

Warsztat 7

mgr inż. Jan Palimąka

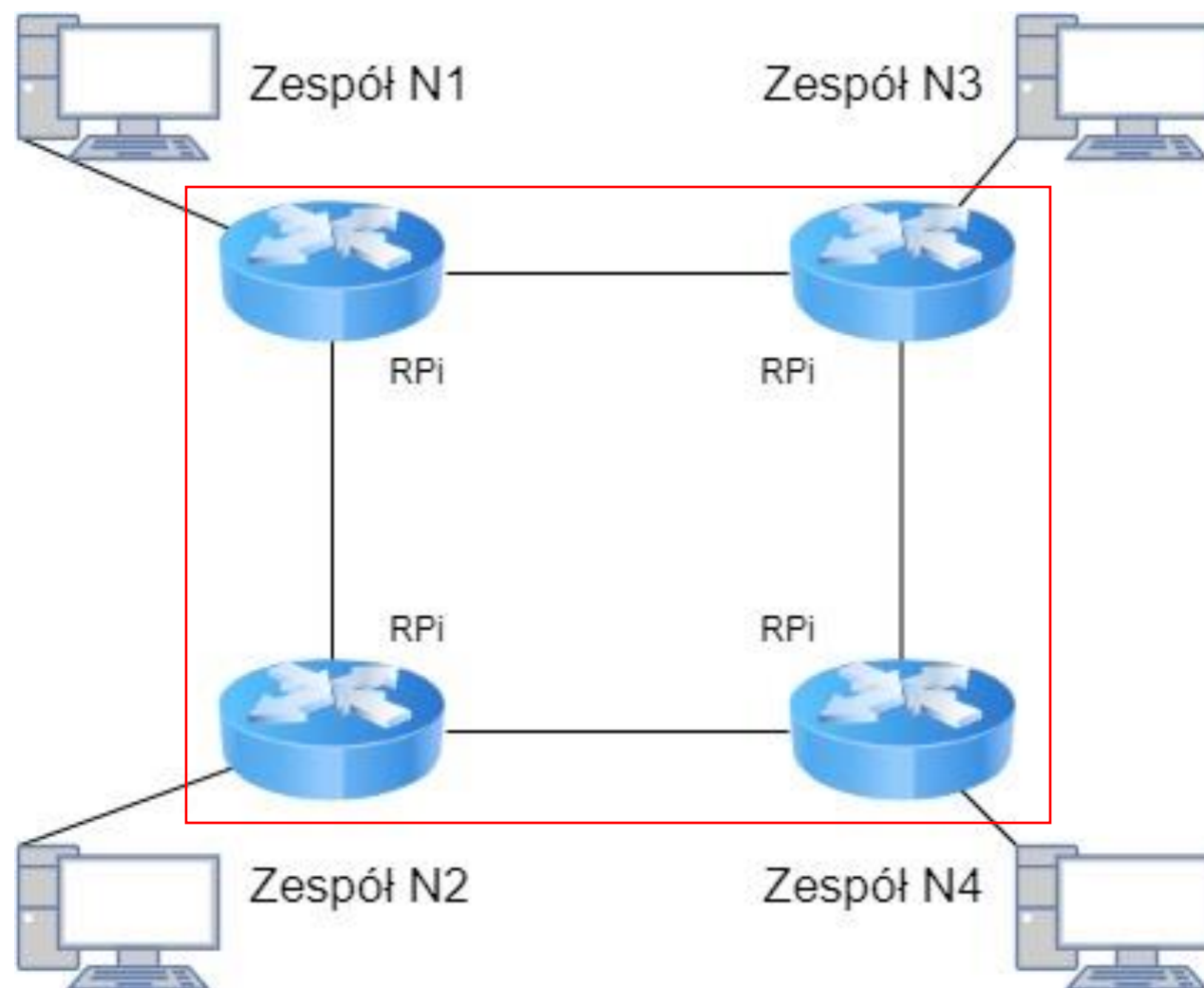
Płaszczyzna sterowania

- Podejmuje działania, których nie może podjąć program P4
 - Decyzje związane z routingiem
 - Obsługa skomplikowanych protokołów (zmienna długość nagłówka) - OSPF
 - Dodawanie/modyfikacja/usuwanie wpisów w tabelach
- Komunikacja za pomocą P4Runtime

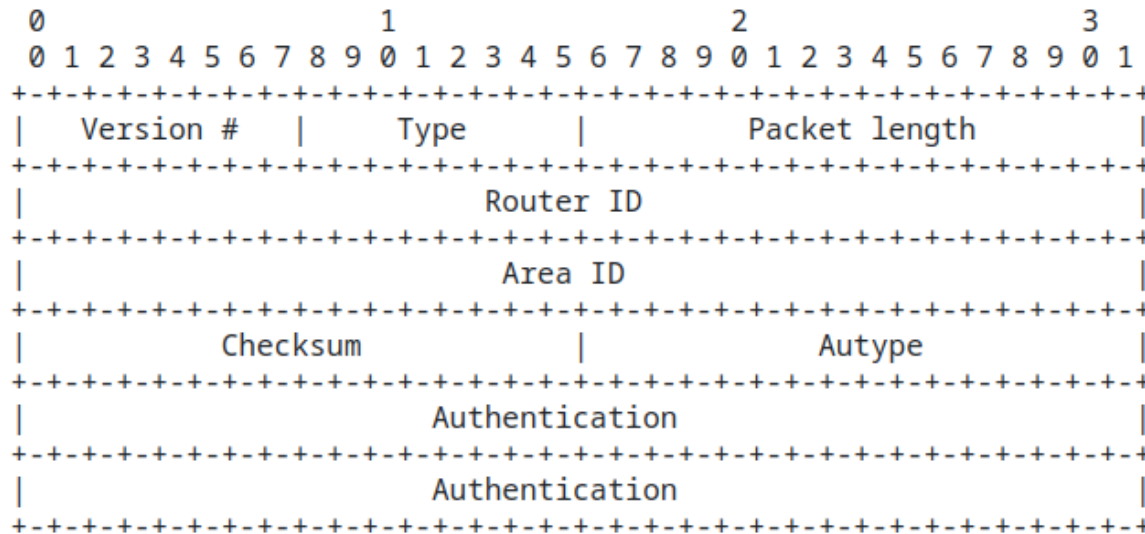
Protokół OSPF (uproszczony)

- Format nagłówków zgodny z OSPFv2
 - RFC 1247
 - Można użyć programu Wireshark do analizy pakietów
- Obsługa 2 rodzajów wiadomości
 - Hello
 - Link State Update (LSU)
 - Brak obsługi pozostałych wiadomości (można je ignorować/odrzucać)

Obszar działania OSPF



Nagłówek OSPF



- Packet length - długość całej wiadomości OSPF (począwszy od pola Version)
- Pole Authentication może zawierać *cokolwiek*, jeśli Autype jest równe 0
- Checksum – suma kontrolna wyliczana w taki sam sposób jak w IP/TCP/UDP (bez pola Authentication)

Walidacja nagłówka przychodzących wiadomości OSPF

- Wersja: 2 (zawsze)
- Suma kontrolna
- ID obszaru (Area ID)
- Rodzaj uwierzytelniania (Autype): 0 (brak)
- Niewłaściwe wartości powinny skutkować odrzuceniem pakietu i zignorowaniem go

Wiadomość Hello

- Służy do odkrywania sąsiednich routerów i wykrywania aktywności
- Nie podlega routingowi – nie jest przekazywana dalej na routerach
- Wysyłanie
 - Wysyłana cyklicznie co `HelloInt` sekund (np. co 30s)
 - Dedykowany wątek w sterowniku, który będzie w pętli wysyłał pakiet na odpowiednią grupę multicast, a potem usypiał się na `HelloInt` sekund
 - Adres docelowy IP to `AllRouters` (224.0.0.5)
 - Można stworzyć grupę multicast w celu wysyłania tej wiadomości
 - Adres docelowy MAC to broadcast (nieznany)
 - Adres źródłowy IP/MAC taki sam jak skonfigurowany na interfejsie, który ma wysłać wiadomość

Wiadomość Hello

- Odbieranie

- Jeśli sąsiedni router nie był aktywny (nie odebrano wiadomości Hello) w ciągu ostatnich $2 \times \text{HelloInt}$ sekund, to należy uznać za nieaktywny
 - Usunięcie z listy sąsiednich routerów
 - Aktualizacja tras
- Walidacja dodatkowych pól nagłówka OSPF: Network Mask, HelloInt
- Aktualizacja stanu sąsiedniego routera (dopasowanie: źródłowy adres IP) lub utworzenie nowego sąsiada
 - Sygnatura czasowa odebrania ostatniej wiadomości Hello

Wiadomość Hello

0										1										2										3											
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Version #										1										Packet length																					
										Router ID																															
										Area ID																															
Checksum																				Autype																					
Authentication																																									
Authentication																																									
										Network Mask																															
HelloInt																				Options										Rtr Pri											
										DeadInt																															
										Designated Router																															
										Backup Designated Router																															
										Neighbor																															
										...																															

- Pola od Options włącznie nieużywane
 - Ustawić wartość na 0
 - Pole Neighbor (jeśli jest) - ignorowane
 - Dzięki ich obecności można użyć Scapy do tworzenia/parsowania wiadomości

Wiadomość Link State Update

- Służy do budowy grafu topologii sieci przez każdy z routerów
- Wysyłana w momencie wykrycia zmiany topologii sieci przez jeden z routerów
 - Dołączył się nowy router
 - Jeden z sąsiednich routerów został oznaczony jako nieaktywny (awaria łącza, routera, itp.)
 - Po każdym wysłaniu należy zwiększyć numer sekwencyjny
- Wiadomości są propagowane (flood) do pozostałych routerów, więc po obsłużeniu wiadomości LSU nie ma potrzeby generowania własnej wiadomości LSU
- Adresy źródłowe IP i MAC odpowiadają tym skonfigurowanym na danym interfejsie (połączenie punkt-punkt) i są wysyłane bezpośrednio do sąsiedniego routera
 - Zmianie nie ulega router ID w wiadomości OSPF

Link State Update – obsługa wiadomości przychodzących

1. Jeśli wiadomość została wygenerowana przez ten router - odrzucić (używając pola Router ID)
2. Jeśli numer sekwencyjny zgadza się z ostatnio odebrany od tego routera (Router ID) - odrzucić, wiadomość była (jest) przetworzona
3. Aktualizacja stanu (topologia sieci, tabela routingu)
 - Algorytm Dijkstry (gotowa implementacja w pliku dijkstra.py)
4. Przesłanie wiadomości do wszystkich pozostałych sąsiednich routerów
 - Innych niż od tego, od którego otrzymano wiadomość
 - Zmiana adresów IP i MAC na odpowiednia na danych łączach
 - Obsługa pola TTL (IP i OSPF) - odrzucić, jeśli osiągnie wartość mniejszą lub równą 0

Wiadomość LSU

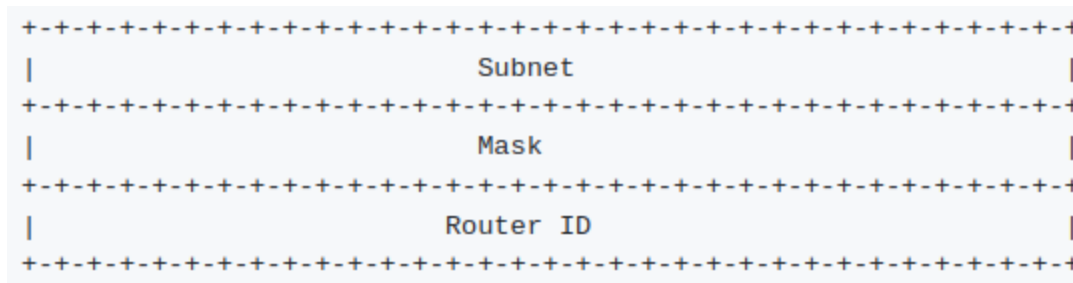
```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Version #   |                        4                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Router ID   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Area ID     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Checksum    |                        Autype                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Authentication
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Authentication
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Sequence    |                        TTL                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   # advertisements
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|
+-+
|   Link state advertisements
+-+
|   ...
|

```

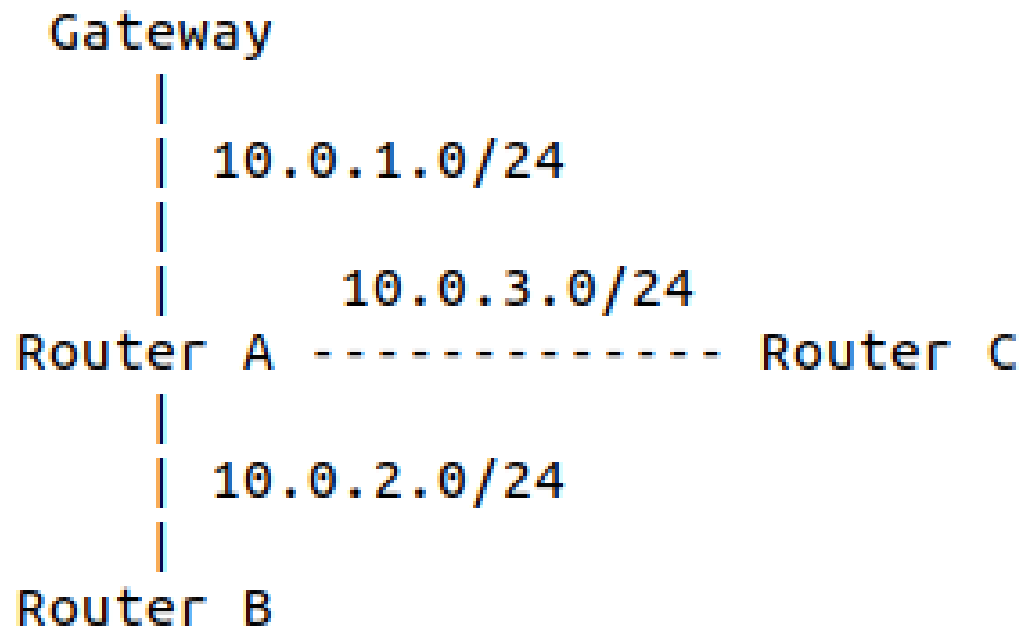
- Nie jest zgodny ze standardem
- Scapy może wymagać napisania obsługi tego nagłówka
 - Standardowa wersja nagłówka kończy się na polu Authentication

Link State Advertisement



- Może być powtarzany wielokrotnie w jednej wiadomości LSU
- Subnet – IP rozgłaszanej podsieci
- Mask – maska rozgłaszanej podsieci
- Router ID – ID sąsiedniego routera na danym łączy (0 jeśli go nie ma)
- Niestandardowy format nagłówka

Link State Advertisement - przykład tras



Router A powinien rozgłaszać następujące trasy:

- 10.0.1.0/24, RID: 0
- 10.0.2.0/24, RID: Router B
- 10.0.3.0/24, RID: Router C

Dzięki obecności RID, każdy router może zbudować pełny graf topologii sieci

Aktualizacja tras

- Dla każdego znanego routera jest wyznaczana najkrótsza ścieżka
 - Liczba węzłów pośredniczących, zakładamy każde łącze o takiej samej wadze
- Listę routerów posortować wg rosnącej liczby węzłów pośredniczących
- Po kolei dodawać wszystkie trasy rozgłaszane przez router z wyjątkiem wcześniej dodanych
 - Dzięki temu zostanie wybrana lepsza trasa, jeśli dwa routery rozgłaszają tą samą podsieć

Parametry protokołu OSPF

- HelloInt = 30s
- Area ID: 0.0.0.0
- Network Mask: 0.0.0.0
- Router ID: 10.0.0.NN
- Rozgłaszane trasy: 10.NN.0.0/16
- Podsieci na łączach między interfejsami: 172.16.NN.(X*4)/30

Gdzie NN to numer (jednego z) zespołu z zapisów, a X to dowolna liczba całkowita z przedziału [0; 63]

Przydatne materiały

- Specyfikacja OSPFv2 - <https://datatracker.ietf.org/doc/html/rfc1247>
- PWOSPF - <https://github.com/p4lang/p4pi/wiki/PWOSPF>
- Implementacja nagłówków OSPF w Scapy: <https://github.com/secdev/scapy/blob/master/scapy/contrib/ospf.py>

Ocenianie projektu

- Punkty przyznawane na koniec projektu
 - Kod źródłowy płaszczyzny przekazu danych (P4) - 14 pkt
 - Ethernet, ARP, IPv4, OSPF, MAC Learning, routing/forwarding, multicast – po 2 pkt
 - Kod sterownika (python) - 20 pkt
 - Wysyłanie/odbieranie wiadomości Hello/LSU - po 5 pkt
 - Uruchomienie – 11 pkt
- Sprawozdanie
 - Kod źródłowy
 - Sposób uruchomienia
- Testów PTF brak