

Technologie sieciowe

Lista 4

Konfiguracja sieci IP

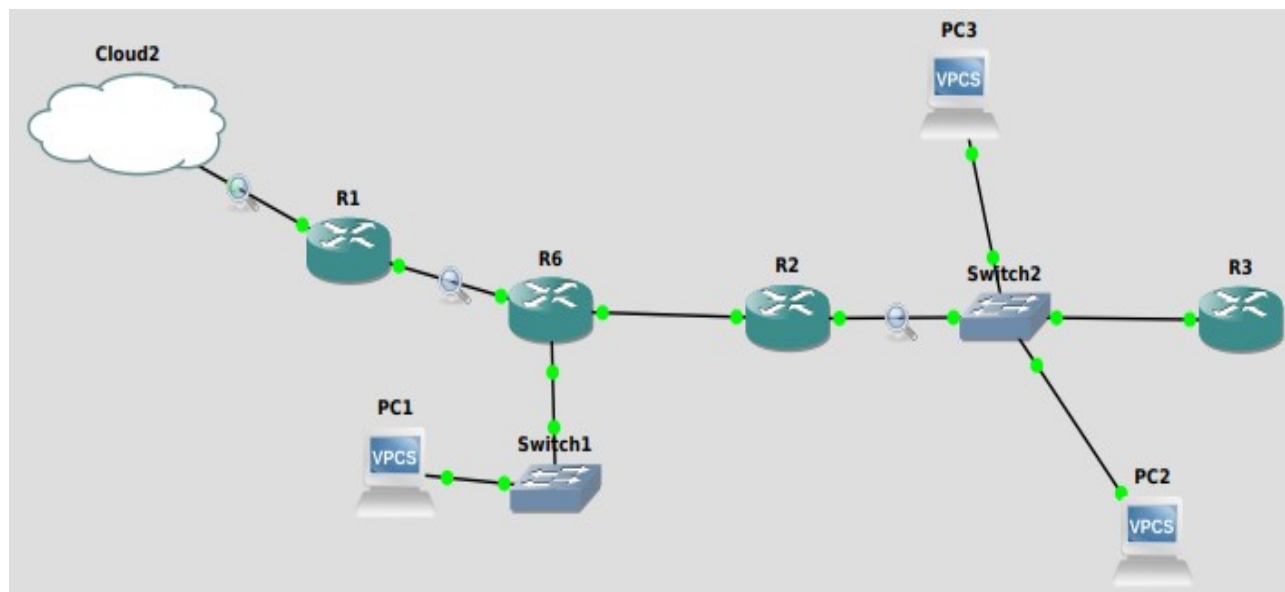
Michał Kalina
250088

KONFIGURACJA

W sieci zostały użyte VPCS (Virtual PC Simulator) pozwalające symulować lekki komputer PC obsługujący DHCP i ping, były one też proste w konfiguracji (ip <adres> <gateway>, ip dns 8.8.8.8), routery CISCO 7200 oraz switch'e bez możliwości programowania.

Adres DNS został ustawiony jako 8.8.8.8. W sieci użyłem protokołu routingu RIP version 2.

Do emulacji wykorzystałem program VirtualBox. Ruchy sieci śledziłem za pomocą Wireshark'a.



Topologia sieci

Konfiguracja R2:

R2(config)#conf t

config terminal

R2(config)#inter et2/1

interface – w tym wypadku Ethernet1

R2(config-if)#ip add 192.168.2.1 255.255.255.0

dodawanie adresu ip

R2(config-if)#duplex full

ustawianie duplex { **full** | **half** | **auto** } (bez tej komendy też działa, bo jest ustawione domyślnie. Jest to informacja że dane są przesyłane w obu kierunkach jednocześnie, bez spadku transferu)

R2(config-if)#no shut

no shutdown - aby zrestartować wyłączony interfejs

R2(config-if)#exit

R2(config)#int et2/0

R2(config-if)#ip add 10.1.1.2 255.255.255.0

R2(config-if)#no shut

R2(config-if)#exit

R2(config)#router rip

Routing Information Protocol – protokół bram wewnętrznych

R2(config-router)#version 2

R2(config-router)#no auto-summary

R2(config-router)#network 192.168.4.0

dodawanie adresów

R2(config-router)#network 192.168.2.0

R2(config-router)#exit

R2(config)#ip domain-lookup

włączenie funkcji wyszukiwania DNS (Domain Name Server)

R2(config)#ip name-server 8.8.8.8

R2(config)#end

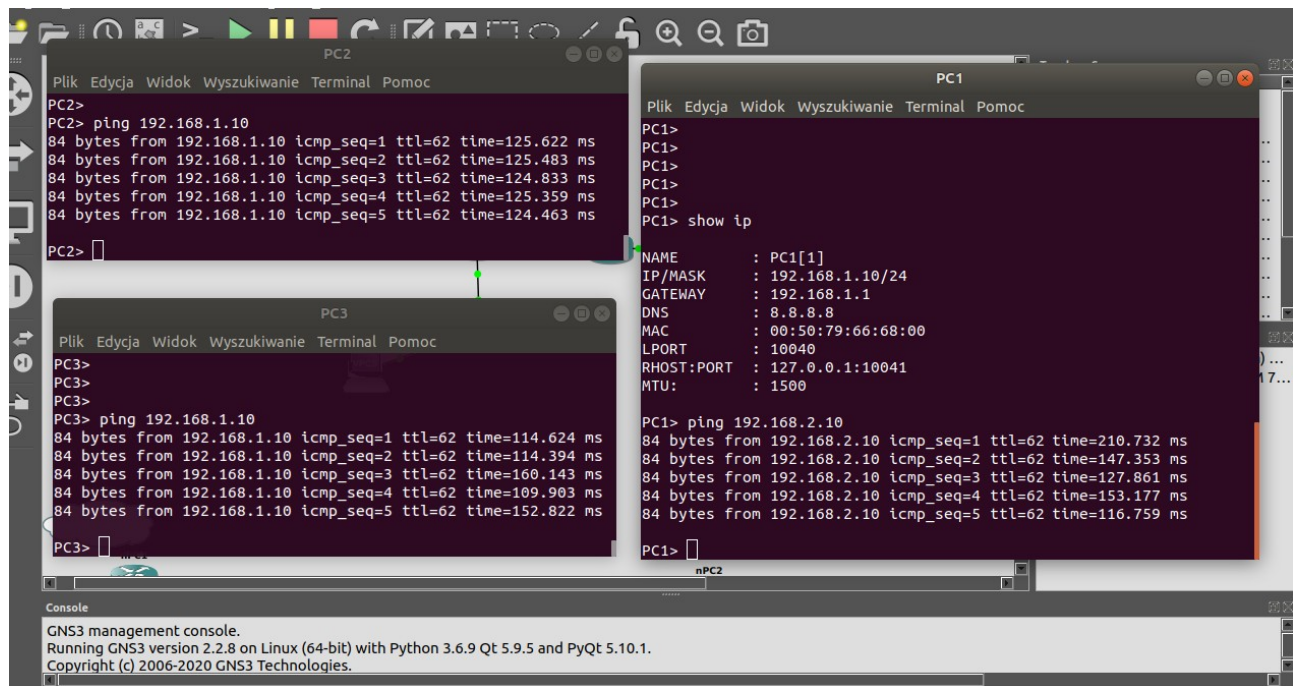
R2(config)# *do copy r s* lub *write* (obie komendy spełniają zadanie) *copy run start*, *copy-kopiowanie klików do pamięci*

Adresy komputerów:

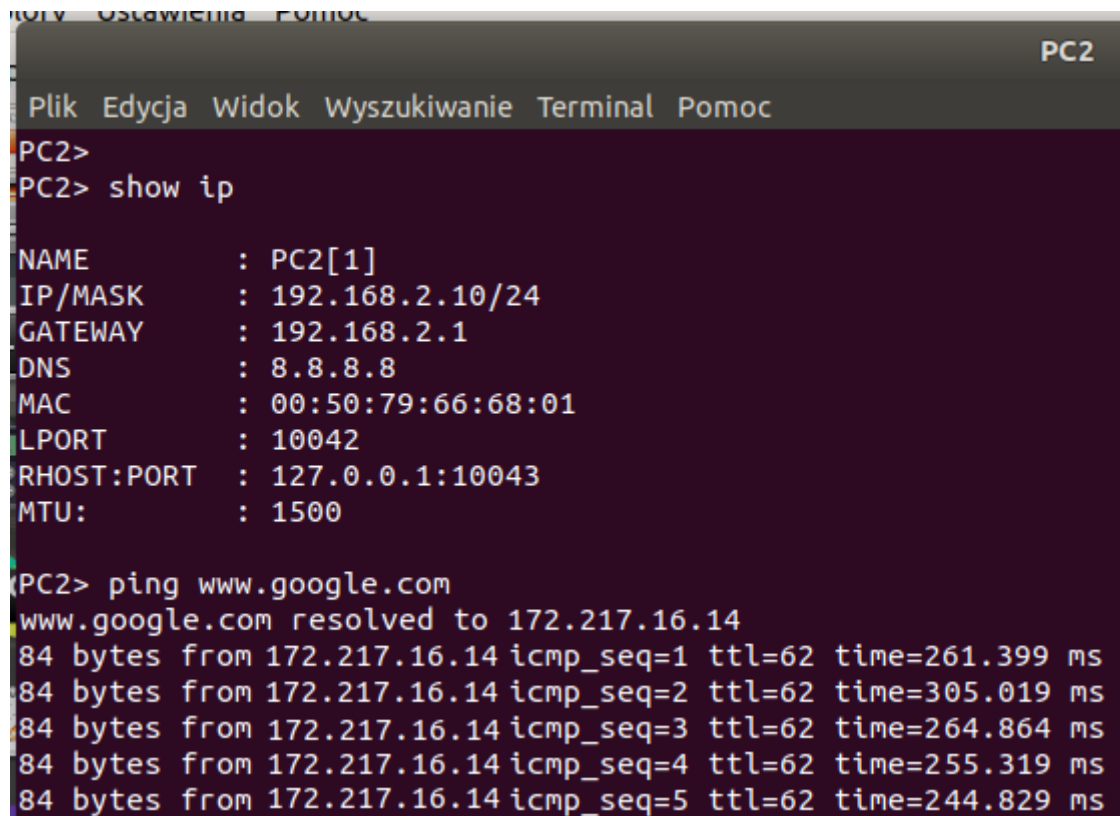
PC1: 192.168.1.10

PC2: 192.168.2.10

PC3: 192.168.2.11



Po skonfigurowaniu każdego komputera



ping google.com z PC2

```

R1#
R1#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 172.217.16.14, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/76/112 ms
R1#

```

Ping google.com z routera 1

Pingowanie w sieci powinno zabierać więcej czasu niż zwykle, ponieważ sieć jest emulowana.

| Time | Source | Destination | Protocol | Length | Info |
|--------------|-------------------|------------------------|----------|--------|---|
| 1 0.000000 | ca:03:19:c2:00:39 | ca:03:19:c2:00:39 | LOOP | 60 | Reply |
| 2 0.865756 | 192.168.2.1 | 224.0.0.9 | RIPv2 | 86 | Response |
| 3 10.014888 | ca:03:19:c2:00:39 | ca:03:19:c2:00:39 | LOOP | 60 | Reply |
| 4 11.725899 | ca:04:19:d1:00:38 | CDP/VTP/DTP/PAGP/UD... | CDP | 362 | Device ID: R3 Port ID: Ethernet2/0 |
| 5 16.395536 | ca:03:19:c2:00:39 | Broadcast | ARP | 60 | Who has 192.168.2.10? Tell 192.168.2.1 |
| 6 16.395756 | Private_66:68:01 | ca:03:19:c2:00:39 | ARP | 60 | 192.168.2.10 is at 00:50:79:66:68:01 |
| 7 16.486122 | 192.168.1.10 | 192.168.2.10 | ICMP | 98 | Echo (ping) request id=0x0cd5, seq=1/256, ttl=62 (reply in |
| 8 16.486280 | 192.168.2.10 | 192.168.1.10 | ICMP | 98 | Echo (ping) reply id=0x0cd5, seq=1/256, ttl=64 (request |
| 9 17.512678 | 192.168.1.10 | 192.168.2.10 | ICMP | 98 | Echo (ping) request id=0x0dd5, seq=2/512, ttl=62 (reply in |
| 10 17.512826 | 192.168.2.10 | 192.168.1.10 | ICMP | 98 | Echo (ping) reply id=0x0dd5, seq=2/512, ttl=64 (request |
| 11 18.539319 | 192.168.1.10 | 192.168.2.10 | ICMP | 98 | Echo (ping) request id=0x0ed5, seq=3/768, ttl=62 (reply in |
| 12 18.539453 | 192.168.2.10 | 192.168.1.10 | ICMP | 98 | Echo (ping) reply id=0x0ed5, seq=3/768, ttl=64 (request |
| 13 19.576032 | 192.168.1.10 | 192.168.2.10 | ICMP | 98 | Echo (ping) request id=0x0fd5, seq=4/1024, ttl=62 (reply in |
| 14 19.576167 | 192.168.2.10 | 192.168.1.10 | ICMP | 98 | Echo (ping) reply id=0x0fd5, seq=4/1024, ttl=64 (request |
| 15 20.008825 | ca:03:19:c2:00:39 | ca:03:19:c2:00:39 | LOOP | 60 | Reply |
| 16 20.613042 | 192.168.1.10 | 192.168.2.10 | ICMP | 98 | Echo (ping) request id=0x11d5, seq=5/1280, ttl=62 (reply in |
| 17 20.613184 | 192.168.2.10 | 192.168.1.10 | ICMP | 98 | Echo (ping) reply id=0x11d5, seq=5/1280, ttl=64 (request |

Przechwytywanie komunikatów w sieci 192.168.4.0.

Podczas przechwytywania wysłałem ping z adresu 192.168.1.10 na adres 192.168.2.10.

Tych 5 zapytań jest zaznaczonych na różowo. U góry widać także pakiet z protokołu CDP (*Cisco Discovery Protocol*), dający dostęp do podsumowania konfiguracji innych bezpośrednio połączonych routerów lub przełączników. CDP jest zastrzeżony przez Cisco. Na czerwono, RIPv2 odpowiada za routing w sieci. ARP to Address Resolution Protocol umożliwiający mapowanie logicznych adresów warstwy sieciowej na fizyczne adresy warstwy łącza danych. Widać także protokół LOOP który jest odpowiednikiem pingu w drugiej warstwie, sprawdza adresy MAC.

| | | | | | |
|------------------|-------------------|-------------------|------|-----|---|
| 3150 5700.009952 | ca:03:19:c2:00:38 | ca:03:19:c2:00:38 | LOOP | 60 | Reply |
| 3151 5703.232887 | 10.1.1.2 | 8.8.8.8 | DNS | 70 | Standard query 0x000e A google.com |
| 3152 5703.352070 | 8.8.8.8 | 10.1.1.2 | DNS | 86 | Standard query response 0x000e A google.com A 172.217.16.46 |
| 3153 5703.383851 | 10.1.1.2 | 172.217.16.46 | ICMP | 114 | Echo (ping) request id=0x0003, seq=0/0, ttl=255 (no response... |
| 3154 5703.482963 | 172.217.16.46 | 10.1.1.2 | ICMP | 110 | Echo (ping) reply id=0x0003, seq=0/0, ttl=44 |
| 3155 5703.494553 | 10.1.1.2 | 172.217.16.46 | ICMP | 114 | Echo (ping) request id=0x0003, seq=1/256, ttl=255 (no respon... |
| 3156 5703.558498 | 172.217.16.46 | 10.1.1.2 | ICMP | 110 | Echo (ping) reply id=0x0003, seq=1/256, ttl=44 |
| 3157 5703.565023 | 10.1.1.2 | 172.217.16.46 | ICMP | 114 | Echo (ping) request id=0x0003, seq=2/512, ttl=255 (no respon... |
| 3158 5703.649161 | 172.217.16.46 | 10.1.1.2 | ICMP | 110 | Echo (ping) reply id=0x0003, seq=2/512, ttl=44 |
| 3159 5703.655619 | 10.1.1.2 | 172.217.16.46 | ICMP | 114 | Echo (ping) request id=0x0003, seq=3/768, ttl=255 (no respon... |
| 3160 5703.719661 | 172.217.16.46 | 10.1.1.2 | ICMP | 110 | Echo (ping) reply id=0x0003, seq=3/768, ttl=44 |
| 3161 5703.726088 | 10.1.1.2 | 172.217.16.46 | ICMP | 114 | Echo (ping) request id=0x0003, seq=4/1024, ttl=255 (no respo... |
| 3162 5703.798163 | 172.217.16.46 | 10.1.1.2 | ICMP | 110 | Echo (ping) reply id=0x0003, seq=4/1024, ttl=44 |

Przechwytywanie komunikatów w sieci 192.168.4.0.

Podczas przechwytywania wysłałem ping google.com

Widać iż na początku próbuje się połączyć poprzez Configuration Testing Protocol: DNS – służy on do tłumaczenia protokołów URL na adresy ip. Następnie łączy się z adresem 172.217.16.46 (google.com).

W powyższej sieci użyłem protokołu routingu RIP version 2 ale można go także zastąpić protokołem OSPF 1 (Open Shortest Path). W osobnym projekcie stworzyłem małą podobną sieć o podobnej funkcjonalności opartej o ten protokół.

| | | | | |
|------------------|-------------------|------------------------|------|--|
| 2200 4092.171782 | 10.1.1.2 | 224.0.0.5 | OSPF | 94 Hello Packet |
| 2201 4093.580193 | ca:02:19:b3:00:38 | CDP/VTP/DTP/PagP/UD... | CDP | 362 Device ID: R6 Port ID: Ethernet2/0 |
| 2202 4096.085112 | 10.1.1.1 | 224.0.0.5 | OSPF | 94 Hello Packet |

Po sprawdzeniu sieci, widać że protokół ten wysyła Hello Pakiet , informujący sąsiadów o istnieniu.

Hello Pakiet zawiera informację o:
masce sieci (**Network Mask**),
interwale wysyłania (**Hello Interval**),
sąsiadach (**Active Neighbor**),
priority routera(**Router Priority**) - najwyższa wartość to DR(designated router),
Dead interval - czas, po jakim pakiet “hello” został odebrany od sąsiadów danego routera.
Designated (backup) router - adres routera (zapasowego) desygnowanego lub “0”,gdy takiego routera jeszcze nie ma

```

Frame 3: 94 bytes on wire (752 bits), 94 bytes captured (752 bits)
Ethernet II, Src: ca:02:19:b3:00:3a (ca:02:19:b3:00:3a), Dst: IPv4
Internet Protocol Version 4, Src: 10.1.3.2, Dst: 224.0.0.5
Open Shortest Path First
  OSPF Header
  OSPF Hello Packet
    Network Mask: 255.255.255.0
    Hello Interval [sec]: 10
    Options: 0x12, (L) LLS Data block, (E) External Routing
    Router Priority: 1
    Router Dead Interval [sec]: 40
    Designated Router: 0.0.0.0
    Backup Designated Router: 0.0.0.0
    Active Neighbor: 10.1.3.1
  OSPF LLS Data Block

```

Szczegółowa analiza ping google.com z PC2

```

.... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 96
  Identification: 0x0000 (0)
  Flags: 0x0000
  Time to live: 44
  Protocol: ICMP (1)
  Header checksum: 0xc693 [validation disabled]
  [Header checksum status: Unverified]
Source: 172.217.16.46
Destination: 192.168.2.10
Internet Control Message Protocol
  Type: 0 (Echo (ping) reply)
  Code: 0
  Checksum: 0x0ee3 [correct]
  [Checksum Status: Good]
  Identifier (BE): 6 (0x0006)
  Identifier (LE): 1536 (0x0600)
  Sequence number (BE): 0 (0x0000)
  Sequence number (LE): 0 (0x0000)
  Data (68 bytes)
    Data: 000000000058cea4abcdabcdabcdabcdabcdabcdabcd...
    [Length: 68]

```


request:

```
ca 02 19 b3 00 38 ca 03 19 c2 00 38 08 00 45 00
00 64 00 1e 00 00 ff 01 f3 70 c0 a8 02 0a ac d9
10 2e 08 00 af 47 00 06 00 00 00 00 00 00 00 58
ce a4 ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd
```

Na czerwono adres MAC odbiorcy. Na niebiesko adres MAC nadawcy.
Następnie 08 00 to wersja protokołu komunikacyjnego IP.

```
ca 02 19 b3 00 38 ca 03 19 c2 00 38 08 00 45 00
00 64 00 1e 00 00 ff 01 f3 70 c0 a8 02 0a ac d9
10 2e 08 00 af 47 00 06 00 00 00 00 00 00 00 58
ce a4 ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd
```

Na fioletowo została zaznaczona długość pakietu, czyli 64₁₆.

```
ca 02 19 b3 00 38 ca 03 19 c2 00 38 08 00 45 00
00 64 00 1e 00 00 ff 01 f3 70 c0 a8 02 0a ac d9
10 2e 08 00 af 47 00 06 00 00 00 00 00 00 00 58
ce a4 ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd
```

Na czerwono zaznaczony został adres nadawcy czyli 172.217.16.46
Na zielono zaznaczony został adres odbiorcy czyli 192.168.2.10

```
ca 02 19 b3 00 38 ca 03 19 c2 00 38 08 00 45 00
00 64 00 1e 00 00 ff 01 f3 70 c0 a8 02 0a ac d9
10 2e 08 00 af 47 00 06 00 00 00 00 00 00 00 58
ce a4 ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd ab cd ab cd ab cd ab cd ab cd ab cd ab cd
ab cd
```

Na pomarańczowo zostały zaznaczone flagi. Po flagach znajdują się TTL równe 255.

Podczas wysyłania i odbierania zmienia się wartość TTL

```
ry response 0x0004 A google.com A 172.217.16.46
request id=0x0008, seq=0/0, ttl=255 (no
reply id=0x0008, seq=0/0, ttl=44
request id=0x0008, seq=1/256, ttl=255
reply id=0x0008, seq=1/256, ttl=44
request id=0x0008, seq=2/512, ttl=255
```