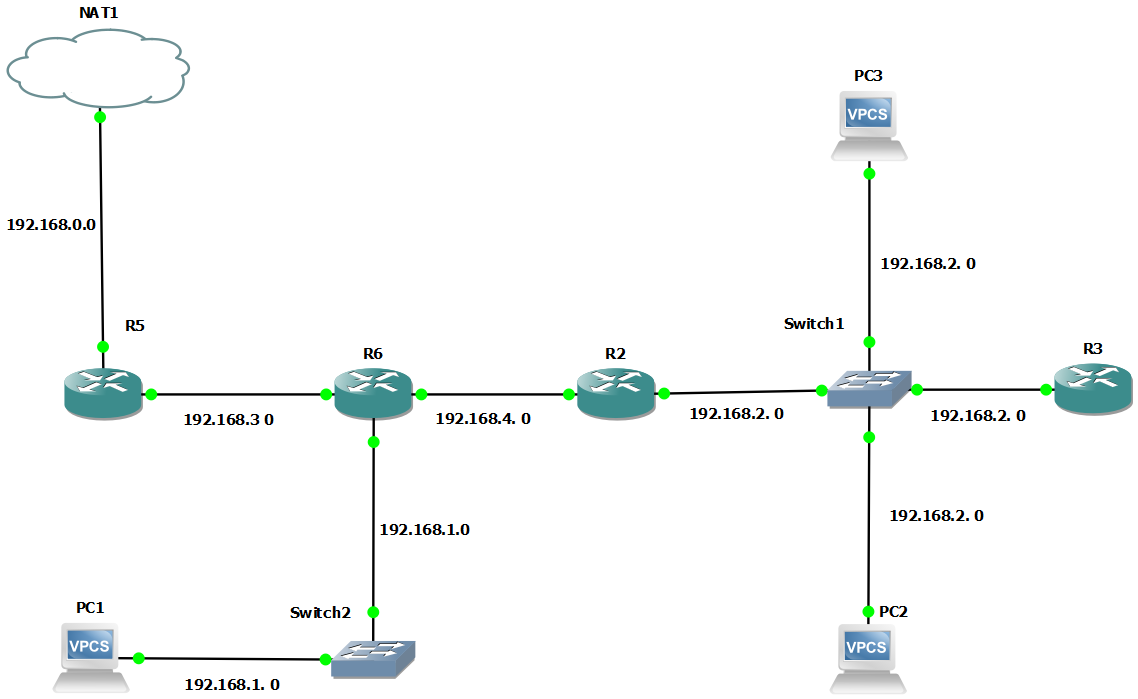
Celem doświadczenia jest skonstruowanie w symulatorze GNS3 wirtualnej sieci o podanej topologii.



- Chcemy, aby wirtualna sieć była połączona z zewnętrzną ('fizyczną') siecią 'Cloud'.
- Ruter R5 uzyskiwał dynamiczny adres IP z sieci 'Cloud'.
- Pozostałe urządzenia posiadały statyczne adresy w swoich sieciach.
- Możliwe było wysyłanie komunikatów "ping" pomiędzy dowolna parą urządzeń sieci wirtualnej.
- Możliwe było wysyłanie komunikatów "ping" z dowolnego urządzenia w sieci wirtualnej na zewnętrzny adres, np. 'google.com'.
- Dodatkowo chcemy uruchomić narzędzie do przechwytywanie komunikatów na sieciach: 192.168.0.0, 192.168.2.0, 192.168.3.0
- Chcemy przeprowadzić analizę przechwyconych komunikatów dla zapytania wysłanego z komputera PC2: 'ping google.com'.

Realizacja zadania.

PO zainstalowaniu oraz skonfigurowaniu środowiska GNS3, możemy przystąpić do konstrukcji naszej sieci wybierając odpowiednie komponenty. Początkowo konfiguracja nie posiadała routera, wobec tego należy pobrać obraz routera oraz skonfigurować go w środowisku GNS3. W realizacji zadania router to CISCO c3600. Po połączniu wszystkich komponentów nasza sieć będzie wyglądała tak.



Gdy mamy już schemat naszej topologii skonfigurujmy poszczególne urządzenia w terminalu. Skonfigurujmy urządzenia tak, aby spełniały wymagania zadane w opisie. Rozpocznijmy od routera R5. Po wpisaniu w terminalu routera R5 komendy “conf t” (uprzywilejowany tryb routera) wpisujemy kolejne komendy zgodnie z przedstawionym poniżej obrazem:

```
R5(config)#int f0/0
R5(config-if)#ip address dhcp
R5(config-if)#ip nat outside

*Mar  1 00:21:04.027: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
R5(config-if)#n
*Mar  1 00:21:10.239: %SYS-3-CPUHOG: Task is running for (2036)msecs, more than (2000)msecs (0/0),process = Exec.
-Traceback= 0x613188B8 0x613182C8 0x612DF970 0x612DFC30 0x612DFD54 0x612DFD54 0x612E0C24 0x61313A74 0x6131FC6C 0x6130A064 0x
6130ACC8 0x6130BC18 0x60F5005C 0x60479074 0x60495338 0x60538528
*Mar  1 00:21:10.663: %SYS-3-CPUYLD: Task ran for (2460)msecs, more than (2000)msecs (0/0),process = Exec
R5(config-if)#no shut
R5(config-if)#en
*Mar  1 00:21:30.375: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:21:31.375: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R5(config-if)#end
R5#
*Mar  1 00:21:34.939: %SYS-5-CONFIG_I: Configured from console by console
R5#
*Mar  1 00:21:40.935: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned DHCP address 192.168.124.129, mask 255.255.
255.0, hostname R5

R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#ip domain-loopup
                  ^
% Invalid input detected at '^' marker.

R5(config)#ip domain-lookup
R5(config)#ip name-server 8.8.8.8
R5(config)#end
R5#
*Mar  1 00:22:26.691: %SYS-5-CONFIG_I: Configured from console by console
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#int f0/1
R5(config-if)#ip add 192.168.3.3 255.255.255.0
R5(config-if)#ip nat inside
R5(config-if)#no shut
R5(config-if)#end
*Mar  1 00:23:52.131: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:23:53.131: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R5(config-if)#end
R5#
*Mar  1 00:23:58.995: %SYS-5-CONFIG_I: Configured from console by console
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 192.168.0.0
R5(config-router)#network 192.168.3.0
R5(config-router)#default-information originate
R5(config-router)#end
R5#
*Mar  1 00:25:47.475: %SYS-5-CONFIG_I: Configured from console by console
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)#ip nat inside source list 10 interface f0/0 overload
```

W dalszej części sprawozdania zrezygnuję ze screenów z konsoli na rzecz obrazów bardziej jakościowo prezentujących kolejne kroki. Poniższe obrazy pokazują kolejno wpisane komendy w terminalu przy konfigurowaniu każdego z urządzeń.

Konfiguracja routera R5

```
conf t
int f0/0
ip address dhcp
ip nat outside
no shut
end
conf t
ip domain-lookup
ip name-server 8.8.8.8
end
conf t
int f0/1
ip add 192.168.3.3 255.255.255.0
ip nat inside
no shut
end
conf t
router rip
version 2
no auto-summary
network 192.168.0.0
network 192.168.3.0
default-information originate
end
conf t
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
ip nat inside source list 10 interface f0/0 overload
end
write
```

Konfiguracja routera R6

```
conf t
int f0/0
no shut
ip add 192.168.3.1 255.255.255.0
end
conf t
ip domain-lookup source-interface f0/0
ip name-server 8.8.8.8
end
conf t
int e1/0
no shut
ip add 192.168.1.1 255.255.255.0
end
conf t
int f0/1
no shut
ip add 192.168.4.1 255.255.255.0
end
conf t
router rip
version 2
no auto-summary
network 192.168.4.0
network 192.168.1.0
network 192.168.3.0
end
write
```

Konfiguracja routera R2

```
conf t
int f0/1
ip add 192.168.2.1 255.255.255.0
no shut
int f0/0
ip add 192.168.4.2 255.255.255.0
no shut
end
conf t
router rip
version 2
no auto-summary
network 192.168.2.0
network 192.168.4.0
end
conf t
ip domain-lookup
ip name-server 8.8.8.8
end
write
```

Konfiguracja routera R3

```
conf t
int f0/0
ip add 192.168.2.3 255.255.255.0
no shut
end
conf t
router rip
version 2
no auto-summary
network 192.168.2.0
end
conf t
ip domain-lookup
ip name-server  8.8.8.8
end
write
```

Konfiguracja PC1

```
ip 192.168.1.2/24 192.168.1.1
ip dns 8.8.8.8
save
```

Konfiguracja PC2

```
ip 192.168.2.2/24 192.168.2.1
ip dns 8.8.8.8
save
```

Konfiguracja PC3

```
ip 192.168.2.4/24 192.168.2.1
ip dns 8.8.8.8
save
```

Router R5 jako jedyny charakteryzuje się dynamicznie przyznawanym adresem ip przyznawanym dzięki komendzie **dhcp**. Kolejne routery mają statycznie przypisane adresy komendą **ip add**.

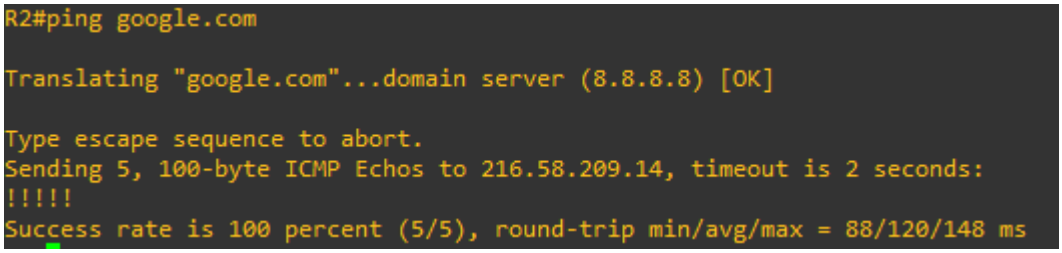
Zastosujemy komendę **ip nat outside** w konfiguracji interfejsu od strony, która będzie poza naszą siecią . Interfejs włączamy komendą **no shut**. Za pomocą komendy **ip domain-lookup** włączamy funkcję DNS (konfiguracji routera), a za pomocą komendy **ip name-server** określamy adres ip serwera.

Wprowadzenie RIP w wersji 2 (komendy rip, version 2) jest używane do zapewnienia routingu w małych sieciach bazujących na adresach IPv4. RIPv2 jest bezklasowym protokołem routingu wektora odległość. Domyślnie w protokole RIPv2 włączona jest automatyczna sumaryzacja tras, poprawimy to dodając komendę **no auto-summary**. Wpisując komendę network (np. network 192.168.0.0) definiujemy, które sieci router ma rozgłaszać.

Aby skorzystać z mechanizmów filtrowania ruchu opartych na listach dostępu ACLs w danym interfejsie, używamy komendy **access-list**. Dzięki listą ACL administrator ma możliwość definiowania obszarów oraz ich dostępności dla poszczególnych urządzeń budujących sieć. Podczas budowy warunków tworzących daną listę ACL należy zdefiniować sposób działania tzn. czy zezwalamy na dostęp (**permit**) czy go odbieramy (**deny**). W komendzie w routerze R5 stosujemy 10 jako numer listy ACL. (10 odpowiada standardowej liście dostępu IP). Za pomocą komendy **ip nat inside source list list 1 interface f0/0 overload**, ustawiamy PAT (port addres translation) . Dzięki temu tylko na jednym adresie IP publicznym jesteśmy w stanie obsłużyć wiele adresów prywatnych. W przypadku komputerów konfiguracja jest bardzo krótka. Określamy jedynie adres IP komputera oraz adres DNS.

Otrzymane wyniki symulacji.

Przetestujmy, czy sieć realizuje zadane funkcjonalności. Poniższy obrazy pokazują, że pingowanie zewnętrznej strony “google" kończy się sukcesem dla każdego routera w naszej sieci.



```
R6#ping google.com

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.217.16.14, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 88/113/124 ms
```

```
R5#ping google.com

Translating "google.com"...domain server (192.168.124.2) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 216.58.215.110, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/84/96 ms
```

```
R3#ping google.com

Translating "google.com"...domain server (192.168.0.1) (8.8.8.8) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 216.58.209.14, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 148/172/272 ms
```

Przetestujmy pingowanie strony google przez PC1, PC2 oraz PC3

```
PC1> ping google.com
google.com resolved to 172.217.16.46
84 bytes from 172.217.16.46 icmp_seq=1 ttl=126 time=61.585 ms
84 bytes from 172.217.16.46 icmp_seq=2 ttl=126 time=47.187 ms
84 bytes from 172.217.16.46 icmp_seq=3 ttl=126 time=49.634 ms
84 bytes from 172.217.16.46 icmp_seq=4 ttl=126 time=60.073 ms
84 bytes from 172.217.16.46 icmp_seq=5 ttl=126 time=63.906 ms
```

```
PC2> ping google.com
google.com resolved to 172.217.16.14
84 bytes from 172.217.16.14 icmp_seq=1 ttl=125 time=83.687 ms
84 bytes from 172.217.16.14 icmp_seq=2 ttl=125 time=67.874 ms
84 bytes from 172.217.16.14 icmp_seq=3 ttl=125 time=76.575 ms
84 bytes from 172.217.16.14 icmp_seq=4 ttl=125 time=69.717 ms
84 bytes from 172.217.16.14 icmp_seq=5 ttl=125 time=84.316 ms
```

```
PC3> ping google.com
google.com resolved to 216.58.208.206
84 bytes from 216.58.208.206 icmp_seq=1 ttl=125 time=83.872 ms
84 bytes from 216.58.208.206 icmp_seq=2 ttl=125 time=84.196 ms
84 bytes from 216.58.208.206 icmp_seq=3 ttl=125 time=80.993 ms
84 bytes from 216.58.208.206 icmp_seq=4 ttl=125 time=77.361 ms
84 bytes from 216.58.208.206 icmp_seq=5 ttl=125 time=74.562 ms
```

Obserwujemy, że operacje kończą się sukcesem.

Sprawdźmy, czy skonfigurowana sieć umożliwia również wysyłanie komunikatu ping pomiędzy dowolną parą urządzeń w sieci.

Przykład przesłania ping z routera R6 do routera R5.

```
R6# ping 192.168.3.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms
```

Przykład pingowania PC1 z routera R6.

```
R6#ping 192.168.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/58/64 ms
```

Przykład pingowania R6 z PC1.

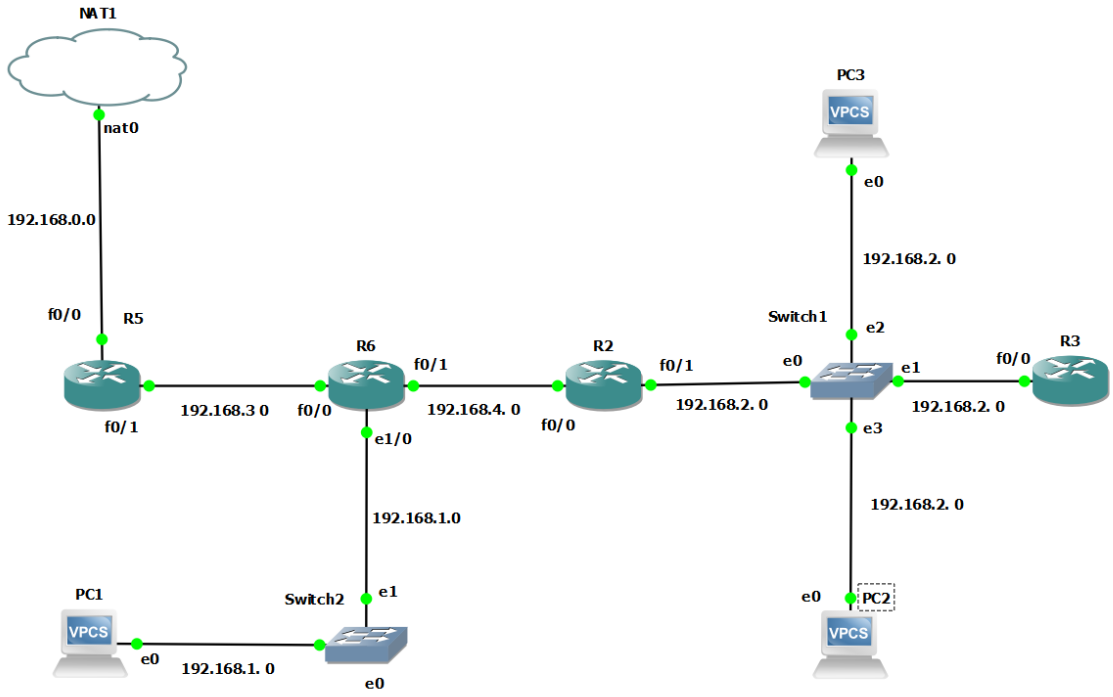
```
PC1> ping 192.168.1.1
84 bytes from 192.168.1.1 icmp_seq=1 ttl=255 time=9.145 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=255 time=6.325 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=255 time=10.422 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=255 time=2.455 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=255 time=12.168 ms
```

Ostatni obraz to pingowanie PC2 z PC1.

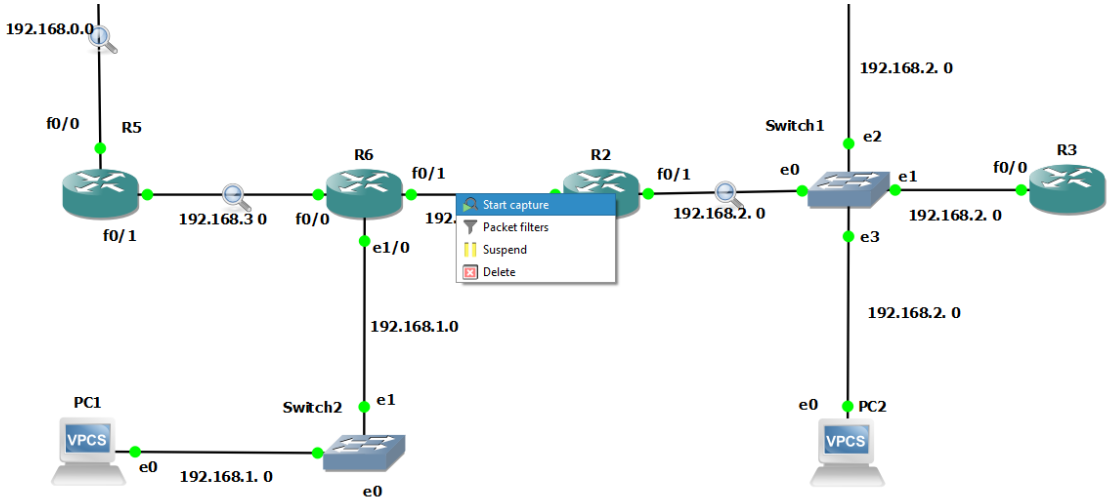
```
PC1> ping 192.168.2.2
84 bytes from 192.168.2.2 icmp_seq=1 ttl=62 time=41.023 ms
84 bytes from 192.168.2.2 icmp_seq=2 ttl=62 time=35.250 ms
84 bytes from 192.168.2.2 icmp_seq=3 ttl=62 time=38.340 ms
84 bytes from 192.168.2.2 icmp_seq=4 ttl=62 time=32.734 ms
84 bytes from 192.168.2.2 icmp_seq=5 ttl=62 time=41.529 ms
```

Obserwujemy, że wszystkie operacje przebiegły pomyślnie.

Prezentacja schematu sieci wraz z widocznymi etykietami interfejsów.



Teraz zaprezentujmy, jak włączyć przechwytywanie komunikatów przy pomocy Wiresharka. Wystarczy wskazać prawym klawiszem na interesujące nas sieci i włączyć przechwytywanie. Automatycznie dla każdego wyboru odpalą nam się instancje Wiresharka dla każdej sieci.



Teraz wyślijmy “ping google.com” z komputera PC2 .

```
PC2> ping google.com
google.com resolved to 172.217.16.46
84 bytes from 172.217.16.46 icmp_seq=1 ttl=125 time=219.855 ms
84 bytes from 172.217.16.46 icmp_seq=2 ttl=125 time=81.545 ms
84 bytes from 172.217.16.46 icmp_seq=3 ttl=125 time=83.578 ms
84 bytes from 172.217.16.46 icmp_seq=4 ttl=125 time=102.916 ms
84 bytes from 172.217.16.46 icmp_seq=5 ttl=125 time=82.877 ms
```

Widzimy, że pakiety zostały pomyślnie wysłane. Zobaczmy co otrzymamy w Wiresharku.

Przechwyt w sieci 192.168.2. po wysłaniu komunikatu ping z PC2.

W linii 78 mamy zapytanie do DNS od komputera PC2 o uzyskanie adresu IP witryny “google”. W lini 79 uzyskujemy odpowiedź przychodzącą do PC2 (192.168.2.2) w postaci IP do witryny “google.com” (172.217.16.46). Od linii 80 zaczyna się propagacja zapytania ping, która wykonuje się 5 razy. Możemy rozróżnić pary komunikatów (np. Linia 80 oraz 81). Pierwszy komunikat z pary to wysyłanie komunikatu ping na adres strony docelowej (google.com). Drugi komunikat to odpowiedź przychodząca od strony do naszej sieci. Rozpoznajemy to po wartościach IP w kolumnach source oraz destination.

Komunikacja, czyli wysłanie zapytania oraz odpowiedź, realizowane są za pomocą protokołu ICMP (Internet Control Message Protocol), który jest internetowym protokołem komunikatów kontrolnych.

No.	Time	Source	Destination	Protocol	Length	Info
71	338.236337	192.168.2.2	172.217.20.174	ICMP	98	Echo (ping) request id=0xa8e3, seq=5/1280, ttl=64 (reply in …
72	338.318323	172.217.20.174	192.168.2.2	ICMP	98	Echo (ping) reply id=0xa8e3, seq=5/1280, ttl=125 (request …
73	340.244251	cc:02:11:54:00:01	cc:02:11:54:00:01	LOOP	60	Reply
74	347.066455	192.168.2.1	224.0.0.9	RIPv2	126	Response
75	350.601502	cc:02:11:54:00:01	cc:02:11:54:00:01	LOOP	60	Reply
76	358.731869	cc:03:62:34:00:00	CDP/VTP/DTP/PAgP/UD…	CDP	341	Device ID: R3 Port ID: FastEthernet0/0
77	360.992691	cc:02:11:54:00:01	cc:02:11:54:00:01	LOOP	60	Reply
78	361.895465	192.168.2.2	8.8.8.8	DNS	70	Standard query 0xcd7c A google.com
79	362.008695	8.8.8.8	192.168.2.2	DNS	86	Standard query response 0xcd7c A google.com A 172.217.16.46
80	362.010646	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xbfe3, seq=1/256, ttl=64 (reply in 8…
81	362.230010	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xbfe3, seq=1/256, ttl=125 (request i…
82	363.232542	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc1e3, seq=2/512, ttl=64 (reply in 8…
83	363.313524	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc1e3, seq=2/512, ttl=125 (request i…
84	364.315914	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc2e3, seq=3/768, ttl=64 (reply in 8…
85	364.398885	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc2e3, seq=3/768, ttl=125 (request i…
86	365.400007	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc3e3, seq=4/1024, ttl=64 (reply in …
87	365.502450	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc3e3, seq=4/1024, ttl=125 (request …
88	366.503861	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc4e3, seq=5/1280, ttl=64 (reply in …
89	366.586116	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc4e3, seq=5/1280, ttl=125 (request …

Przechwyt w sieci 192.168.3. po wysłaniu komunikatu ping z PC2.

Tutaj nie obserwujemy zmiany w stosunku do sieci 192.168.2. . Zadaniem sieci 192.168.3. jest propagacja komunikatów przez lokalną sieć wirtualną tak aby dotarły do routera R5.

No.	Time	Source	Destination	Protocol	Length	Info
129	386.183077	cc:06:3e:c8:00:00	CDP/VTP/DTP/PAgP/UD…	CDP	341	Device ID: R6 Port ID: FastEthernet0/0
130	388.269766	cc:05:41:14:00:01	cc:05:41:14:00:01	LOOP	60	Reply
131	391.208814	cc:06:3e:c8:00:00	cc:06:3e:c8:00:00	LOOP	60	Reply
132	396.735632	192.168.3.1	224.0.0.9	RIPv2	106	Response
133	396.990612	192.168.3.3	224.0.0.9	RIPv2	66	Response
134	398.555513	cc:05:41:14:00:01	cc:05:41:14:00:01	LOOP	60	Reply
135	400.413828	192.168.2.2	8.8.8.8	DNS	70	Standard query 0xcd7c A google.com
136	400.487041	8.8.8.8	192.168.2.2	DNS	86	Standard query response 0xcd7c A google.com A 172.217.16.46
137	400.529693	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xbfe3, seq=1/256, ttl=62 (reply in 1…
138	400.708603	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xbfe3, seq=1/256, ttl=127 (request i…
139	401.366693	cc:06:3e:c8:00:00	cc:06:3e:c8:00:00	LOOP	60	Reply
140	401.748961	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc1e3, seq=2/512, ttl=62 (reply in 1…
141	401.791904	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc1e3, seq=2/512, ttl=127 (request i…
142	402.835253	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc2e3, seq=3/768, ttl=62 (reply in 1…
143	402.877219	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc2e3, seq=3/768, ttl=127 (request i…
144	403.916379	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc3e3, seq=4/1024, ttl=62 (reply in …
145	403.980833	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc3e3, seq=4/1024, ttl=127 (request …
146	405.023200	192.168.2.2	172.217.16.46	ICMP	98	Echo (ping) request id=0xc4e3, seq=5/1280, ttl=62 (reply in …
147	405.064157	172.217.16.46	192.168.2.2	ICMP	98	Echo (ping) reply id=0xc4e3, seq=5/1280, ttl=127 (request …
148	408.328330	cc:05:41:14:00:01	cc:05:41:14:00:01	LOOP	60	Reply

Przechwyt w sieci 192.168.0. po wysłaniu komunikatu ping z PC2.

Obserwujemy wysłanie z routera R5 (dynamicznie przyznany IP na 192.168.124.131) zapytania o IP strony “google.com” (linia 187). Następnie widzimy komunikację pomiędzy

routerem a zewnętrznym serwerem spoza naszej sieci. Router R5 po otrzymaniu IP do strony (linia 188) zaczyna wysyłać zapytania ping (od linii 189). Źródłem przesłanego zapytania jest router R5, a nie komputerPC2, który zainicjalizował wysyłanie komunikatu. Dzieje się tak dlatego, że to właśnie router R5 jest reprezentantem naszej sieci globalnie i to on koordynuje i wysyła zapytania, bazując na komunikatach i rozkazach powstałych wewnątrz naszej lokalnej sieci wirtualnej (w tym przypadku pingowania z komputera PC2).

No.	Time	Source	Destination	Protocol	Length	Info
182	409.902433	172.217.20.174	192.168.124.131	ICMP	98	Echo (ping) reply id=0xa8e3, seq=5/1280, ttl=128 (request …
183	410.885797	cc:05:41:14:00:00	cc:05:41:14:00:00	LOOP	60	Reply
184	421.218430	cc:05:41:14:00:00	cc:05:41:14:00:00	LOOP	60	Reply
185	429.259217	cc:05:41:14:00:00	CDP/VTP/DTP/PAgP/UD…	CDP	341	Device ID: R5 Port ID: FastEthernet0/0
186	431.504274	cc:05:41:14:00:00	cc:05:41:14:00:00	LOOP	60	Reply
187	433.532093	192.168.124.131	8.8.8.8	DNS	70	Standard query 0xcd7c A google.com
188	433.588835	8.8.8.8	192.168.124.131	DNS	86	Standard query response 0xcd7c A google.com A 172.217.16.46
189	433.647677	192.168.124.131	172.217.16.46	ICMP	98	Echo (ping) request id=0xbfe3, seq=1/256, ttl=61 (reply in 1…
190	433.812681	172.217.16.46	192.168.124.131	ICMP	98	Echo (ping) reply id=0xbfe3, seq=1/256, ttl=128 (request i…
191	434.867443	192.168.124.131	172.217.16.46	ICMP	98	Echo (ping) request id=0xc1e3, seq=2/512, ttl=61 (reply in 1…
192	434.891966	172.217.16.46	192.168.124.131	ICMP	98	Echo (ping) reply id=0xc1e3, seq=2/512, ttl=128 (request i…
193	434.937437	192.168.124.1	192.168.124.255	UDP	86	57621 → 57621 Len=44
194	435.953823	192.168.124.131	172.217.16.46	ICMP	98	Echo (ping) request id=0xc2e3, seq=3/768, ttl=61 (reply in 1…
195	435.980568	172.217.16.46	192.168.124.131	ICMP	98	Echo (ping) reply id=0xc2e3, seq=3/768, ttl=128 (request i…
196	437.034907	192.168.124.131	172.217.16.46	ICMP	98	Echo (ping) request id=0xc3e3, seq=4/1024, ttl=61 (reply in …
197	437.080886	172.217.16.46	192.168.124.131	ICMP	98	Echo (ping) reply id=0xc3e3, seq=4/1024, ttl=128 (request …
198	438.141166	192.168.124.131	172.217.16.46	ICMP	98	Echo (ping) request id=0xc4e3, seq=5/1280, ttl=61 (reply in …
199	438.167625	172.217.16.46	192.168.124.131	ICMP	98	Echo (ping) reply id=0xc4e3, seq=5/1280, ttl=128 (request …
200	441.277133	cc:05:41:14:00:00	cc:05:41:14:00:00	LOOP	60	Reply

Wnioski

Symulator GNS3 to bardzo rozbudowane narzędzie do konstrukcji oraz analizy sieci komputerowych. Daje nam możliwości symulowania oraz wdrażania projektów sieci i wraz z dołączonymi do niego innymi programami pozwala na szczegółową analizę dróg pakietów, analizowania ich budowy oraz celów.