Politechnika Gdańska

WYDZIAŁ ELEKTRONIKI TELEKOMUNIKACJI I INFORMATYKI

Katedra: Teleinformatyki

Imię i nazwisko dyplomanta: Jan Niedźwiedź

Nr albumu: 102127

Forma i poziom studiów: jednolite magisterskie

Kierunek studiów: Informatyka – Sieci komputerowe

Wyciąg z pracy dyplomowej magisterskiej na potrzeby przygotowania się studentów do zajęć laboratoryjnych

Temat pracy: Analiza porównawcza bezpieczeństwa protokołów routingu dynamicznego – ćwiczenie laboratoryne

Kierujący pracą: dr hab. inż. Wojeciech Molisz, prof. nadzw. PG

Konsultant pracy: mgr inż. Tomasz Gierszewski

Gdańsk, 2010 rok

1 Spis treści

2 Stosy sieciowe

Każdy system używający połączenia sieciowego używa stosu sieciowego. Stos jest podzielony na warstwy i każda warstwa zawiera zbiór protokołów posiadających właściwą funkcjonalność. Stosy to pojęcia koncepcyjne, nie fizyczne. Różne stosy mogą być kompatybilne w różnych rodzajach sieci.

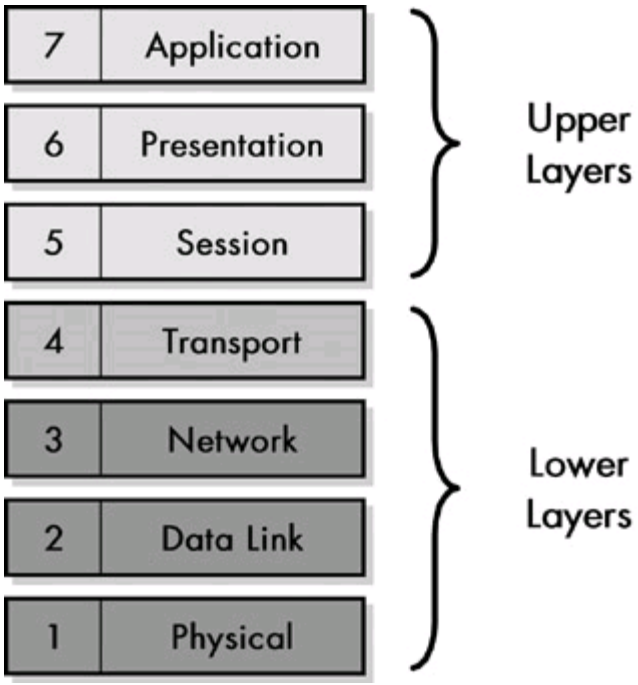
Dwa główne stosy to ISO OSI oraz TCP/IP. Oba definiują warstwy realizujące odpowiednie zadania. W każdej warstwie zdefiniowane są protokoły i standardy służące wymianie informacji.

Stosy mają tę własność, że mogą być używane jednocześnie, sekwencyjnie lub na przemian, w celu zaprezentowania dodatkowej funkcjonalności lub gdy użycie pojedynczego modelu jest niewystarczające.[1]

2.1 Stos ISO OSI

Stos został opisany przez międzynarodową organizacje ISO w 19xx roku. Potrzeba zbudowania takiego modelu wynikała z kilku celów jakie chciano osiągnąć. Były to: ułatwienie treningu i sporządzania dokumentacji, specjalizacja protokołów w wykonywaniu zadań, łatwiejsza modyfikacja stosu oraz modularność.

Model OSI zbudowany jest z siedmiu teoretycznych warstw, ponumerowanych od 1 do 7. Im niższy numer tym bliżej warstwie do sprzętu na którym pracuje protokół.

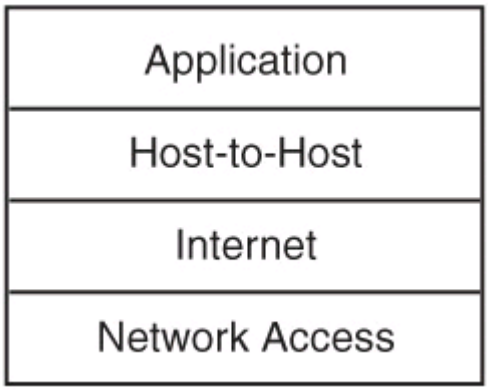


Każda z warstw oraz protokoły im przypisane spełniają określone zadania, aby wspólnie zrealizować cel jakim jest przesłanie określonej porcji danych.[2]

2.2 Model TCP/IP

Model ten jest główną nazwą zestawu protokołów zaprojektowanych przez Departament Obrony Stanów Zjednoczonych(DoD) w 1970 roku w celu wsparcia tworzeniu sieci Internet. Internet jest oparty na protokole TCP/IP.

Model stosu TCP/IP jest podobny do modelu OSI, jednak zamiast siedmiu, definiuje tylko cztery warstwy.



- Warstwa aplikacji – zawiera aplikacje oraz procesy które używają połączenia sieciowego
- Warstwa Host-to-Host – zapewnia połączenie z jednego końca do drugiego, dla warstwy aplikacji
- Warstwa Internet – Definiuje pakiet IP oraz zapewnia trasowanie danych w sieci
- Warstwa sieci – zapewnia fizyczny oraz elektryczny dostęp do medium.[3]

3 Opis protokołów routingu

Zawarty w tym rozdziale materiał omawia szczegółowo protokoły routingu dynamicznego, które poddane są analizie bezpieczeństwa w kolejnych rozdziałach.

3.1 Rodzaje protokołów routingu dynamicznego

Protokoły routingu dynamicznego możemy podzielić na dwie główne grupy ze względu na sposób działania, tj. stanu łącza oraz wektorowo-odległościowe. Oba rodzaje protokołu charakteryzują się pewnym wspólnym zestawem cech, bez względu na to, który standard w zadanej grupie rozpatrujemy.

Odpowiednio dla protokołów:

- Wektorowo – odległościowych

  Są to:

  - Przekazywanie całej tablicy routingu do sąsiadujących routerów, na podstawie których sąsiedzi budują swoją tablice routingu
  - Używają wszelakich timerów, aby zapobiegać pętlom routingu
  - Mają wiedze tylko o routerach/sieciach przylegających do routera bezpośrednio
- Stanu łącza

  Są to:

  - Przekazywanie topologii sieci do swoich sąsiadów
  - Uaktualnienia wysyłane są na skutek zachodzącej zmiany w sieci, a nie periodycznie

- Routery mają wiedzę o topologii całej sieci, nie tylko bezpośrednio przylegających routerach/sieciach[4]

3.2 RIP

RIP(ang. Routing Information Protocol) jest najstarszym wewnętrznym protokołem, który nadal jest powszechnie używany i szeroko wspierany, pomimo iż nie jest on odpowiedni dla dużych sieci.

3.2.1 Wprowadzenie do RIP

RIP był pierwszym wewnętrznym protokołem routingu przeznaczonym do powszechnego użycia. Jest to protokół wektorowo-odległościowy idealny dla małych sieci, które nie posiadają nadmiarowych połączeń do innych sieci. Jak wiele technologii komputerowych, tak i RIP został stworzony przez firmę Xerox PARC (Palo Alto Research Center) w późnych latach 70 tych XX wieku.

RIP bazuje na algorytmie Bellmana-Froda, który oblicza sumaryczne metryki dla tras. Algorytm do liczenia metryki używa ilość przeskoków. Ilość przeskoków jest to ilość routerów(przeskoków) jakie musi pokonać pakiet aby dotrzeć do celu. Przy wielu możliwych trasach dla tego samego celu, protokół RIP wybierze więc drogę, która ma najmniej przeskoków, mimo iż niekoniecznie musi być ona najodpowiedniejsza ze względu np. na wysokość dostępnego pasma.

RIP komunikuje się poprzez port 520 przy użyciu protokołu UDP. Dystans administracyjny w urządzeniach firmy Cisco wynosi 120.

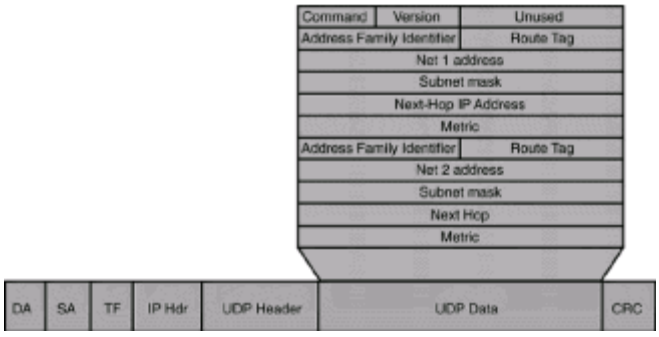
3.2.2 Wersje protokołu

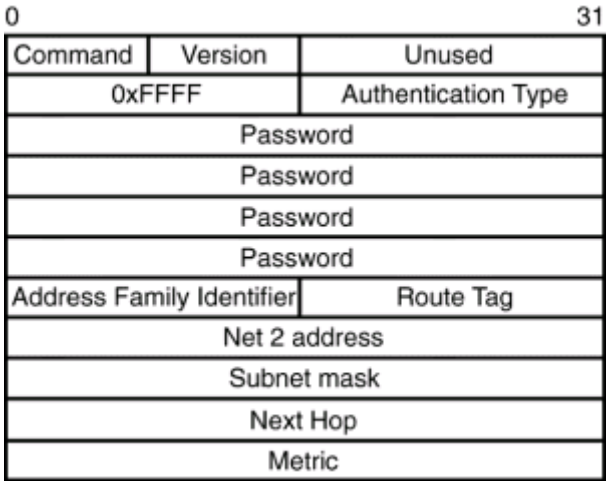
Opublikowane zostały dwie wersje protokołu RIP. Wersja 1(RIPv1) jest klasowym protokołem routingu, co oznacza że wspiera ograniczoną możliwość adresacji sieci. W czasach bezklasowej adresacji ta wersja nie jest rozważana jako realnie stosowana w środowiskach produkcyjnych.

Wersja 2(RIPv2), opublikowana w roku 1993 w dokumencie RFC 2453 dodała kilka unowocześnień. Jednymi z ważniejszych były:

- adresacja bezklasowa,
- dodanie maski sieci w uaktualnieniach przesyłanych pomiędzy routerami,
- możliwość budowania podsieci ze zmienną długością maski,
- definiowanie nieciągłych adresacji sieci
- automatyczna sumaryzacja sieci
- użycie adresu multicast do rozsyłania uaktualnień zamiast adresu broadcast, dając możliwość odbioru pakietu tylko przez zainteresowanych oraz redukując zużycie zasobów przez innych użytkowników sieci
- wsparcie autoryzacji pakietu prostym hasłem
- dodanie pola etykiety w nagłówku umożliwiającego przydzielanie pakietom etykiet
- dodanie pola następnego skoku, w celu ułatwienia integracji z protokołem OSPF oraz potencjalnego uniknięcia pętli routingu, gdyż sąsiad nadawcy mógł określić kto jest następny i czy to właśnie on[5]

3.2.3 Budowa pakietu RIP





Rysunek 1 - nagłówek protokołu RIPv2

3.2.4 Ogłaszanie tras

Kiedy router na którym uruchomiony jest protokół RIP zostanie uruchomiony, buduje sobie tablicę routingu zawierającą sieci połączone bezpośrednio do tegoż routera lub trasy wpisane na stałe. Następnie rozgłasza on swoją tablicę do wszystkich przylegających routerów. W RIP przylegające routery to takie, które współdzielą to samo połączenie. Innymi słowy są to routery oddzielone tylko o jeden przeskok.

Domyślnie RIP komunikuje swoją tablicę routingu co 30 sekund poprzez wszystkie swoje interfejsy. Każdy interfejs może być skonfigurowany do nie ogłaszania uaktualnień. To pozwala na uniknięcie wysyłania uaktualnień tam gdzie jest to niepotrzebne oraz do uniknięcia ujawnienia danych o sieci.

3.2.5 Nauka tras

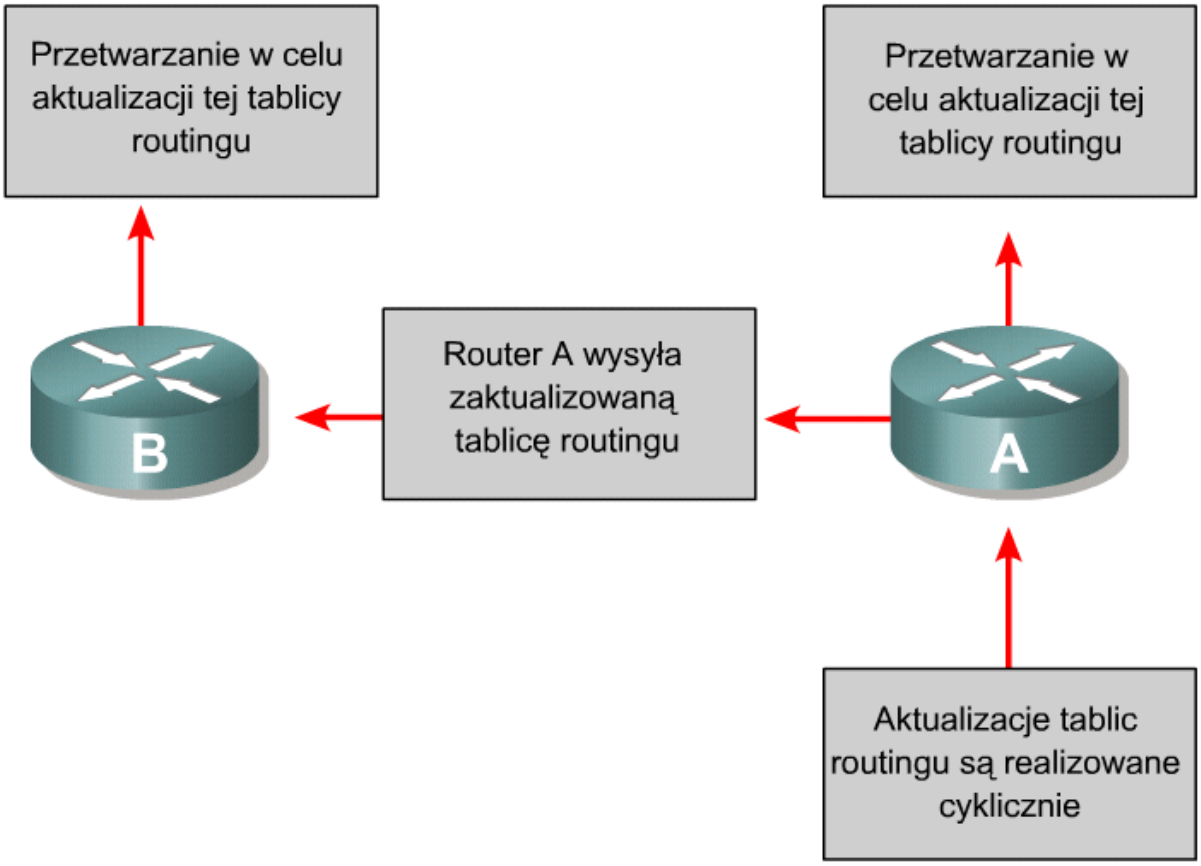
Nauka trasy odbywa się poprzez opisany w poprzednim podrozdziale mechanizm rozgłaszania tras. Kiedy router odbierze trasę, która jest jeszcze nieznana lub jest krótsza od istniejącej, dodaje ją do swojej tablicy routingu jednocześnie zwiększając odległość do celu o jeden.

Następnie router rozgłasza swoje trasy z zapisaną już nową trasą do routerów przylegających co określony czas.

Uaktualnienia RIP rozsyłane są kaskadowo wewnątrz systemu autonomicznego. Po inicjalizacji każdy router uczy się o scieżce do każdej sieci rozgłaszanej przez RIP. Kiedy następuje zmiana w sieci, router przekazuje uaktualnienie do swoich routerów przylegających, te następnie do swoich itd.

Czas po którym wszystkie routery mają tą samą wiedze o wszystkich trasach nazywa się czasem konwergencji, a sieć w tym stanie nazywa się znormalizowaną i konwergentną.

Nawet gdy sieć jest konwergentna, protokół RIP rozgłasza całą swoją tablicę routingu co 30 sekund, przez to potrzebuje całkiem dużej ilości pasma, co czyni go niezbyt optymalnym rozwiązaniem.



3.2.6 Informacje przechowywane o ścieżkach

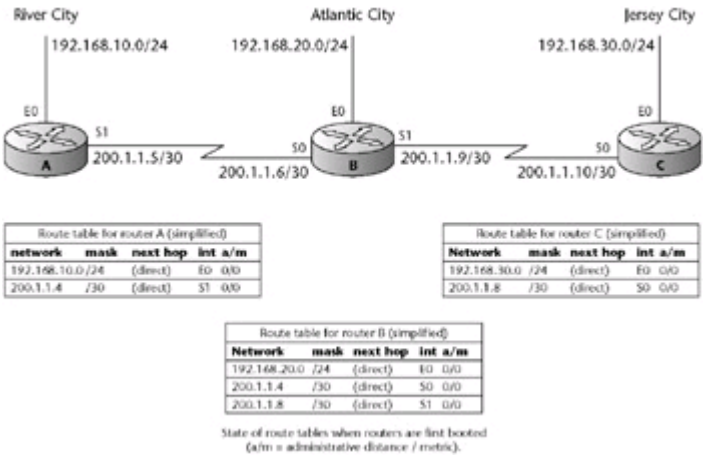
RIP przechowuje większość informacji nauczonych o każdej z tras.

Oto te informacje:

- Adres sieci docelowej
- Maska sieci docelowej
- Dystans administracyjny i metryka sieci docelowej
- Nazwa lokalnego interfejsu przez który można dotrzeć do następnego węzła
- Adres następnego węzła przez który można dotrzeć do sieci docelowej
- Liczniki dla każdej z tras

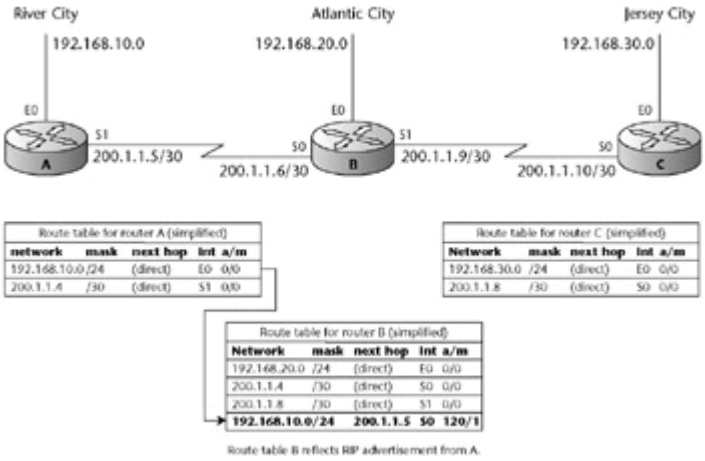
3.2.7 Sposób propagowania tablic routingu

Gdy router zostaje pierwszy raz uruchomiony, wie tylko o sieciach bezpośrednio połączonych do siebie(i tych statycznych), więc w tym momencie będzie wiadomo tylko o tych sieciach(Fig.1). W tym momencie stacja w sieci 192.168.10.0 nie ma szans dotrzeć do 192.168.20.0 i 192.168.30.0, zakładając, że nie są skonfigurowane żadne statyczne trasy. Należy zauważyć, że odległość od bezpośrednio połączonych sieci wynosi 0.



Rysunek 2 – Tablice routing zaraz po starce routerów.

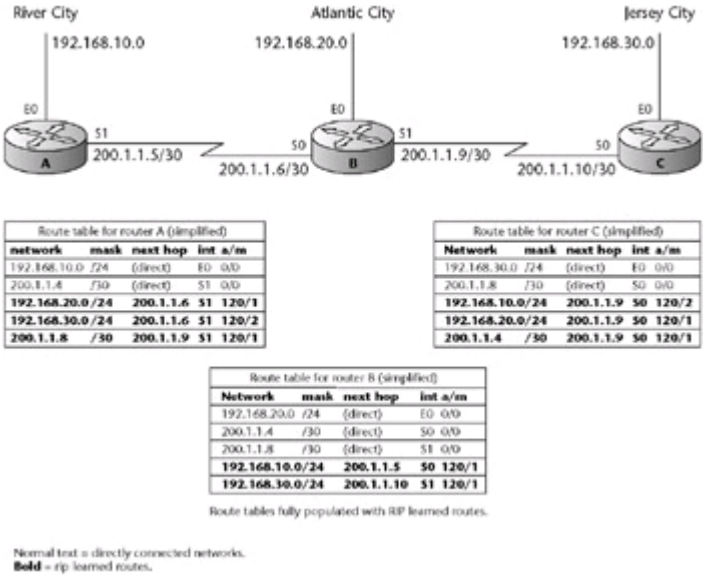
Następnie router A przesłał do routera B swoje trasy.



Rysunek 3 – Router A przesłał swoje trasy do routera B

Należy zauważyć, że:

- Router A wysłał swoje trasy do routera B. Router B skopiował te trasy do swojej tablicy routingu(pogrubione).
- Router B zwiększył odległość do nowych tras o 1.
- Router B zignorował zawiadomienie o trasie do 200.1.1.4, ponieważ posiadał już wpis w swojej tablicy o niższej metryce.



Rysunek 4 – Tablice routingu po pełnej wymianie informacji, gdy sieć jest konwergentna.

Należy zauważyć, że:

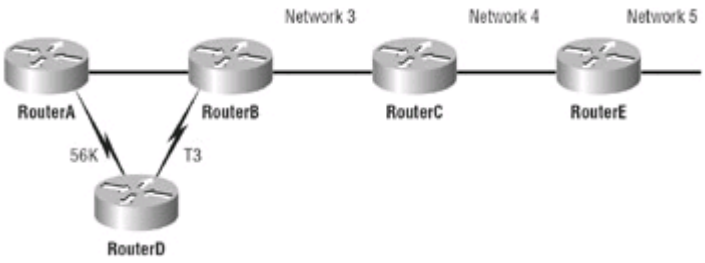
- Każdy router w sieci rozgłosił swoją całą tablicę. Każdy router jest teraz świadom każdej sieci. Sieć jest konwergentna.

- Router B działał jak przekaźnik, umożliwiając routerowi C naukę o sieciach dostępnych poprzez router A i vice versa. Routery A i C nigdy nie komunikowały się bezpośrednio

3.2.8 Pętle routingu

Protokoły dystansowo-odległościowe śledzą zmiany zachodzące w sieci poprzez okresowe rozgłaszanie swoich tablic routingu. Działa to dobrze, jednak pochłania znaczną ilość zasobów procesora routera oraz pasma. W momencie kiedy nastąpi awaria sieci, prawdziwy problem może się pojawić, biorąc pod uwagę powolne tempo konwergencji sieci przy używanym mechanizmie synchronizacji. Problemem stają się niejednakowe tablice routingu w domenie oraz pętle routingu.

Pętle routingu mogą powstać z powodu niejednoczesnego uaktualniania tablicy routingu na wszystkich routerach w domenie.



Rysunek 5 – Przykład pętli routing

Kiedy sieć nr 5 rozłączy się, router E ogłasza to routerowi C. To powoduje, że router C przestaje przekazywać pakiety do sieci nr 5 przez router E. Jednak routery A, B i D nie wiedzą o awarii w sieci nr 5, więc kontynuują wysyłanie uaktualnień tablicy routingu. Router C może ewentualnie wysłać uaktualnienie do B i zmusić go przez to do zaprzestania wysyłania pakietów do sieci nr 5, jednak router A i D wciąż nie mają aktualnej tablicy. Dla nich sieć nr 5 jest wciąż dostępna przez router B z metryką 3.

Problem powstaje w momencie, kiedy router A wysyła regularne uaktualnienie do przyległych sąsiadów, mówiące, że wciąż ma dostęp do sieci nr 5. W tym momencie routery B i D otrzymują wspaniałą wiadomość, że sieć nr 5 jest osiągalna przez router A, więc routery B i D rozgłaszają informacje, że sieć nr 5 jest znów dostępna.

Każdy pakiet kierowany teraz do sieci nr 5 przejdzie do routera A, B i powrotem do A. To jest właśnie pętla routingu.

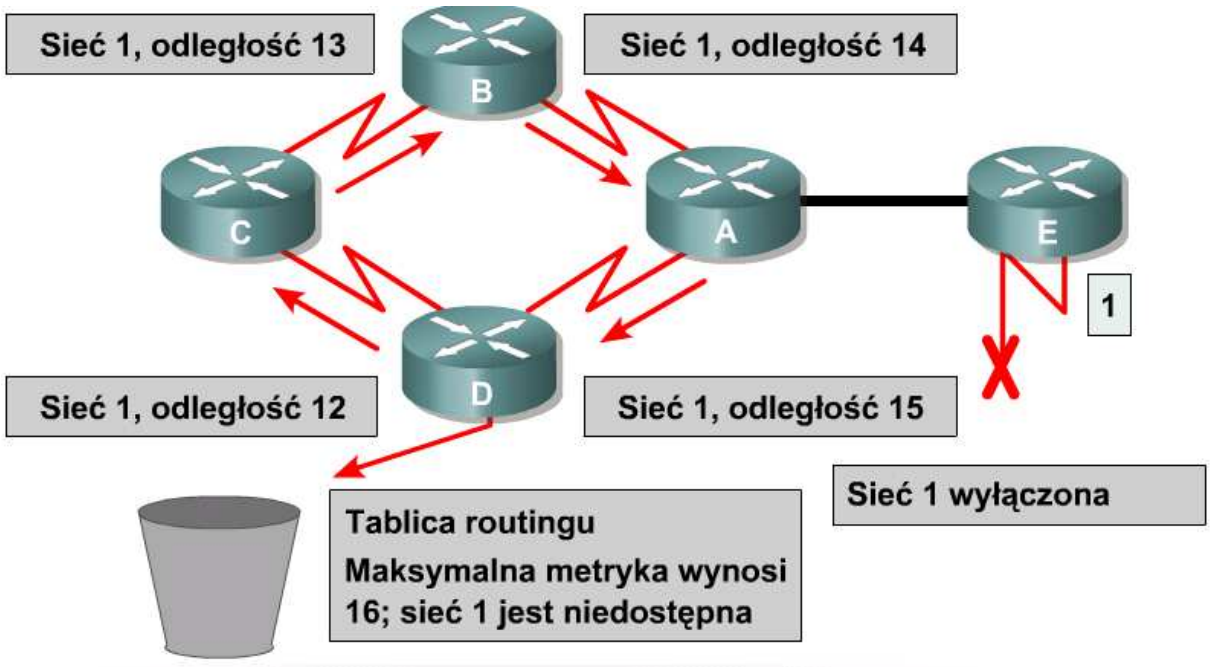
3.2.9 Metody przeciwdziałania pętlom routingu

Protokół RIP podczas działania, używa różnego rodzaju mechanizów w celu zapewnienia prawidłowego działania sieci i uniknięcia pętl routingu.

Maksymalna ilość przeskoków

Problem pętl nazywany jest również odliczaniem do nieskończoności(ang. Counting to infiniti) i jest spowodowany propagowaniem nieprawdziwych danych w domenie. Bez odpowiedniej interwencji, podczas pętli metryka wzrastałaby w niekończonośc.

Jedną z metod rozwiązania tego problemu jest zdefiniowanie maksymalnej metryki, jako maksymalnej ilości przeskoków w domenie. RIP pozwala, aby metryka dla prawidłowej trasy nie przekraczała liczby 15 przeskoków. Wszystko co wymaga 16 przeskoków lub więcej, jest uznawane za nieosiągalne. Dzięki temu pętla routingu istnieje do momentu przekroczenia metryki 15 przeskoków.

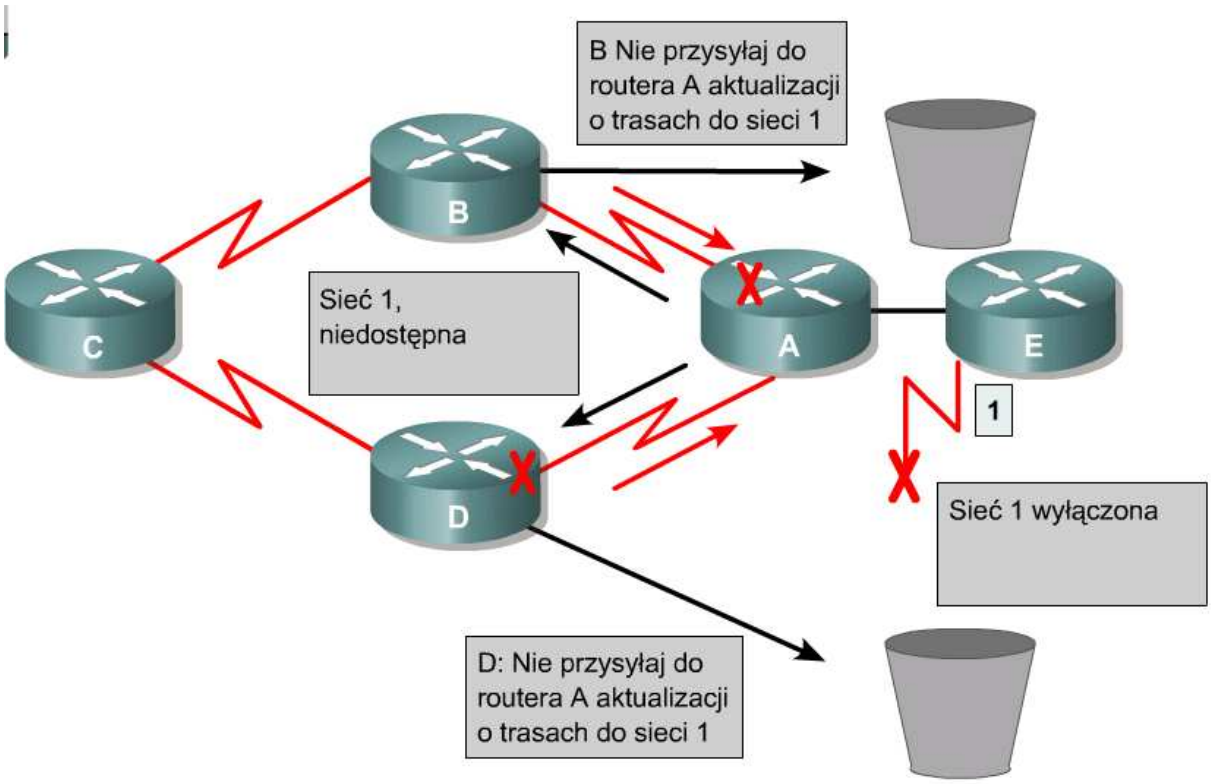


Rysunek 6 - przekroczenie dozwolonej ilości przeskoków

Split Horizon

Kolejnym rozwiązaniem problemu pętl routingu jest mechanizm Split Horizon. Polega on na tym, że w momencie, kiedy dana sieć przestaje być osiągalna, router bezpośrednio do niej

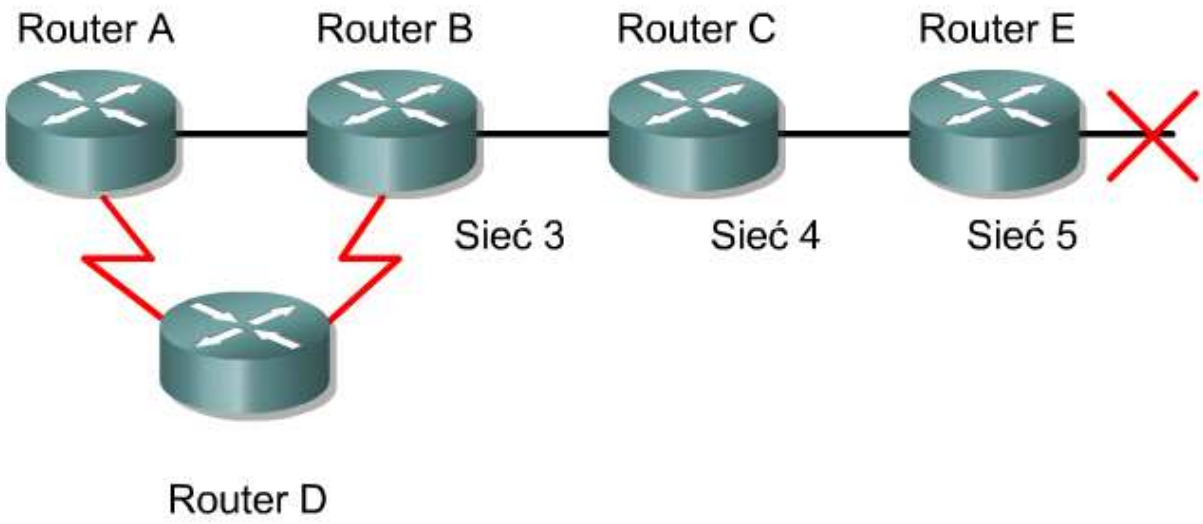
przyległy rozgłasza ją z metryką o wartości 16. Następnie routery do niego przyległe potwierdzają otrzymanie takiej trasy. Kluczowym aspektem tej metody jest to, że router który rozgłosił informację o nieosiągalnej sieci, nie akceptuje uaktualnień o tej sieci, z interfejsów, na które wcześniej wysłał tą informacje.



Rysunek 7 - Metoda działania mechanizmu Split Horizon

Zatruwanie trasy

Metoda route poisoning jest używana przez różne protokoły z wektorem odległości w celu zapobieżenia powstawaniu większych pętli routingu oraz dostarczenia dokładnych informacji o niedostępności podsieci lub sieci. W tym celu licznikowi przeskoków jest nadawana wartość o jeden większa niż maksymalna.



Metoda route poisoning jest jedną z metod zapobiegających niespójnym aktualizacjom. Gdy sieć 5 ulegnie awarii, router E ustawi odległość 16 dla sieci 5, aby zablokować tę trasę. Oznacza to, że sieć jest niedostępna. Gdy trasa jest zablokowana (ang. poisoned), router C nie bierze pod uwagę nieprawidłowych aktualizacji dotyczących trasy do sieci 5. Gdy router C otrzyma informację o zablokowaniu trasy od routera E, wyśle aktualizację o nazwie poison reverse do tego routera. Dzięki temu wszystkie routery w segmencie otrzymają informacje o zablokowanej trasie.

Metoda route poisoning używana w połączeniu z wyzwalanymi aktualizacjami skraca czas uzyskiwania zbieżności, ponieważ sąsiednie routery nie muszą czekać 30 sekund, aby wysłać ogłoszenie o zablokowanej trasie.

Metoda route poisoning powoduje, że protokół routingu ogłasza dla niedostępnej trasy metrykę nieskończoną. Metoda route poisoning nie jest sprzeczna z regułą split horizon. Reguła split horizon w połączeniu z metodą poison reverse jest to metoda route poisoning stosowana do łączy, przez które w przypadku stosowania reguły split horizon nie byłyby przesyłane informacje o routingu. W obu przypadkach niedostępne trasy są ogłaszane z metryką nieskończoną[6]

Licznik uaktualnień

Licznik ten mówi o tym, jak często wysyłane są pakiety uaktualnień tablicy routingu. Pakiety te rozgłaszane są poprzez wszystkie interfejsy biorące udział w procesie RIP na adres broadcast(RIPv1) lub multicast(RIPv2). Domyślna częstotliwość to 30 sekund dla routerów Cisco.

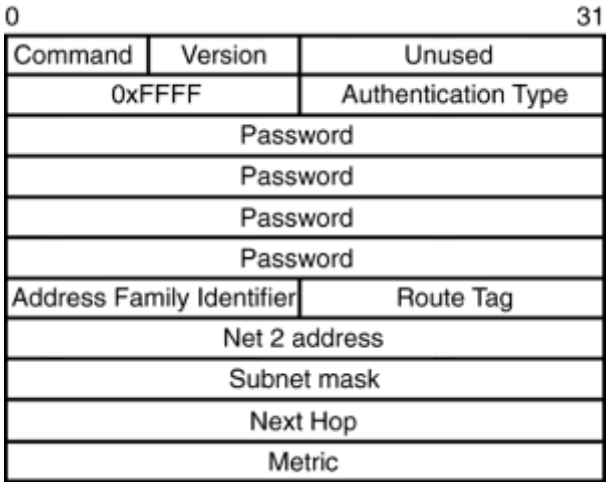
Licznik nieważności trasy oraz licznik czyszczenia pamięci, zatruwanie trasy

Routery na których działa proces RIP nie utrzymują raz nauczonych się wpisów bezterminowo. Proces ten utrzymuje stały kontakt z routerami przyległymi poprzez periodyczne otrzymywanie oraz wysyłanie uaktualnień tablicy routingu, gwarantując, że ogłaszane sieci dalej istnieją i są dostępne. Jeżeli uaktualnienie dla jednej lub wielu sieci przestaje przybywać, RIP zakłada, że te sieci przestały być osiągalne. Dla tego celu stworzone zostały dwa liczniki, nieważności trasy oraz czyszczenia pamięci. Oba są zerowane gdy otrzymają kolejne uaktualnienie. Utrzymywane one są z osobna dla każdej trasy i sprawdzają jej ważność z osobna.

Licznik nieważności odlicza czas po którym, przy nieotrzymywaniu uaktualnień o sieżce, oznacza ją jako nieprawidłową w tablicy routingu oraz rozgłasza ją z metryką 16. Jest to znane jako zatruwanie trasy(ang. Route poisoning). Następnie router kontynuuje rozgłaszanie tej trasy jako nieosiągalnej przez czas odliczany przez licznik czyszczenia pamięci. To daje routerowi czas na rozgłoszenie nieważności trasy, aż do momentu całkowitego usunięcia jej z pamięci. Kiedy upłynie czas odliczany przez licznik czyszczenia pamięci, trasa jest usuwana na stałe z tablicy routingu.
Licznik czyszczenia pamięci jest zawsze ustawiony na wyższą wartość niż czas nieważności, aby pozwolić na rozgłoszenie nieważnej trasy do przylegających routerów. Domyślne ustawienie dla licznika nieważności to 180 sekund(6 cykli uaktualnień). Domyślne ustawienie licznika czyszczenia pamięci to 240 sekund(60 sekund dłużej niż czas nieważności).

Domyślne ustawienia liczników można modyfikować, jednak oznacza to większe zużycie pasma, oraz potrzebę standaryzacji w organizacji. Należy też nadmienić, że ustawienia liczników powinny być takie same w jednej domenie rozgłoszeniowej.[7]

3.2.10 Autoryzacja



W protokole RIP nie było przeznaczonego miejsca na autoryzację, jednak gdy zaczęto go używać wraz z OSPF do poważniejszych zadań, stworzono miejsce na autoryzację. Pole AFI(ang. Address family identifier) jest używane do tego celu. Jeżeli pole AFI zawiera wartość 0xFFFF, wtedy pierwsza wartość w polach tras, zawiera hasło. Nagłówek zmienia się wtedy tak jak pokazano na rysunku. Typ autoryzacji ma wartość 2(zwykłe hasło) i następne 16 bajtów zawiera hasło(każda ilość bajtów, aż do 16). RIPv1 zignoruje ten pierwszy wpis, jeżeli AFI nie jest ustawione na żaden adres rodzaju IP.

Jeżeli router z RIPv2 jest skonfigurowany bez autoryzacji, zaakceptuje on pakiety RIPv1 oraz RIPv2 nieautoryzowane hasłem, jak i odrzuci pakiety autoryzowane hasłem. Jeżeli router jest skonfigurowany dla RIPv2 z autoryzacją, zaakceptuje on pakiety RIPv1 i RIPv2 jeżeli posiadają autoryzację.

Należy pamiętać, że nie każde implementacje RIPv1 są zgodne ze standardami RFC, gdzie różne są znaczenia pól w nagłówku. Kolejnym niebezpieczeństwem jest to, że zwykłe pakiety RIPv2 nie zostaną zaakceptowane.[8]

3.3 OSPF

Protokół w wersji pierwszej został oficjalnie opisany w dokumencie RFC1131 w październiku 1989 roku. Dwa lata później, w roku 1991, ze względu na postęp technologiczny

oraz braki w bezpieczeństwie został zastąpiony specyfikacją protokołu w wersji drugiej w dokumencie RFC1247.

Ze względu na to, iż protokół w tej wersji jest na tyle przestarzały, że producenci sprzętu już dawno zaprzestali jego implementacji a do istniejących urządzeń wydali stosowne uaktualnienia, protokół w tej wersji jest poza spektrum analizy tego opracowania.

Protokół w drugiej wersji jest najszerzej implementowany w środowiskach produkcyjnych i zostanie poddany najgłębszej analizie.

Implementacja protokołu została opisana w uaktualnianych kolejno dokumentach: RFC 1247 (rok 1991), RFC 1583 (rok 1994), RFC 2178 (rok 1997) oraz finalnie RFC 2328 (rok 1998).

Protokół w wersji trzeciej, opisany w RFC 2740, nie posiada szczególnych ulepszeń bezpieczeństwa, dlatego rozważania dla protokołu OSPF należy opierać na wersji drugiej opisanej w RFC 2328 i najszerzej stosowanej w produkcji.

Trzecia wersja protokołu została opisana i wydana głównie na potrzeby obsługi protokołu IPv6. Jego implementacja w urządzeniach Cisco na potrzeby redystrybucji informacji o ścieżkach została przedstawiona w Cisco IOS 12.0(24)S, 12.2(18)S oraz 12.2(15)T.

OSPF ma dwie główne charakterystyczne cechy. Jest on protokołem otwartym, co oznacza, że może być szeroko implementowany przez różnych producentów sprzętu. Drugą jest to, że jest oparty na algorytmie SPF(ang. Shortest Path First) nazywanym czasem algorytmem Dijkstry.

OSPF jest protokołem stanu łącza, który używa powiadomień LSA(ang. Link State Advertisment) do komunikacji z innymi routerami w tym samym obszarze. W momencie, kiedy routery mają już informacje o sieci, używają algorytmu SPF do obliczenia najkrótszych ścieżek do każdej lokacji.

3.3.1 Pojęcia OSPF

W celu wyjaśnienia jak działa OSPF oraz jak można go zaatakować, należy zapoznać się z podstawowymi pojęciami używanymi w opisie tego protokołu.

- **Stan łącza(ang. Link-state)**

  Jest to opis interfejsu routera i jego relacji z routerami bezpośrednio podłączonymi do niego. Na opis składają się:

  - IP adres interfejsu routera
  - Maska sieci
  - Typ podłączonej sieci

- Routery podłączone do tej sieci

- **Baza stanu łącza(ang. Link-state database)**

  Przechowywane w niej są informacje o stanie wszystkich połączeń w sieci. Baza ta musi być identyczna dla wszystkich routerów w tym samym obszarze.

- **Baza sąsiedztwa(ang. Adjacency database)**

  Przechowuje listę routerów z którymi została nawiązana dwustronna komunikacja i ustalone zostanie sąsiedztwo. Ta baza danych jest unikalna dla każdego z routerów.

  Dwa routery nawiążą relacje sąsiedztwa po wzajemnej wymianie informacji oraz gdy ich bazy stanu łącza będą identyczne. Cała procedura jest podzielona na 8 faz:

  1. Down

     Routery nie otrzymały żadnych informacji przez swoje interfejsy
  2. Attempt

     Używane w chmurach NBMA(Frame Relay, X.25). Pakiety Hello są wysyłane z wyższą częstotliwością
  3. Init state

     Pakiety Hello zostały wysłane. Oczekiwanie na odpowiedź.
  4. Establish Bi-directional (two-way)

     Zakończona faza wymiany pakietów Hello i wyboru routerów DR i BDR.
  5. Exstart

     Relacja Master / Slave została ustalona.

     Numery sekwencyjne LSA zostały uzgodnione.
  6. Exchange state

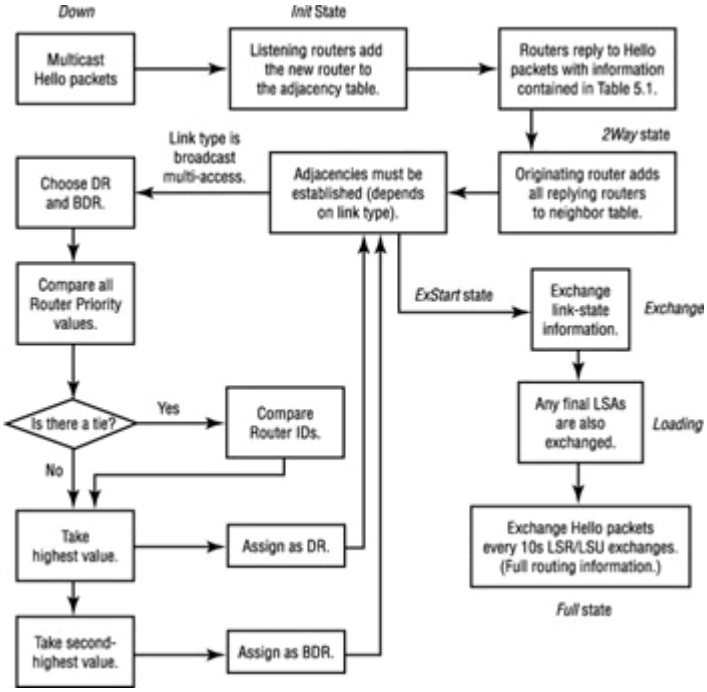
     Dwa routery posiadają ustaloną relacje Master / Slave. Master wysyła pakiety DD(Database Description) a Slave je odbiera I potwierdza odbiór.
  7. Loading state

     Router przy pomocy pakietów Link-State Request odpytuje sąsiadów o jakiekolwiek informacje, które mogą istnieć a router nie posiada ich w bazie stanu łącza.

8. Full state

   Oba routery posiadają te same bazy stanu łącza. W tym momencie są w pełnej relacji sąsiedztwa.

Poniższy rysunek przedstawia algorytm inicjalizacji relacji przylegania



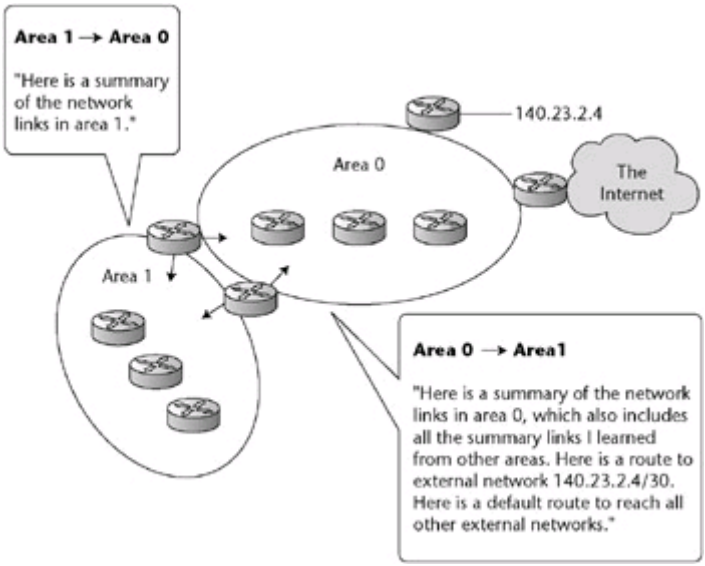
Rysunek 8 - Algorytm inicjalizacji relacji przylegania

- **Area**

  Każdy router OSPF musi zostać przypisany do obszaru. Każdy obszar definiowany jest przez ID, które jest 32-bitową liczbą, wyrażoną w postaci adresu IP lub liczby dziesiętnej. Obszar 0 jest rdzeniem, który jest wymagany, gdy mamy więcej obszarów w sieci niż jeden. Wszystkie inne obszary muszą być połączone do obszaru 0, z wyjątkiem użycia wirtualnych połączeń(ang. Virtual Links)

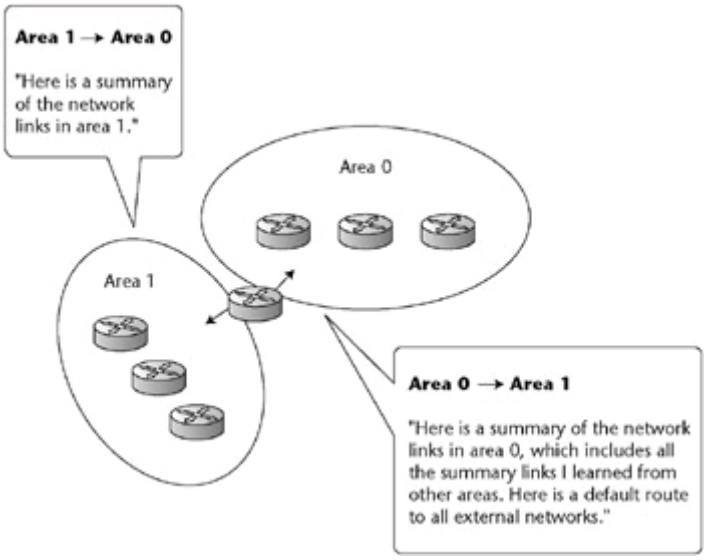
  Rozróżniamy 5 typów obszarów:

  - **Standard area**



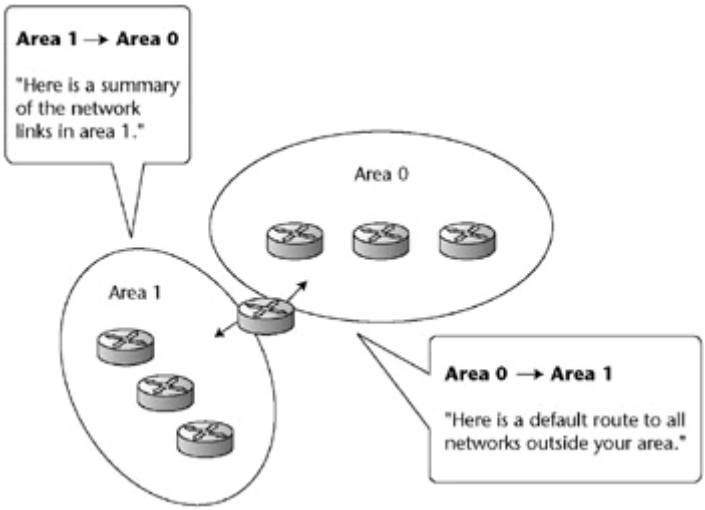
Rysunek 9 - Przykład topologii sieci ze standardowymi obszarami

- **Stub area**



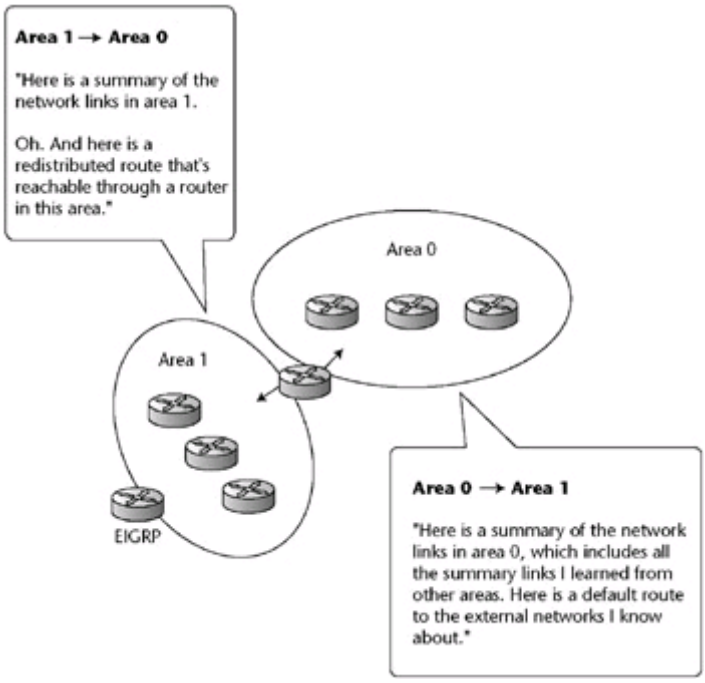
Rysunek 10 - Przykład topologii z obszarem Stub

- **Totally stubby area (TSA)**



Rysunek 11 - Przykład topologii z obszarem Totally Stubby

- **Not-so-stubby area (NSSA)**



Rysunek 12 - Przykład topologii z obszarem Not So Stubby

- **Totally not-so-stubby area (TNSSA)**

- **Cost**

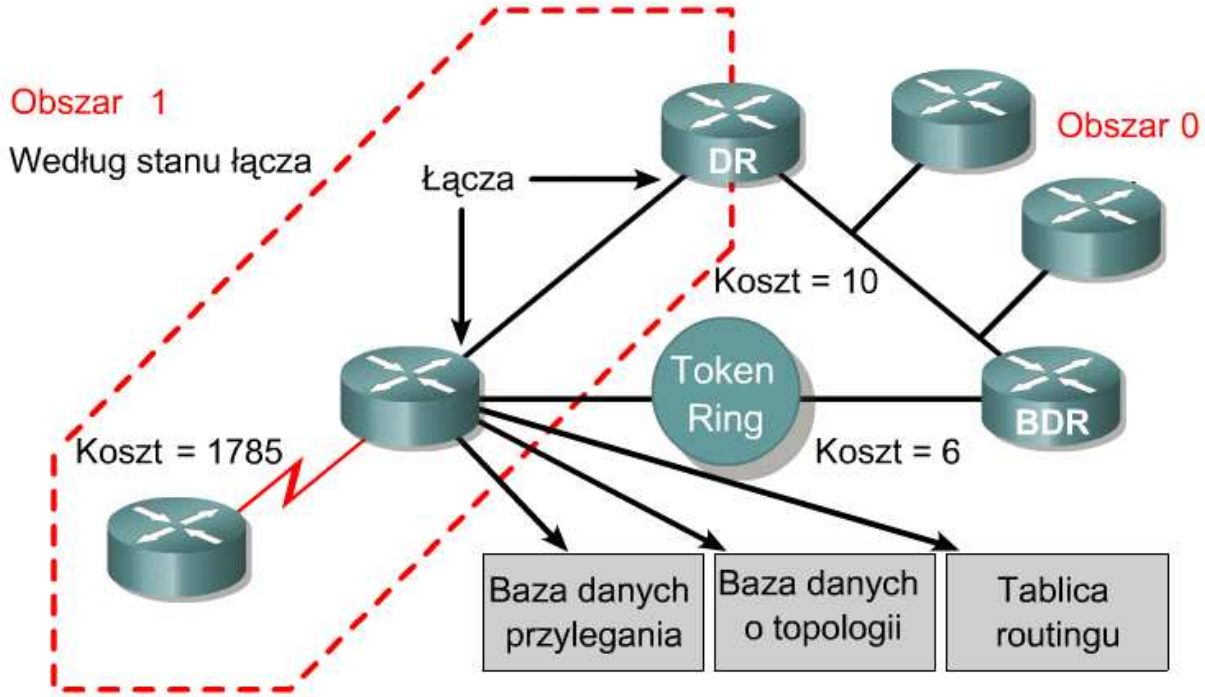
  Koszt jest metryką OSPF. Formuła na obliczenie metryki to 10^8/bandwidth. Jest oczywiste, że niższa wartość jest lepsza. Koszt można również ustawić ręcznie.

- **Designated Router(DR)**

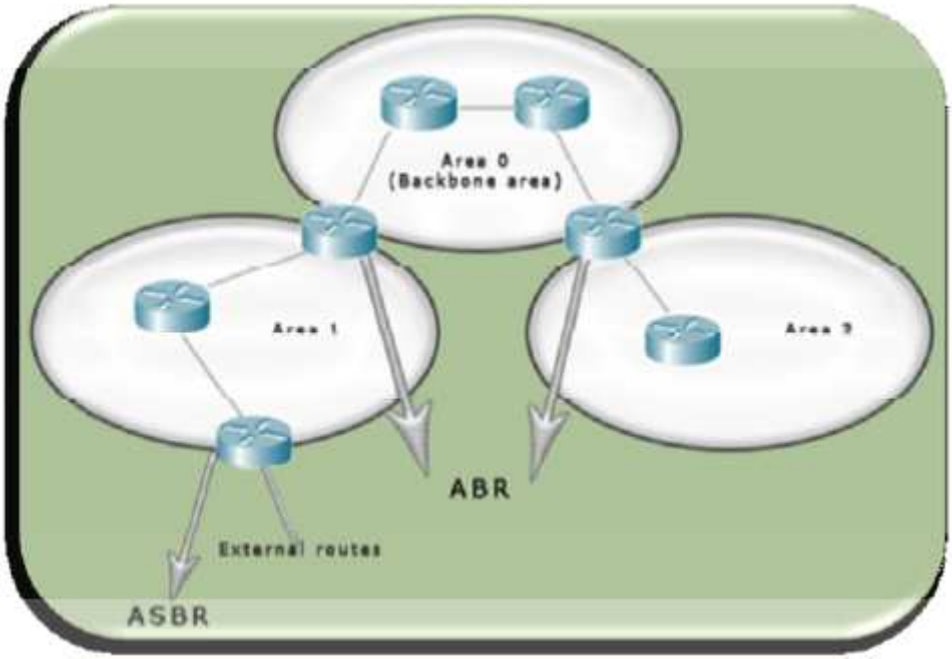
Dla każdego routera możemy ustawić priorytet. Router z najwyższym priorytetem będzie wybrany jako DR. W sytuacji, gdy dwa lub więcej routerów ma przypisany ten sam priorytet, wybrany będzie router z najwyższym Router ID. Router ID jest to najwyższy(jako numer) adres IP, który jest przypisany do routera na jakimkolwiek interfejsie. DR jest używane ze względów wydajnościowych, aby uniknąć parowania się routerów każdy z każdym, istnieje w sieci router który odbiera od wszystkich informacje i wysyła je do wszystkich.

- **Backup Designated Router(BDR)**
  Jest to router który jest wybierany jako DR w sytuacji, gdy DR ulegnie awarii.
- **Area Border Router(ABR)**
  Router, który posiada 2 lub więcej interfejsów, należących do różnych obszarów
- **Autonomous System Border Router(ASBR)**
  Router który używa OSPF jako protokołu na jednym interfejsie a innego protokołu routingu na drugim(np. drugiej instancji OSPF, BGP)[9]

W kolejnych rozdziałach pojawią się także nowe pojęcia wprowadzane przy okazji omawiania szczegółów protokołu.



Rysunek 13 - Zestawienie pojęć OSPF 1



Rysunek 14 - Zestawienie pojęć OSPF 2

3.3.2 Algorytm SPF

Algorytm SPF(ang. Shortest Path First) lub inaczaj algorytm Dijkstry, jest podstawą funkcjonowania protokołu OSPF. Jest to algorytm odnajdywania najkrótszej scieżki, gdzie miarą długości scieżki może być metryka o dowolnym znaczeniu w rzeczywistości.

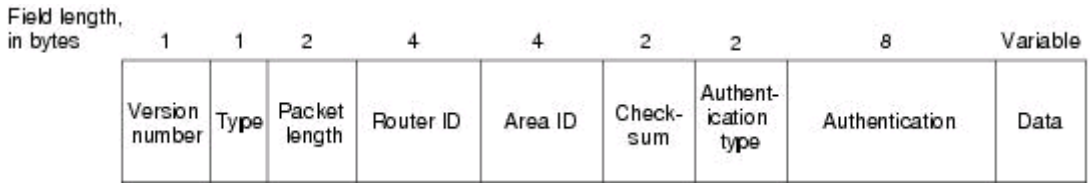
Algorytm uruchomiony na zadanym routerze, umieszcza go w centrum zbudowanego przez siebie drzewa i oblicza najkrótszą ścieżkę do każdego węzła docelowego, bazując na łącznej odległości do niego(koszcie).

Każdy router będzie miał zbudowaną przy pomocy tego algorytmu własną topologie najkrótszych tras i będzie ją utrzymywał niezależnie od innych routerów, mimo iż zbuduje tą topologie na podstawie tych samych informacji co posiadają inne węzły.[10]

3.3.3 Budowa pakietu

Wszystkie pakiety OSPF rozpoczynają się 24-bajtowym nagłówkiem. Nagłówek jest identyczny dla wszystkich rodzajów pakietów. Nagłówek jest istotny ze względów

bezpieczeństwa, gdyż przechowuje on dane autoryzacyjne. Pole autoryzacji może mieć różny format w zależności od wybranego rodzaju, co więcej, pole sumy kontrolnej przyjmuje wartość 0000, jeżeli została wybrana autoryzacja szyfrowana.



Rysunek 15 - nagłówek OSPF

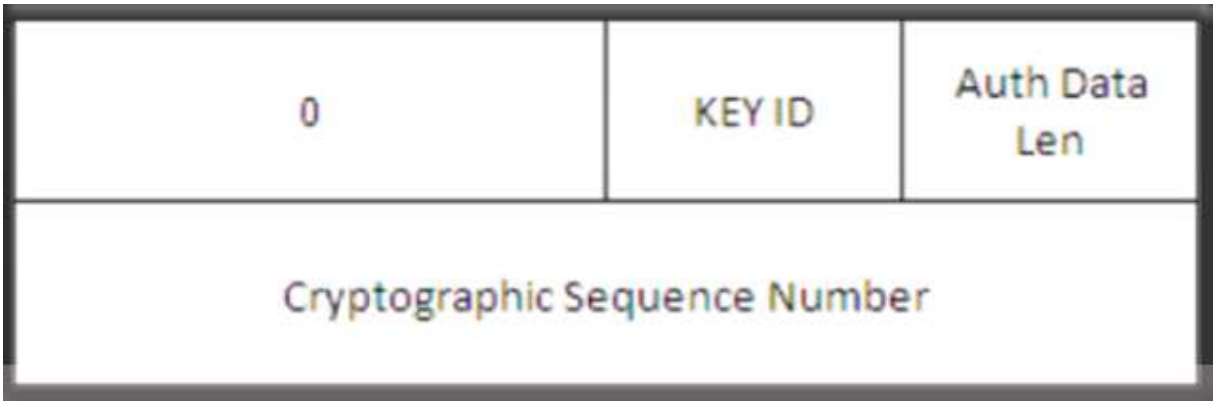
Nagłówek OSPF posiada następujące pola: (Rysunek 15 - nagłówek OSPF)

- **Numer wersji** – identyfikuje wersję protokołu OSPF
- **Typ** – identyfikuje typ pakietu jako jeden z następujących:
  - Hello – Nawiązuje i utrzymuje relacje przylegania
  - Database description – Opisuje zawartość bazy topologii. Ta wiadomość wymieniana jest w momencie inicjalizacji przylegania
  - Link-state request(LSR) – Odpytuje o szczegóły dotyczące wpisu w bazie topologii. Ten pakiet wysyłany jest w momencie stwierdzenia przez router(poprzez przegląd bazy topologii), że pewien wpis jest nieaktualny
  - Link-state update(LSU) – Odpowiedz na Link-state request. Ten pakiet odpowiada za regularną komunikacje z routerami w tym samym obszarze. Kilka uaktualnień LSA może być zawartych w jednym pakiecie LSU.
  - Link-state acknowledgment – Potwierdza odbiór pakietu LSU.
- **Długość** – wskazuje długość pakietu wyrażoną w bajtach, uwzględniając nagłówek OSPF.
- **Router ID** – Identyfikuje nadawce pakietu.
- **Area ID** – Identyfikuje obszar do którego należy pakiet. Wszystkie pakiety OSPF są powiązane z jednym obszarem.
- **Suma kontrolna** – Umożliwia weryfikacje czy pakiet dotarł w całości do odbiorcy.
- **Typ autoryzacji** – Zawiera typ wymaganej autoryzacji pakietu. Wszystkie wymiany pakietów OSPF są autoryzowane. Rodzaj autoryzacji można konfigurować dla każdego obszaru osobno. Wyróżniamy następujące typy autoryzacji:

- 0 – brak autoryzacji
- 1 – autoryzacja hasłem przesyłanym czystym tekstem
- 2 – autoryzacja szyfrowana

- **Autoryzacja** – zawiera dane wymagane do autoryzacji.[11]

Jeżeli została wybrana autoryzacja szyfrowana, wtedy pole Autoryzacja ma odmienny format niż w przypadku, gdy została użyta autoryzacja czystym tekstem.
Dane autoryzacyjne (16 bajtów w przypadku MD5) są dołączane na końcu pakietu, bądź przed blokiem danych LLS, jeżeli użyto rozszerzenia LLS dla OSPF.
Poniżej rysunek przedstawiający budowę pola autoryzacji wraz z krótkim objaśnieniem ich znaczenia:

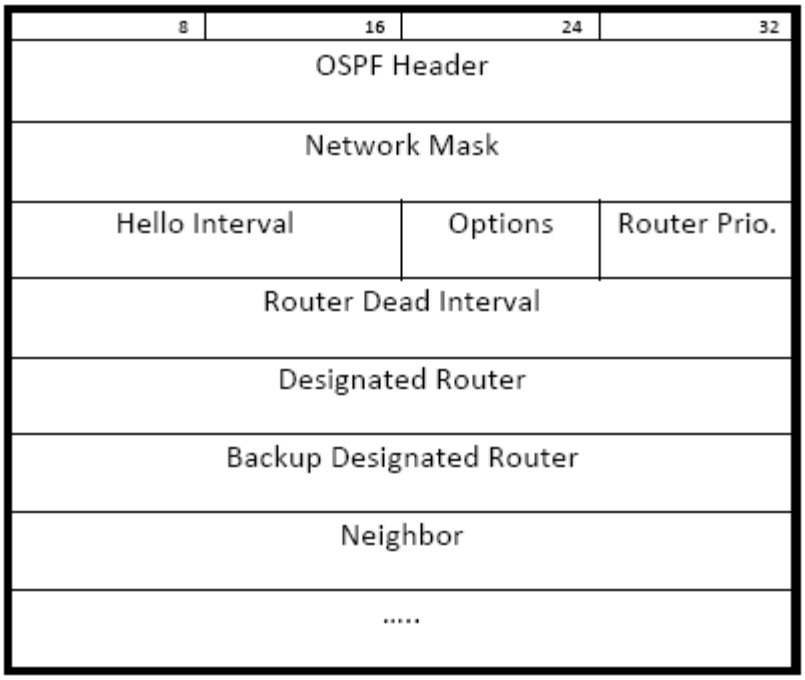


Rysunek 16 - format pola autoryzacja w nagłówku OSPF

- Key ID – definiuje lokalną wartość klucza znajdującego się na routerze zdalnym jak i lokalnym. Klucze muszą być jednakowe na obu routerach uczestniczących w komunikacji.
- Długość danych – definiuje długość danych autoryzacyjnych – hasha, który został dodany do pakietu.
- Liczba 32-bitowa w ciągu niemalejącym. Pozwala zapewnić ochronę przed Reply Attacks. Wiadomości z niższym numerem niż ostatnio odebrany zostaną odrzucone.

3.3.3.1 Pakiet Hello

Kiedy rozpoczyna się proces OSPF na interfejsie, pierwszym pakietem który jest przezeń wysyłany to pakiet Hello. Pakiet Hello ogłasza w sieci istnienie Routera OSPF oraz ustanawia relacje sąsiedztwa pomiędzy przylegającymi routerami. Pakiet Hello zawiera także informacje, które oba routery muszą wspólnie zaakceptować, aby móc wymieniać informacje o stanie łącz, takie jak Router Dead Interval, Hello Interval, DR, BDR. Poniższy Rysunek 17 - Budowa pakietu Hello



Rysunek 17 - Budowa pakietu Hello

Poniżej znajduje się krótki opis i znaczenie pól w pakiecie Hello.

- **Network Mask:**
  Używany w mediach rozgłoszeniowych. Reprezentuje maskę sieci, do której podłączony jest interfejs.

- **Hello Interval:**
  Jest to ilość sekund, określająca w jakich odstępach czasu router będzie wysyłał kolejne pakiety Hello.

- **Options:**
  Pole to pozwala określić jakie dodatkowe funkcjonalności obsługuje router. Oba routery muszą uzgodnić listę wspieranych funkcjonalności, aby móc się komunikować. Pole to zawiera 8 bitów. Każdy bit określa opcjonalną

funkcjonalność. Niektóre z bitów nie zostały przypisane do żadnej usługi, dlatego poniżej znajduje się opis 5 bitów już przypisanych do pewnych funkcjonalności. Poniżej na Rysunek 18 - kolejność i znaczenie bitów Options w nagłówku Helloznajduję się tabela przedstawiająca kolejność oraz znaczenie bitów.

-	-	DC	EA	N/P	MC	E	-

E	External Bit. It defines whether an external LSA will be flooded to the network.
MC	Multicast Bit. It defines whether multicast datagrams will be forwarded.
N/P	It refers to NSSA and how Type7 LSAs will be handled
EA	External Attribute bit. It shows whether the router will accept External-Attributes-LSAs
DC	Demand Circuits bit.

Rysunek 18 - kolejność i znaczenie bitów Options w nagłówku Hello

- **Rtr Priority:**

  Pole to określa priorytet routera w momencie wyboru DR i BDR. Router z najwyższym priorytetem jest wybierany jako DR, a drugi kolejny router z najwyższym priorytetem jako BDR. W przypadku gdy dwa lub więcej routerów posiada ten sam priorytet, wybierany jest router z wyższym Router ID. Router ID jest to najwyższe IP przypisane do jakiekolwiek interfejsu na routerze lub może być zdefiniowane ręcznie.

- **Router Dead Interval:**

Określa czas po którym router uzna swojego sąsiada za nieosiągalnego, jeżeli w przeciągu tego czasu nie otrzyma pakietu Hello. W routerach Cisco, domyślnie wartość ta ustawiana jest na czterokrotność Hello Interval.

- **Designated Router:**
  Określa adres IP routera DR. Uzywa 0.0.0.0 jeżeli nie wybrano DR.

- **Backup Designated Router:**
  Określa adres IP routera BDR. Używa 0.0.0.0 jeżeli nie wybrano BDR.

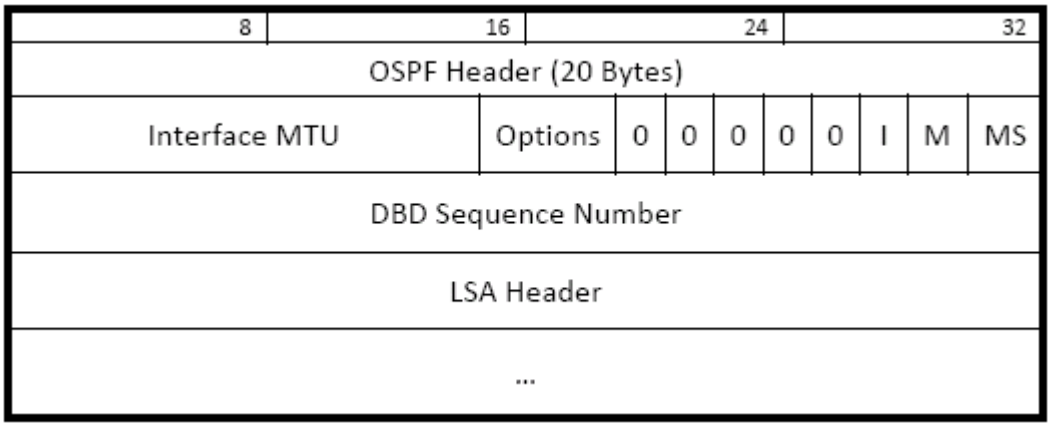
- **Neighbor:**
  Pole to określa listę Router ID, od których otrzymał pakiety Hello.

Kiedy zostały wymienione pakiety Hello pomiędzy routerami, router wie o swoich sąsiadach i o sąsiadach którzy są do nich podłączeni. W dodatku wybrany został DR i BDR oraz zostały uzgodnione Hello Interval oraz Dead Interval.

Aby móc użyć algorytmu Dijkstry do obliczenia najkrótszych ścieżek i zbudować tablicę routingu, router musi mieć obraz topologii całej sieci. Wykonywane jest to dzięki wymianie pakietów Database Description Packets(DBD). Pierwszy DBD jest używany do określenia relacji Master-Slave między dwoma routerami i do ustawienia początkowego numeru sekwencyjnego. Numer sekwencyjny jest zwiększany przez Master a potwierdzany przez Slave.

Jak już zostanie zakończona elekcja Master-Slave, rozpoczyna się synchronizacja. Na potrzeby tego, rozpoczyna się wymiana pakietów z nagłówkiem LSA.

3.3.3.2 Pakiet Database Description



Rysunek 19 - Database Description Packet

- **Interface MTU:**

  Wykazuje wielkość(Maximum Transfer Unit), jak dużo bajtów może zostać przesłanych w pakiecie przez ten intefejs.
- **Options**:

  To pole ma takie samą strukturę i znaczenie jak w przypadku pakietu Hello oraz rysunku Rysunek 18 - kolejność i znaczenie bitów Options w nagłówku Hello

- **Database Description bits:**
  - ***Bit I:*** Jest to bit inicjalizujący. Określa czy jest to pierwszy wysłany pakiet DBD. Kiedy został użyty, należy dodać 4 do bitu M.
  - ***Bit M:*** Jest używany do określenia ilości pakietów(jeden lub więcej) które jeszcze nadejdą.
  - ***Bit MS:*** Jest to bit Master-Slave. Jeżeli jest ustawiony na 1, wtedy to jest Master, w przeciwnym razie (0), oznacza ze router jest Slave. Każdy router porównuje swój priorytet i Router ID z innymi routerami, aby zdecydować kto powinien być Master, a kto Slave.

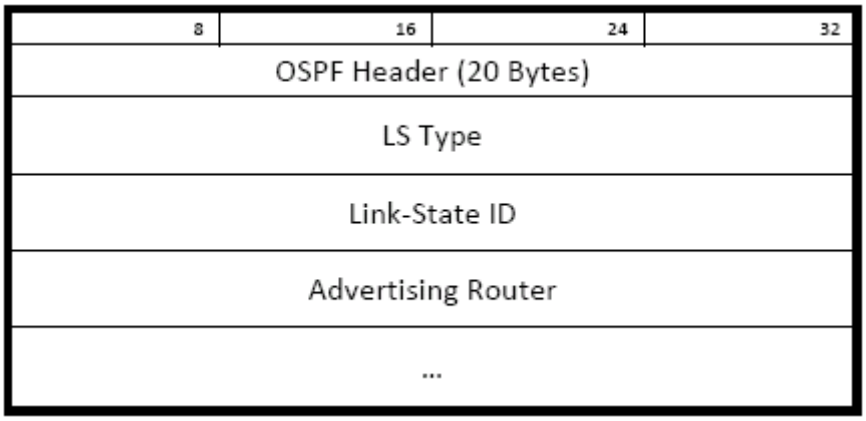
- **DBD Sequence Number:**

  Jest to wartość numeru sekwencyjnego uaktualnienia. Wartość ustawiana jest przez Master i zwiększana tylko przez niego.
- **LSA Header:**

Może zostać to opisane jako tytuł LSA. Informacje które przenosi LSA Header identyfikują stan łącza, który jest przechowywany w bazie. Nagłówek LSA będzie opisany dokładnie w kolejnych podrozdziałach.

3.3.3.3 Pakiet Link-State Request

Protokół OSPF daje routerowi możliwość zapytania się o stany konkretnych łącz, których mu brakuje lub informacje które posiada są przedawnione. Dokonuje tego poprzez pakiety Link-State Reuqest. Jest to ostatni krok w ustanawianiu sąsiedztwa. Jak już zostało wspomniane, aby unikalnie zidentyfikować dane łącze(Link-State) w bazie, musimy podać typ LS, Link-State ID i Advertising Router. Z tego możemy zgadywać, że pakiet wygląda jak na Rysunek 20 - Struktura pakietu LSR



Rysunek 20 - Struktura pakietu LSR

3.3.3.4 Pakiet Link-State Update

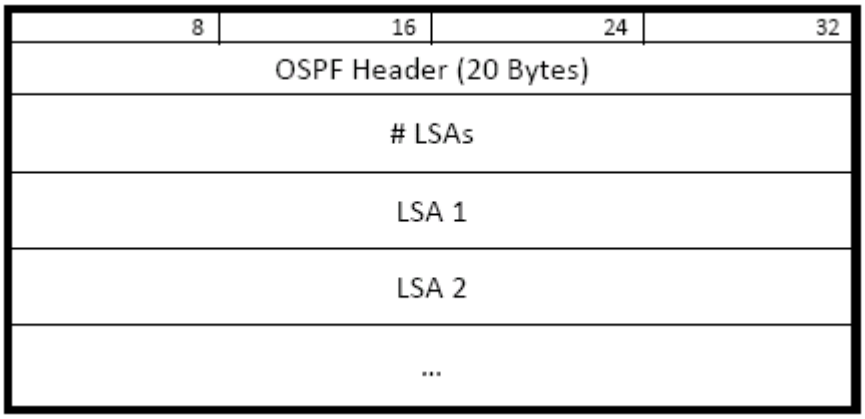
Pakiety te są używane do przenoszenia pakietów LSA. Każdy pakiet zawiera nagłówek OSPF.W celu optymalizacji i zmniejszenia zużycia pasma informacjami o nagłówku, można wysłać pakiet zbiorczy. Pakiet LSU może przenosić jeden lub więcej pakietów LSA w jednym pakiecie. LSU są wysyłane na adres multicast.

Tutaj mamy zaimplementowany Flooding Mechanism. Jest to zrobione w celu zapewnienia ochrony przed spoofingiem. Wszystkie routery wymieniają się pakietami mówiącymi, od kogo dostały wiadomość i treść wiadomości. Pozwala to na weryfikację nadawcy. Wszyscy odbiorcy sprawdzą czy ich nazwa jest w pakiecie i ustalą co w nim się

znajduje. Jeżeli wszystko się zgadza, to sprawdzą pakiet. W przeciwnym wypadku uruchomiony zostanie mechanizm obronny.

Pakiet LSU potwierdzany jest pakietem LSAck.

Istotnym tutaj szczegółem ze względów bezpieczeństwa jest to, że w przypadku retransmisji pakiet LSU wysyłany jest jako unicast.

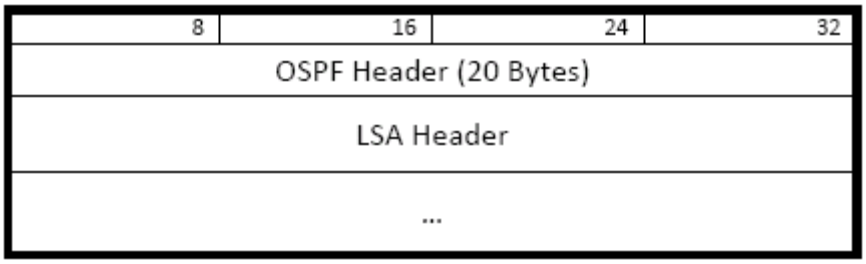


Rysunek 21 - Struktura pakietu LSU

Jak widać na Rysunek 21 - Struktura pakietu LSU, nagłówek OSPF jest na początku, następnie ilość zawartych pakietów LSA. Kolejnymi są pakiety LSA - nagłówek LSA i dane LSA.

3.3.3.5 Pakiet Link-State Acknowledgment

Pakiet ten używany jest to potwierdzenia odbioru pakietu LSU. Pakiet ten może zostać wysłany jako multicast do wszystkich routerów biorących udział w procesie OSPF lub jako multicast do wszystkich routerów DR lub jako unicast. Pakiet ten ma bardzo prostą strukturę. Składa się z nagłówka OSPF i nagłówka LSA pakietu który potwierdza. Pakiet ten może zawierać jeden lub więcej nagłówków LSA, co oznacza, że może potwierdzać odbiór większej ilości pakietów LSA.



Rysunek 22 - Struktura pakietu LSAck

3.3.4 Pakiet LSA

Aby zrozumieć w jaki sposób pakiety LSA są używane to wymiany informacji routingu oraz jakie istnieją podatności, musimy znać strukturę tych pakietów oraz informacje jakie one przenoszą.

Struktura pakietu pasuje do ogólnego szablonu pakietów OSPF. Pakiet OSPF ze swym nagłówkiem przenosi pakiet LSA.

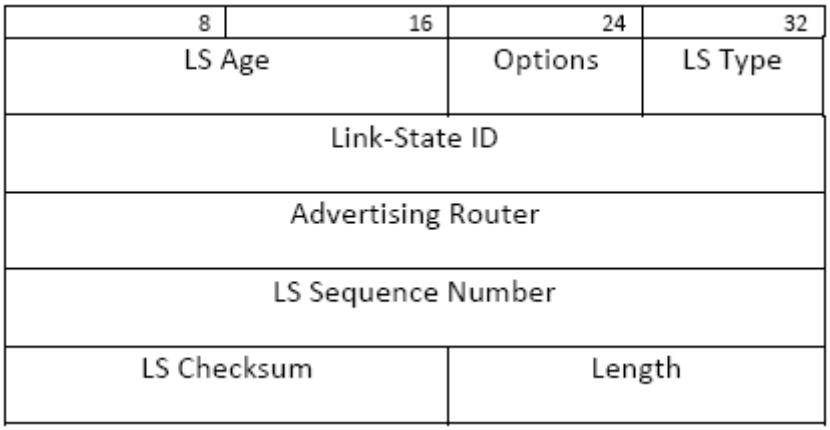
LSA ma 6 różnych typów:

1. Router LSA
2. Network LSA
3. Summary Network
4. Summary ASBR
5. External
6. NSSA

3.3.4.1 Nagłówek LSA

Jak możemy zauważyć na Rysunek 23 - Nagłówek LSA, nagłówek przenosi informacje pozwalające określić stan łącza oraz niektóre informacje dotyczące jego czasu. Zawiera także pole Options, Length i Checksum.

Poniżej struktura nagłówka pakietu LSA oraz krótki opis każdego pola.



Rysunek 23 - Nagłówek LSA

- **LS Age:**
  Określa czas życia informacji zawartej w LSA. Maksymalna wartość wynosi 3600 sekund, jest to jedna godzina przez którą LSA może być w bazie routera, jeżeli nie zostało uaktualniane. Czas odświeżania tej informacji wynosi połowę maksimum, czyli 1800 sekund.

- **Options:**
  To pole ma takie samą strukturę i znaczenie jak w przypadku pakietu Hello oraz rysunku Rysunek 18 - kolejność i znaczenie bitów Options w nagłówku Hello

- **LS Type:**
  Jak już wspomniano wcześniej, zawiera typ pakietu LSA. Dokładniejszy opis każdego z nich zostanie podany w kolejnych sekcjach.

- **Link-State ID:**
  Jest zależny od pola LS Type, zostanie opisany w kolejnych sekcjach.

- **Advertising Router:**
  Zawiera Router ID routera, który wygenerował pakiet.

- **LS Sequence Number:**
  Pole to jest bardzo istotne z perspektywy bezpieczeństwa. Jest ono używane do uniknięcia ataków powtórzeniowych lub bardziej ogólnie, do odrzucenia starych

pakietów, które zostały odebrane. Wartość początkowa tego pola wynosi 0x80000001 i może osiągnąć wartość maksymalnie 0x7FFFFFFF. Istnieją ataki ukierunkowane na to pole i zostaną opisane w późniejszych rozdziałach.

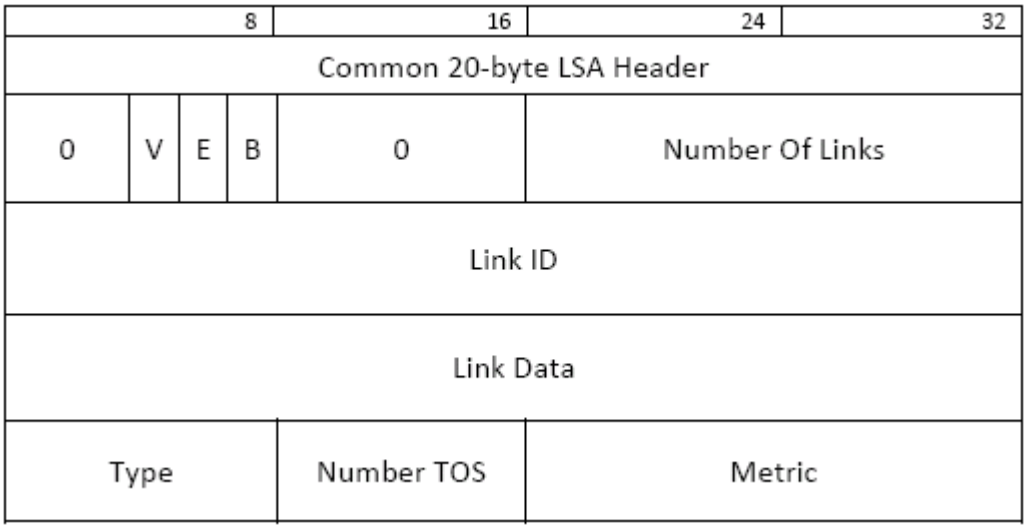
- **Checksum:**
  Suma kontrolna jest liczona przy pomocy algorytmu Fletcher 8bit. Nie bierze on pod uwagę pola LS Age. Bierze on pod uwagę cały pakiet LSA, nie tylko nagłówek który jest dołączany.

- **Length:**
  Określa długość całego pakietu – danych wraz z nagłówkiem.

  LSA jest identyfikowane na podstawie trzech pól:
  - LS Type
  - Link-State ID
  - Advertising Router

3.3.4.2 Router LSA

Każdy router wysyła Router LSA, który opisuje stan jego interfejsów.



Rysunek 24 - Router LSA

- **Bity V,E,B:**
  Flagi te używane są do oznaczenia typu Routera. V jest używane dla punktów podłączonych przez Virtual Links, E do zadeklarowania się jako Autonomous

System Boundary Router (ASBR), a B do deklaracji jako Area Border Router (ABR).

- **Number of Links:**
  Liczba ta określa ilość połączeń Routera, które są opisane w pakiecie Router LSA.

- **Link ID, Link Data, Type:**
  Te trzy wartości są od siebie zależne nawzajem. Mamy 4 typy Router Links.
  Poniższy Rysunek 25 - Router LSA - opis znaczenia poszczególnych pól, przedstawia zestawienie i opis możliwych kombinacji wartości poszczególnych pól.

Type	Description	Link ID	Link Data
1	Point-to-point numbered	Neighbor's router ID	Interface IP address
1	Point-to-point unnumbered	Neighbor's router ID	MIBII IfIndex value
2	Transit	IP address of the DR	Interface IP address
3	Stub	IP network number	Subnet mask
4	Virtual link	Neighbor's router ID	Interface IP address

Rysunek 25 - Router LSA - opis znaczenia poszczególnych pól

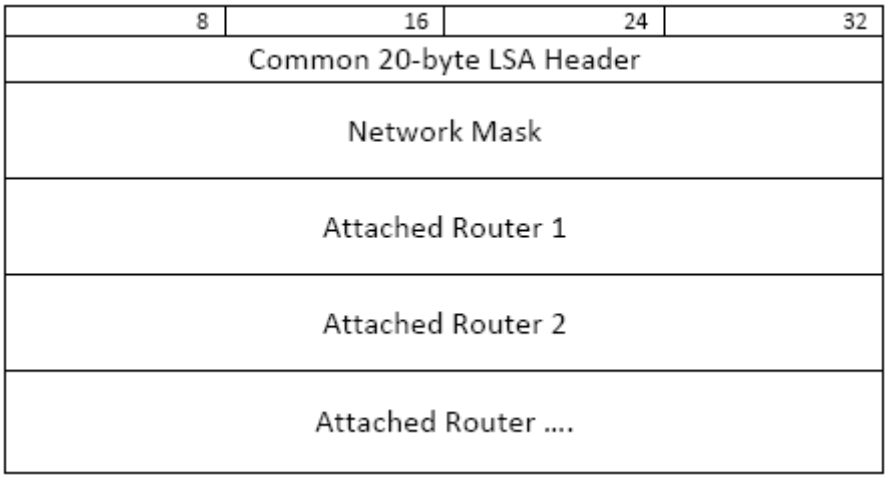
- **Number ToS:**
  Jest to pole Typ Usługi, zwykle ustawione na 0.

- **Metric:**
  Pole to określa koszt danego połączenia. Koszt w protokole OSPF liczony jest poprzez podzielenie 10^8 przez pasmo łącza. (100000000/pasmo)
  Dla przykładu: jeżeli mamy interfejs 100Mbps, wtedy to będzie 100x10^6=10^8, co daje metrykę o wartości 1.

3.3.4.3 Network LSA

Ten typ LSA jest generowany przez DR. Zadaniem tego pakietu jest opisanie wszystkich routerów, które są podłączone do sieci.

W niektórych typach sieci DR nie jest wybierany. W tym przypadku ten typ LSA nie jest generowany. Network LSA ma bardzo prostą strukturę. Składa się z nagłówka LSA, maski sieci i identyfikatorów routerów(Router ID), które są podłączone do tej sieci. Struktura pakietu przedstawiona jest na poniższym rysunku(Rysunek 26 - Network LSA).



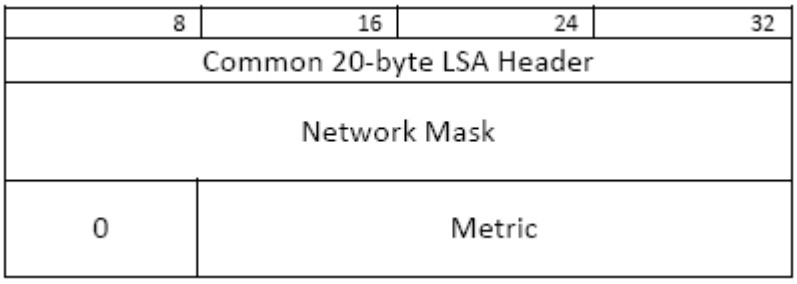
Rysunek 26 - Network LSA

3.3.4.4 Summary LSA

Pakiety Summary LSA generowane są przez ABR. Pakiet ten jest przeznaczony do opisu sieci która należy do tego samego Autonomous System, ale jest w innym obszarze. Informacje takie są wymieniane, aby można było utrzymać łączność pomiędzy obszarami, przy jednoczesnej minimalizacji przesyłanych informacji, co pozwala zaoszczędzić pasmo.

Warunkiem wygenerowania tego pakietu, jest to, że obszar 0 musi być skonfigurowany oraz przynajmniej jeszcze jeden obszar podłączony do obszaru 0.

Struktura pakietu jest bardzo prosta, gdyż główną informacją, którą przenosi pakiet jest maska sieci. Reszta pól została już objaśniona i z samej nazwy możemy określić ich funkcjonalność.



Rysunek 27 - Summary LSA

Należy nadmienić, że dane, które pakiet przenosi, różnią się, w zależności od tego czy pakiet pochodzi z obszaru 0 (Backbone area) czy z zewnątrz.

Kiedy pakiet pochodzi z wewnątrz obszaru głównego(0), wtedy może przenosić informacje o:

- Przyłączonych drogach
- Drogach dla obszarów bezpośrednio przyłączonych do routera
- Drogach dla obszarów nie bezpośrednio przyłączonych do routera

Natomiast kiedy pakiet pochodzi z innego obszaru, może przenosić informacje o:

- Przyłączonych drogach
- Drogach dla obszarów bezpośrednio przyłączonych do routera

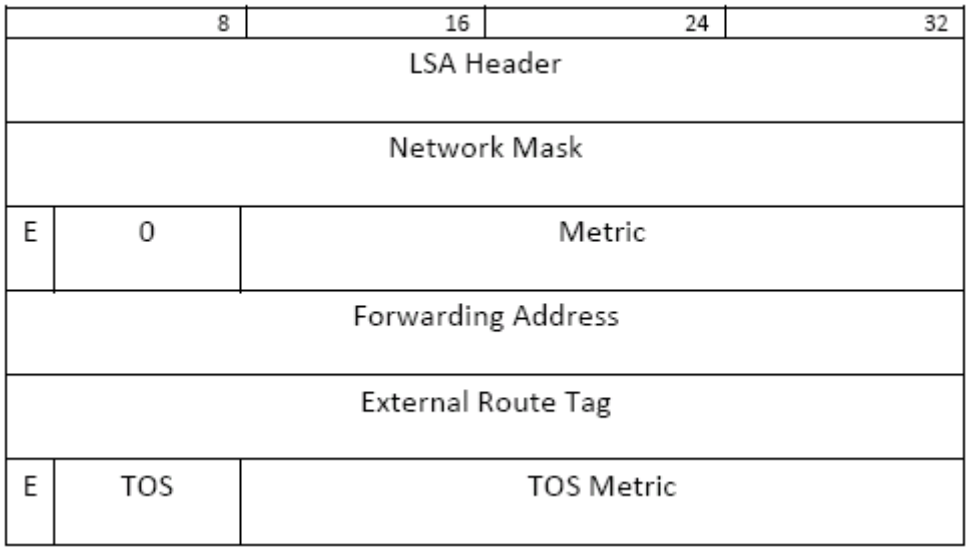
Jak już zostało wspomniane, pakiet ten generowany jest przez ABR, więc pole Link-State ID z nagłówka LSA brane jest jako wartość określająca adres sieci.

3.3.4.5 Summary ASBR LSA

Pakiet Summary ASBR LSA ma ten sam format jak Summary LSA. Różnicą jest natomiast to, że pakiet generowany jest przez ASBR i to, że maska sieci ma wartość 0 (0.0.0.0). Dodatkowo, Link-State ID w nagłówku LSA ma wartość ASBR.

3.3.4.6 External LSA

External LSA opisuje trasy do miejsc docelowych na zewnątrz Autonomous System. Większość External-LSA opisuje trasy to określonybch zewnętrznych miejsc docelowych; w tym wypadku Link-State ID w pakiecie LSA, ustawiony jest na adres IP sieci docelowej(jeżeli potrzeba, Link-State ID może posiadać więcej ustawionych bitów hosta dla danej sieci; szczegóły opisane są w załączniku E). Trasa domyślna dla Autonomous System może być opisana w AS-external-LSA poprzez ustawienie Link-State ID na DefaultDestination (0.0.0.0). AS-external-LSA są nadawane przez ASBR. ASBR generuje pojedynczy pakiet AS-external-LSA dla każdej zewnętrznej trasy, której się nauczył poprzez inny protokół routingu(np. BGP) lub poprzez ręczną konfiguracje.[12]



Rysunek 28 - External LSA

Poniżej opis pól w pakiecie, które nie pojawiły się we wcześniejszych opisach:

- **Network Mask:**
  Informuje o masce sieci, której dotyczy dany pakiet.

- **Bit E:**
  Bit ten używany jest do zdefiniowania typu metryki. Jeżeli ustawiony jest na 0, wtedy metryka jest typu 1 i koszt łącza zmieniany jest po każdym przeskoku przez węzeł sieci. Jeżeli bit E jest ustawiony na 1, wtedy mamy do czynienia z metryką typu 2, która to nie zmienia się przez całą trasę, którą pokonuje pakiet.

- **Forwarding Address:**
  Adres określa, gdzie ruch powinien być kierowany. Pole to może przyjmować wartość 0.0.0.0, jeżeli pakiety mają być kierowane do ASBR.

3.3.5 Działanie protokołu OSPF

Poniżej przykład działania OSPF w momencie inicjalizacji relacji przylegania.

Router 1	Packets Sent	Router 2
Down	Hello [DR=0, Seen=0) →	Down
	← Hello [DR=R2,Seen=R1]	Init
ExStart	DD [Seq=x, I, M, Master] →	
	← DD [Seq=y, I, M, Master]	ExStart
Exchange	DD [Seq=y, M, Slave] →	
	← DD [Seq=y+1, Master]	Exchange
	DD [Seq=y+1, Slave] →	
	← DD [Seq=y+n, Master]	
Loading	DD [Seq=y+n, Slave] →	
	← LS Request	Full
	LS Update →	
	← LS Request	
Full	LS Update →	

4 Systematyka i opis ataków na protokoły routingu

W poprzednich rozdziałach dogłębnie przeanalizowane zostały protokoły routingu, ich działanie, format ramek, jak routery współpracują przy ich pomocy.

Protokoły te zostały zaprojektowane w taki sposób, że zapewniają bezpieczeństwo i efektywność.

Realizacja tych dwóch czynników zależy w głównej mierze od poprawnej konfiguracji routerów w sieci. Poprawa efektywności działania sieci może stać się przyczyną gorszego

bezpieczeństwa, a z drugiej strony poprawa bezpieczeństwa może oznaczać zmniejszenie efektywności działania sieci.

Luki które może wykorzystać atakujący mogą powstać również na skutek błędów oprogramowania, które jest wykonywane na routerze. Niestety na te luki, administrator nie ma wpływu, zwłaszcza, gdy nie są one jeszcze powszechnie znane.

4.1 Systematyka ataków

Ataki na protokoły routingu można grupować w różny sposób. Poniżej przedstawione zostały podziały biorące pod uwagę różne kryteria, takie jak cel, zastosowanie, spektrum działania oraz spektrum oddziaływania. Podziały te autor wprowadził we własnym zakresie i maja one na celu usystematyzowanie wiedzy przekazanej w tej pracy jak i zarazem ukazują ogrom i różnorodność ataków z jakimi można się spotkać w środowisku sieciowym.

Podział ataków ze względu na błędy powstałe w cyklu produkcyjnym routera:

- **Błąd koncepcji**
  Przypadek ten może zaistnieć w momencie, kiedy przy złej analizie warunków w jakich przyjdzie pracować temu protokołowi, nie wzięliśmy wszystkich zmiennych pod uwagę.

- **Błąd bycia ufnym**
  Zakładamy, że w sieci są tylko dobrzy gracze, nikt nie oszukuje i nie atakuje.

- **Błąd implementacji**
  Błąd powstały w skutek błędu programisty oprogramowania operującego na routerze.

Ataki można podzielić także, ze względu na miejsce w którym styka się działająca sieć i atakujący:

- **Atak z wykorzystaniem routera obcego**

Do atakowanej sieci wprowadzamy własne urządzenie, które ma symulować działanie routera. Atak taki jest ukierunkowany na długotrwałe, niewykrywalne działanie.

- **Atak z wykorzystaniem routera przejętego**
  Atakujący uzyskuje nieautoryzowany dostęp do routera zaufanego i istniejącego w sieci. Może w ten sposób zmieniając pewne ustawienia wpływać na przepływ pakietów.

- **Atak z wykorzystaniem podszywania się**
  Ten rodzaj ataku jest dość prymitywny i na ogół wiąże się z szybkim wykryciem atakującego oraz brakiem możliwość przeprowadzenia wszystkich rodzajów ataków, a co za tym idzie, umożliwia w większości jedynie unieruchomienie sieci, a nie zmianę przepływu danych.

Atak wreszcie, można podzielić uwzględniając szkodliwość:

- **DoS**
  Atak ukierunkowany na całkowite unieruchomienie sieci.

- **Privilage Escalation**
  Dzięki temu atakowi zyskujemy większe uprawnienia w sieci czy systemie. Możemy wydostawać się poza granice narzucane nam przez politykę bezpieczeństwa, co może powodować, że będziemy mieli nieuprawniony dostęp do większej ilości informacji.

- **Performance**
  Atak ten drastycznie i zauważalnie zmniejsza wydajność sieci.

- **Sniffing**
  Jest to jeden z najniebezpieczniejszych ataków, gdyż ukierunkowany jest na długotrwałe działanie i przechwytywanie danych. Wiąże się to z tym, iż atakujący będzie się starał jak najmniej ingerować w działanie sieci, aby przedłużyć swoją egzystencje w systemie i móc przechwycić największą ilość danych.

- **Spoofing**

  Podszywanie się pod inne węzły w sieci jest dość łatwo wykrywalne i może prowadzić w głównej mierze do zakłóceń działania sieci.

Kolejnym podziałem jest pochodzenie ataku:

- **Wewnętrzny**
- **Zewnętrzny**

Ostatecznie można ataki podzielić ze względu na spektrum działania:

- **Jednoobszarowe**
- **Obejmujące System Autonomiczny**
- **Obejmujące globalną sieć**

4.2 Opis ataków na protokół RIP

Protokół RIP ze względu na swoją prostotę nie wymaga zbyt dużego wywiadu, zanim zostanie zaatakowany. Posiada on dość poważne podatności na atak, zwłaszcza RIPv1, która nie powinna być w ogóle używana. RIPv2 posiada pewne zabezpieczenia, jednak przy dostępnych teraz mocach obliczeniowych komputerów dostępnych w sklepach oraz dzięki dostępności tęczowych tablic, złamanie nawet autoryzacji zaszyfrowanej MD5 nie stanowi większej przeszkody.

Prostota czyni ten protokół łatwym w konfiguracji i utrzymaniu, jednocześnie powodując pewne jego ograniczenia i łatwość manipulacji nim.

4.2.1 Malicious Route Insertion

Protokół ten mimo swoich ograniczeń opisanych w rozdziale 3.2, jest nadal szeroko stosowany w małych sieciach. Znając podatności RIP, można manipulować pakiety w ten sposób, aby routery uznawały inne ścieżki za najkrótsze, aniżeli wynikałoby to z ich analizy.

Dzięki temu możemy przekserować ruch w sieci w ten sposób, że będzie on przechodził przez urządzenie pod naszą kontrolą, dzięki czemu będziemy w stanie przechwytywać ruch krążący w sieci. Kolejnym problemem, może być stworzenie takich wpisów tras w routerach,

że będą one wysyłały pakiety do nieistniejących węzłów, co sprawi, że komunikacja w sieci zostanie uniemożliwiona.

4.2.2 Downgrading Attack

Protokół RIPv1 nie wspiera żadnego mechanizmu bezpieczeństwa autoryzacji pakietów, jedyna możliwość to listy dystrybucyjne i listy dostępu. Niestety obie te listy mogą zostać łatwo ominięte, gdy użyjemy IP Spoofing.

Jeżeli więc atakujący wstawi do atakowanej sieci router z RIPv1 lub wyśle pakiety spreparowane jako RIPv1, może zmusić routery do przejścia na niższą wersje protokołu, co w konsekwencji prowadzi do wyłączenia wszelkich mechanizmów bezpieczeństwa wprowadzonych w RIPv2.

Szczęśliwie dla właściciela sieci, system operacyjny routerów Cisco – IOS nie daje się zmusić do takiego zachowania. Istnieje za to cała gama tańszych rozwiązań mniejszych dostawców, która jest podatna na ten atak. Komponuje się to idealnie, gdy weźmiemy pod uwagę to, że mniejsze firmy wybiorą tańsze urządzenia i budują mniejsze sieci oparte na RIP, co stwarza idealną mieszankę dla atakującego.

4.2.3 MD5 Hash Cracking Attack

Jeżeli atakujący trafi na to najlepsze dla RIPv2 zabezpieczenie, to dalej jest w stanie wykorzystać całą gamę narzędzi dostępnych do złamania hashu MD5. Aktualnie znane są algorytmy pozwalające skrócić łamanie o rzędy wielkości i zrobić to w rozsądnym czasie. Drugim sposobem jest wykorzystanie tęczowych tablic, które dają niemal natychmiast rozwiązanie, jedynym warunkiem jest to, że ktoś już wcześniej złamał i opublikował dany hash.

Biorąc pod uwagę pragmatyzm administratorów sieci, którzy niechętnie zmieniają hasła w urządzeniach sieciowych i raczej stosują stałe hasła, możemy być pewni sukcesu.

4.3 Opis ataków na protokół OSPF

Protokół ten ze względu na poziom skomplikowania i szeroką gamę funkcjonalności jakie ma w sobie zawarte, stwarza duże pole do popisu atakującym, a jednocześnie jest dużym

wezwaniem dla osób odpowiedzialnych za jego bezpieczeństwo w sieciach w których został wdrożony.

Jednak z drugiej strony skomplikowanie protokołu czyni go bardziej odpornym na ataki, ze względu na potrzebę tworzenia relacji sąsiedztwa oraz hierarchizacje budowy sieci.

4.3.1 Przejmowanie urządzenia

Jednym z największych zagrożeń a zarazem najcięższych do wykrycia, jest atak przejęcia urządzenia. W momencie udanego ataku, sprawca jest w miejscu, z którego może uczynić znaczne straty dla całej sieci. Dzięki posiadaniu władzy nad istniejącym urządzeniem, można usuwać istniejące trasy lub dodawać nowe, tak aby pakiety były przekazywane w miejsce odpowiednie dla atakującego. W momencie wykrycia ataku, wszystkie dane przechodzące przez przejęty węzeł powinny być traktowane jako niezaufane i prawdopodobnie przechwycone. Dodatkowym zagrożeniem przy protokołach stanu łącza jest możliwość manipulacji trasowaniem w innych routerach niż tylko ten przejęty, co stwarza dodatkowe ryzyko wycieku lub utraty danych.

Scenariusz przejęcia istniejącego urządzenia powinien być rozpatrywany jako sprawa najwyższego ryzyka, dlatego więc routery powinny być dobrze chronione tym silniej, im bardziej w centrum hierarchii sieci występują, aby zminimalizować ryzyko takich ataków. Możliwe jest to poprzez zastosowanie haseł, certyfikatów czy innych sposobów centralnego uwierzytelniania takich jak TACACS+ czy RADIUS, obsługiwanych przez dane routery.

Ze względu na to, że atak na urządzenie może być przeprowadzony w różny sposób nie obejmujący ataku na sam protokół trasowania, temat ten jest poza zakresem zainteresowań tego dokumentu.

4.3.2 Naruszanie połączenia

Bardziej interesującym scenariuszem w temacie tego opracowania jest kompromitowanie logicznego połączenia. Naruszenie połączenia jest łatwiejsze niż atak na urządzenie, gdyż każdy kto ma dostęp do sieci ma możliwość dokonania tego. Najważniejszym warunkiem jest posiadanie dostępu do połączenia, którym wymieniane są komunikaty protokołu trasowania.
Pierwszym krokiem jest podsłuchanie łącza na występowanie odpowiednich pakietów. Dzięki temu atakujący może zebrać garść niezbędnych do przeprowadzenia udanego ataku informacji na temat topologii sieci jak i konfiguracji samego protokołu. W skład tego

wchodzą między innymi to jaki rodzaj autoryzacji jest używany, jej konfiguracja, kto jest DR i BDR czy w jakim obszarze się znajdują urządzenia. Gromadzenie wszystkich tych informacji na ogół nie wymaga ujawniania się w sieci, jest więc to dość użyteczna i pasywna metoda ograniczająca wykrycie na etapie zbierania informacji.

Po zebraniu niezbędnych informacji, można przystąpić do aktywnego ataku. Mając dostęp do połączenia, można wygenerować i wstrzyknąć odpowiednie pakiety. Dzięki znajomości protokołu trasowania można przeprowadzić skuteczny atak na cały mechanizm trasowania w sieci i wprowadzić w błąd węzły biorące udział w procesie wyznaczania tras.

Kolejnym sposobem ataku połączenia jest powtórne wysłanie tych samych pakietów lub usuwanie pakietów, w ten sposób że nie dotrą do odbiorcy. Ataki te muszą być przeprowadzane w sposób dostosowany do konkretnego protokołu trasowania i sposobu jego obrony przed atakami. W przypadku stosowania numerów sekwencyjnych atak powtórzeniowy jest jednym ze sposobów radzenia sobie z tym zabezpieczeniem. Z kolei dla przykładu, przy pomocy usuwania pakietów Hello możemy skutecznie przeprowadzać atak, który spowoduje desynchronizacje węzłów i ich rozłączenie, gdyż po upływie czasu DeadInterval uznają, że ich sąsiedzi przestali istnieć.

Następnym atakiem na połączenie jest modyfikacja wiadomości, jednak atakujący musi być w stanie przeliczyć sumę kontrolną pakietu i ewentualny hash w przypadku użycia zabezpieczeń kryptograficznych. Co ważne, modyfikacja może występować w polach używanych ściśle przez protokół lub co bardziej niebezpieczne, w polach protokołów niższych warstw, a te z kolei nie są chronione w żaden sposób poza sumą kontrolną.

Atak więc podzielić można na kilka faz. Pierwszą fazą jest przechwycenie wiadomości, po czym stworzenie innej lub zmodyfikowanie istniejącej a następnie wstrzyknięcie jej do sieci. Dzięki temu atakujący może z powodzeniem zmienić postrzeganie topologii sieci przez węzły biorące udział w procesie trasowania.

Zmiana postrzegania sieci może przynieść dwa rezultaty. Pierwszy jest to sytuacja, gdzie inny węzeł będzie widział fałszywy węzeł i nawiąże z nim relacje sąsiedztwa. Wtedy fałszywy węzeł zostanie wpisany do tablicy sąsiedztwa danego oszukiwanego węzła, przez co atakujący może przystąpić do wymiany pakietów Hello jak i informacji o trasach. Dzięki temu może manipulować trasami jak i zdobywać więcej informacji o samej sieci, jej topologii i konfiguracji protokołu. Kolejną sytuacją, jaka może nastąpić na skutek zmiany postrzegania sieci jest sytuacja, w której na skutek manipulacji przy informacjach wymienianych między węzłami wprowadzimy błędne czy nieistniejące trasy. W tej sytuacji powinien zadziałać

mechanizm obronny OSPF próbując poprawić nieprawdziwe informacje, jednak jak opisano w poprzednich rozdziałach, atakujący może wykorzystać odpowiednie luki w protokole aby utrzymać swoje informacje w węzłach lub może uniknąć wykrycia oszustwa poprzez wysyłanie pakietów unicast skierowanych bezpośrednio do swoich ofiar.

Jeżeli atakujący chce wstrzyknąć pakiety w sieci, musi rozpoznać w jakim typie obszaru się znajduje, gdyż użycie nieodpowiedniego typu LSA spowoduje automatyczne odrzucenie pakietu i brak powodzenia samego ataku. Jak wspomniano w 3.3.4 istnieją różne rodzaje pakietu LSA używane w odpowiednich typach obszarów. Typ 1 i 2 mogą być wstrzyknięte w każdym obszarze, typ 3 tylko w obszarze 0, zwykłym i stubby, typ 5 tylko w głównym i normalnym, a typ 7 tylko w NSSA.

Poprzez wstrzyknięcie niektórych pakietów, atakujący może zmienić postrzeganie topologii w taki sposób, że węzły odniosą wrażenie przeniesienia sieci totally stub w miejsce gdzie atakowane połączenie się znajduje. Innym sposobem jest twierdzenie, że węzeł w atakowanym połączeniu jest ABR lub ASBR, co zwiększy ruch na danym połączeniu i może uniemożliwić komunikacje. Nie tylko dodanie nowych tras jest możliwe podczas ataku na połączenie, ale także dodanie nowych obszarów czy zmiana sposobu przepływu ruchu jest możliwa dzięki atakowi na połączenie.

Jak wynika z przedstawionych informacji, jest wiele sposobów i scenariuszy przeprowadzenia ataku na sieć. Jest jasnym, że decyzja co do sposobu implementacji ataku zależy od celu jaki atakujący chce osiągnąć, wybierając pomiędzy atakiem DoS a przechwyceniem informacji. W tym wypadku kluczem do przeprowadzenia ataku jest zbieranie informacji, tak aby można było uwzględnić wszystkie czynniki wpływające na powodzenie ataku i wybrać odpowiednią metodologię. Jest wiele pytań na które atakujący musi odpowiedzieć zanim rozpocznie atak. Jednymi z tych pytań są: Czy informacja którą chcę przechwycić przechodzi przez skompromitowane połączenie? Jeżeli nie, to jak zmusić węzły to przekazywania tej informacji odpowiednią dla atakującego drogą? W jakiego rodzaju obszarze się znajduję? Odpowiedzi na te i inne pytania należy znać zanim zostanie rozpoczęte wstrzykiwanie odpowiednich pakietów.

4.3.3 Podatności protokołu OSPF

W poprzednich rozdziałach pokazane zostało jak wyglądają ramki protokołu OSPF oraz jak wymienia komunikaty w celu poprawnej wymiany informacji o trasach w Automonous System.

Sposób w jaki pakiety zostały zaprojektowane miał zapewnić bezpieczeństwo oraz wydajność. Problemem do pogodzenia w środowiskach produkcyjnych stał się wybór między wydajnością sieci a bezpieczeństwem i często administrator sieci musi dobierać wagę tych dwóch czynników indywidualnie bazując na analizie ryzyka i potrzeb danej sieci. Dochodzą do tego błędy w implementacji protokołu na urządzeniach, co nie zawsze jest do przewidzenia. Tak więc ustawienie zbyt dobrej wydajności może stwarzać luki w bezpieczeństwie jak i zbytnie postawienie na bezpieczeństwo może drastycznie zmniejszyć wydajność sieci. Za każdym razem administrator musi dokonać wyważonego wyboru.

Poniżej zostały przedstawione ataki których może spodziewać się administrator sieci oraz ich potencjalna metoda wykorzystania.

4.3.3.1 Breaking Adjacency

Jak zostało opisane w rozdziale 3.3.3.1 na stronie 27 protokół OSPF w celu utrzymania relacji sąsiedztwa wymienia pakiety Hello. Jeżeli jeden z dwóch routerów przestanie wysyłać pakiety hello lub czas między dwoma kolejnymi pakietami przekracza DeadInterval, wtedy relacja sąsiedztwa ulegnie przerwaniu.

Przerwanie relacji sąsiedztwa powoduje serie czynności które muszą wykonać routery, aby przywrócić ponownie stabilizacje w sieci. Po pierwsze router musi uaktualnić swój Router LSA. Uaktualnienie tego spowoduje wykonanie kalkulacji tras przy pomocy SPF. To może spowodować uaktualnienia w tablicy routingu. W dodatku, jeżeli router który przerwał nadawanie pakietu Hello to DR, wtedy musi zostać uaktualniony pakiet Network LSA.

Jedną z metod ataku przerwania relacji sąsiedztwa jest usuwanie, zmiana pakietów Hello lub po prostu opóźnianie ich dostarczenia na dłużej niż wynosi DeadInterval.

Kolejną metodą ataku może być użycie podatności niższych warstw, takie jak IP Spoofing. Nawet w momencie użycia metod kryptograficznych jako zabezpieczeń, do wyliczania hasha nie brane są pod uwagę dane z nagłówka IP. W rezultacie możemy dowolnie zmodyfikować nagłówek, gdyż odbiorca nie ma narzędzi do zweryfikowania nadawcy.

Mając na uwadze opis 3.3.3.1 na stronie 27 wiemy, że pakiet Hello na ostatnim polu(przedostatnim jeżeli użyte jest szyfrowane hasło) przenosi jego aktywnych sąsiadów.

Załóżmy że pobieramy takie pakiet pochodzący od Routera A do B. Pakiet zostaje zmodyfikowany poprzez zamianę Źródłowego IP z docelowym IP i odesłany powrotem do routera A. W tym momencie Router A nie może znaleźć siebie na liście sąsiadów. Wnioskiem z tego jest to, że połączenie z routerem B nie jest dwustronne i w tym momencie relacja sąsiedztwa jest przerywana.
Co ważne, atak ten może się powieść nawet, gdy włączone są pełne zabezpieczenia OSPF.
W niektórych implementacjach OSPF, sprawdzane jest także pole Router ID do potwierdzenia gdzie został wygenerowany pakiet i dopiero ta czynność może zapobiec temu atakowi, pod warunkiem że zostało włączone zabezpieczenie z szyfrowanym hasłem, gdyż zamiana pola, które podlega ochronie szyfrowania staje się nieużyteczne i atak może skończyć się niepowodzeniem lub nawet wykryciem.

4.3.4 Ataki Denial of Service

Ataki te związane są z unieruchomieniem sieci lub spowolnieniem jej działania. Jest kilka sposobów aby tego dokonać. Może to zostać wykonane przez wysłanie ruchu, który w odpowiedzi wygeneruje jeszcze większą ilość ruchu na łączach, można przepełnić bazy routerów lub wstrzyknąć informacje o nieistniejących trasach to tablic routingu lub takie trasy aby spowodować pętle routingu.

Zużycie pasma może zostać sprowokowane dzięki wysłaniu wielu niepotrzebnych pakietów. Może to zostać osiągnięte także w lepszy dla atakującego sposób, czyli wykorzystując pakiety LSA Request.

4.3.4.1 Wysyłanie pakietów Link State Request

Pakiety LSA Request jak opisano w 3.3.3.3 służą do uzupełniania informacji o scieżkach. Jeżeli atakujący będzie ciągle wysyłał pakiety LSA Request, wtedy odbiorca będzie musiał odpowiedzieć na każde zapytanie pakietem LSA Update opisanym w 3.3.3.4. Jeżeli weźmiemy pod uwagę ograniczenie jakim jest MinLSInterval(domyślnie 5 sekund) wtedy możemy zakładać, że pasmo zużyte przez ten atak jest ograniczone. Z drugiej strony, pakiet LSA Request może zawierać kilka osobnych zapytań, przez co zwiększa wielkość odpowiedzi, czyli pakiet LSA Update.

Tutaj administrator może zwiększyć czas MinLSInterval, aby uniknąć zbyt częstych uaktualnień. Jednak chcąc zapobiec temu atakowi poprzez zmianę tego parametru, narażamy się na ataki, w których mechanizm obronny OSPF mógłby nas strzec, jednak tylko jeżeli parametr MinLSInterval jest niski. Powyższe jednoznacznie świadczy o tym, że każda zmiana konfiguracji ustawień OSPF musi być rozważana wielotorowo, aby uniknąć sytuacji w której chcąc uniknąć jednego ataku, otwieramy sieć na kolejne.

4.3.4.2 Przepełnianie bazy Link State

Atakujący może wygenerować Router LSA lub Network LSA, który będzie zawierał wpisy o nieistniejących routerach. Te pakiety LSA i routery nie wezmą udziału w procesie liczenia trasy wykorzystującym SPF, jednak będą przechowywane w bazie Link-State przez jedną godzinę. Należy nadmienić, że wpisy te nie zostaną usunięte przez mechanizm obronny OSPF, gdyż pakiety te nie istnieją dla tego mechanizmu.

Dodatkową kwestią jest to, że External LSA opisane w 3.3.4.6 mogą zostać użyte do tego celu. Jak poprzednio nie są one brane pod uwagę przy procesie SPF, więc mechanizm obronny OSPF nie zadziała. Kluczowym aspektem w ataku tymi pakietami jest to, że zostaną one rozesłane do wszystkich routerów oprócz tych w obszarach Stub, powodując przepełnienie bazy w większej liczbie routerów. Fałszywe wpisy zostaną usunięte z bazy dopiero po upłynięciu czasu Maximum Link-State Age, który domyślnie wynosi jedną godzinę.

4.3.4.3 Przepełnianie Listy Retransmisji

Routery utrzymują listę pakietów LSU, których odbiór nie został potwierdzony jak i listę zapytań, które zostały bez odpowiedzi. Jest oczywiste, że dzięki temu atakujący, generując informacje o niestniejących routerach lub sieciach, spodziewa się braku odpowiedzi do routera który o nie zapyta(Ack lub Update).

Możemy rozważyć dwa przykładowe scenariusze wykorzystania tego ataku;

Jeżeli router odbierze Database Description Packet(w stanie Exchange) z nagłówkiem LSA, które wskazuje na pochodzenie z fałszywego routera, to wtedy LSA będzie zachowane w Link-State Database. Ofiara wyśle zapytanie o więcej informacji na temat tego LSA, które właśnie otrzymała używając pakietu Link-State Request. Naturalnie ofiara nie otrzyma

odpowiedzi na to zapytanie i będzie przetrzymywać informacje o tym w swojej liście retransmisji.

Kolejnym scenariuszem wykorzystania tego ataku jest nie potwierdzanie odebrania pakietu LSA Update. Jeżeli odbiorca jest nieistniejącym routerem lub gdy przez przypadek nie potwierdzi odbioru pakietu LSA Update, ofiara będzie przetrzymywać informacje o tym w swojej liście retransmisji. Wykonywanie tej czynności w sposób ciągły, może pochłonąć zasoby routera.

5 Wykorzystanie ataków, opis przypadków użycia

Znając już zasady działania protokołów routingu oraz opis możliwych sposobów ataków, możemy przystąpić do zaplanowania ataku na router.

Ataków uniemożliwiających działanie sieci jest sporo, jednak nie przynoszą one zysku w postaci przechwyconych informacji. Z tego punktu widzenia atak, w którym możemy sterować przepływem danych jest bardziej istotny i niebezpieczny. Jest tak dlatego iż możemy przechwycić newralgiczne dane, jak i w ostateczności unieruchomić sieć.

5.1 Wstrzyknięcie fałszywej trasy w RIP

Protokół RIP jest jednym z pierwszych protokołów routingu dynamicznego. Mimo swoich wielu ograniczeń, jest wciąż szeroko stosowany, zwłaszcza w małych sieciach, gdzie skaluje się idealnie. Jego ciągłe istnienie w środowiskach produkcyjnych spowodowane jest po części dlatego że jest on prosty w konfiguracji, po części ze względu na inercje środowiska oraz po części dlatego że wiele urządzeń sieciowych nie wspiera innych protokołów.

Jedną z podstawowych zalet RIPv2 jest to, iż posiada on autoryzacje uaktualnień, czy to podpisywaną czystym tekstem czy zaszyfrowanym. Niestety z różnych powodów wciąż możliwe jest zaatakowanie takich sieci, gdyż administratorzy nie używają uwierzytelniania w ogóle lub stosują uwierzytelnianie czystym tekstem.
Ponad to zawsze istnieje szansa ataku typu downgrade version, która zmusi ofiarę do pracy pod kontrolą RIPv1, gdzie nie jest wspierana żadna autoryzacja uaktualnień.

Jak wspomniano w we wcześniejszym rozdziale, protokół RIP jest bezpołączeniowy i działa na porcie 520 UDP. Dzięki temu faktowi, łatwo jest więc wysłać sfałszowane pakiety.

Pierwszą rzeczą jaką atakujący musi wykonać, to przechwycenie pakietów RIP, które rozsyłane są domyślnie co 30 sekund lub przy zmianie topologii sieci. Innym sposobem na przechwycenie pakietów jest odpytanie o to routera, używając narzędzia „ass” napisanego przez grupę Phenoelit’s.

```
arhontus / # ass -v -i eth0 -D <router IP> -P <1 | 2>
```

gdzie 1 | 2 to wersje RIP.
Drugą możliwością jest użycie rprobe:

```
arhontus / # rprobe -a -v <router IP>
```

gdzie –v oznacza RIPv2. W przeciwieństwie do „ass” rprobe nie przechwyci żadnych danych, więc należy uruchomić sniffer na odpowiednim interfejsie.

```
arhontus / # tcpdump -i eth0 host <router IP>
```

Oto przykładowy pakiet jaki możemy przechwycić w sieci, w której działa RIP:

```
arhontus / # tcpdump -i eth0 host 192.168.66.202 -vv
     tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size
68 bytes
     <RIP request is sent with rprobe -a -v 192.168.66.202>
     22:47:20.941167 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF],
length:
     52) tester.arhont.com.route > tested.arhont.com.route: [udp sum ok]
             RIPv2, Request, length: 24
             0x0000: 0102 0000 0000 0000 0000 0000 0000 0000
             0x0010: 0000 0000 0000 0010
     22:47:20.944184 IP (tos 0xc0, ttl 255, id 0, offset 0, flags [none],
     length: 72) tested.arhont.com.route > tester.arhont.com.route:
             RIPv2, Response, length: 44, routes: 2
               AFI: IPv4: 192.168.30.0/24, tag 0x0000, metric: 1, next-hop:
      self[|rip]
              0x0000: 0202 0000 0002 0000 c0a8 1e00 ffff ff00
              0x0010: 0000 0000 0000 0001 0002 0000 0000 0000
              0x0020: 1900 0000 686f 7374 2031 3932
```

Kiedy już rozpoznaliśmy sieć i wiemy, że RIP działa w tej sieci, znamy jego wersje, mamy wiedze o sposobie autoryzacji(brak lub czysty tekst) i znamy topologie sieci, możemy wtedy przystąpić do ataku.

Najprostrzym przypadkiem jest przekserowanie ruchu do specjalnie przeznaczonej do tego celu maszyny, na której uruchomiony jest sniffer lub innej maszyny pod nasza kontrolą. Należy pamiętać, że długość trasy jest ograniczona do 15 przeskoków.
Oczywiście komputer przez który będziemy kierować ruch musi mieć włączone przekazywanie pakietów:

```
arhontus / # echo 1 > /proc/sys/net/ipv4/ip_forward
```

lub

```
arhontus / # fragrouter -B1
```

W tym momencie rozgłaszamy naszą maszynę jako router RIP z metryką najlepszą dla sieci, do której transferowane dane chcemy przechwytywać. Idealnym do tego celu programem jest Quagga lub inne narzędzie symulujące działanie routera.

Typową konfiguracją dla demona „ripd” możemy zaobserwować poniżej, w której używamy RIPv2 z pojedynczym kluczem autoryzacyjnym:

```
!
     ! Zebra configuration saved from vty
     ! 2005/08/12 23:44:33
     !
     hostname legitimate.ripd
     password 8 jhahnGuSsan.g
     enable password 8 Cb/yfFsI.abqs
     log file /var/log/quagga/ripd.log
     service advanced-vty
     service password-encryption
     !
     !
     key chain dmz_auth
      key 1
       key-string secret_key
     !
     interface eth0
      description DMZ_network
      ip rip authentication mode md5 auth-length old-ripd
      ip rip authentication key-chain dmz_auth
     !
     router rip
      version 2
      redistribute connected
      network 192.168.20.0/24
     !
     line vty
      exec-timeout 30 0
     !
```

Załóżmy, że chcemy przepuścić cały ruch przez nasz podstawiony komputer oparty o system Linux. Zakładając, że nasze uaktualnienie zostanie zaakceptowane przez ofiarę, musimy poczynić pewne dodatkowe kroki, aby wszystko przebiegło bez zarzutu.

Musimy włączyć przekazywanie pakietów:

```
arhontus / # echo 1 > /proc/sys/net/ipv4/ip_forward
```

Następnie, jeżeli chcemy aby przez naszą maszynę przechodziły pakiety w obie strony – przychodzące i wychodzące, musimy użyć NAT (Network Address Translation), tak aby odpowiedzi na pakiety wychodzące które wyślemy od nas do bramy, zamiast od hosta do bramy bezpośrednio, trafiały powrotem do nas, a nie do oryginalnego nadawcy. Jeżeli brama domyślna również bierze udział w procesie routingu, to w jej tablice również możemy wstrzyknąć fałszywe wpisy.

NAT może być łatwo ustawiony przy pomocy pakietu Netfilter, znajdującego się praktycznie w każdej dystrybucji Linux:

```
arhontus / # iptables -t nat -A POSTROUTING -o eth0 -s victim_IP -j SNAT
     --to-source your_IP
```

Kiedy poczyniliśmy przygotowania, można spokojnie wstrzykiwać fałszywe wpisy. Tutaj pojawia się dość ważna kwestia, a mianowicie musimy zaobserwować czy nasze wstrzyknięcia są akceptowane przez ofiarę. Ponad to, musimy wstrzykiwać pakiety używając metody unicast, zamiast broadcast czy multicast, aby uniknąć pętli routingu.

W systemie operacyjnym Linux, standardowy mechanizm redystrybucji tras statycznych nie znajdzie zastosowania, gdyż jeżeli ustawimy trasę statyczną, tak aby cały ruch przechodził przez naszą maszynę, to wtedy nie będziemy w stanie wytransferować żadnego pakietu.

Na szczęście, autorzy aplikacji Quagga przewidzieli taką potrzebę i bez potrzeby definiowania statycznych tras w systemie, możemy stworzyć takową w osobnym pliku i redystrybuować po przez protokół RIP. Jedyne co musimy skonfigurować w Quagga, aby tego dokonać znajduje się poniżej:

```
router rip
      version 2
      default-information originate
      neighbor 192.168.20.200
      route 198.133.219.25/32
```

W listingu powyżej mamy użyte: neighbor, co oznacza, że wyślemy uaktualnienia tylko do tej maszyny, default-information originate pozwoli nam wysłać naszą statyczną trasę oraz oczywiście route, które typuje sieć, którą obraliśmy sobie jako ofiarę.

Oto wszystko co należy wykonać. Jedyną kwestią jaką należy jeszcze uwzględnić, to hasła uwierzytelniające; należy je przechwycić lub złamać.

5.2 Wstrzyknięcie fałszywej trasy w OSPF

Atak na protokół OSPF jest bardziej skomplikowany niż na protokoły wektorowo-odległościowe. Jest tak z powodu ich bardziej skomplikowanej architektury i działania, tj.:

- Proces przyłączania sąsiednich routerów poprzez wymiane pakietów Hello. Fałszywy router musi wymienić odpowiednio spreparowane pakiety Hello, aby być akceptowanym w wymianie informacji o trasach.
- Występowanie hierarchii, takiej jak obszary OSPF. Każda informacja o obszarach oraz przekraczająca te obszary musi być wzięta pod uwagę podczas ataku. Z drugiej jednak strony, występowanie DR, jako głównego routera dla danego obszaru, stwarza nowe możliwości ataku.

Kiedy chcemy zaatakować siec poprzez wstrzyknięcie fałszywych tras, skomplikowanie procesu wymusza najbardziej zautomatyzowane metody. Jedną z metod jest przejęcie i przekonfigurowanie istniejącego routera w sieci. Drugim sposobem jest uruchomienie w sieci komputera z aplikacją symulującą działanie routera, takich jak Quagga.

Celem ataku jest ogłoszenie naszego routera, np. maszyny pod kontrolą systemu Linux z włączonym przekazywaniem pakietów. Nasz router powinien być uznany jako następny przeskok dla możliwie największej ilości sieci docelowych. Otrzymać to można poprzez ogłoszenie routera z jak najniższym kosztem lub możliwie największym dostępnym pasmem, które jest brane pod uwagę do obliczania kosztu trasy, tak jak wspomniano w rozdziale 3.3.1.

6 Przykładowe ćwiczenia laboratoryjne

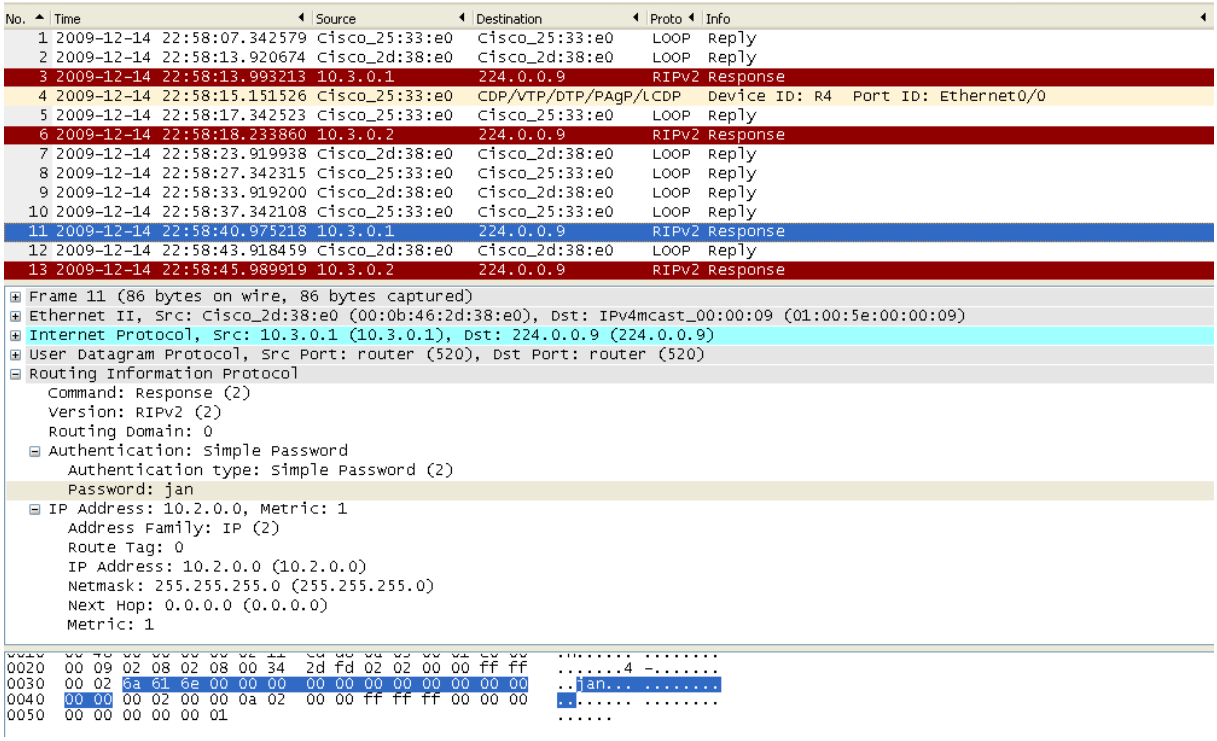
Poniżej znajdują się przykładowe ćwiczenia laboratoryjne wraz z krótką instrukcją przygotowania i przeprowadzenia.

Sposób zmuszenia sieci do wysyłania pakietów również do nas jest poza obszarem zainteresowań tych ćwiczeń. Dla uproszczenia ćwiczenia do poziomu zadania akademickiego w ćwiczeniach zastosowano koncentrator zamiast przełącznika.

6.1 RIPv2 – podsłuchiwanie, przechwycenie hasła nieszyfrowanego

Przechwycenie hasła przesyłanego w sposób jawny jest najprostszym zadanie. Do tego celu użyjemy najpopularniejszego analizatora pakietów – Wireshark. Posiada on wbudowany system rozpoznawania typu pakietu i wyświetlania poszczególnych wartości przypisanych do pól w danym pakiecie.

1. Uruchamiamy przechwytywanie pakietów.
2. Czekamy aż przybędą pakiety protokołu RIP. Warto tutaj odczekać chwilę, aby zebrać pakiety RIP od wszystkich węzłów znajdujących się w domenie rozgłoszeniowej. Pomoże to atakującemu przeanalizować i odtworzyć topologię sieci możliwie dokładnie.
3. Po odebraniu pakietów, należy odczytać hasło. Oznaczenia pól są intuicyjne.



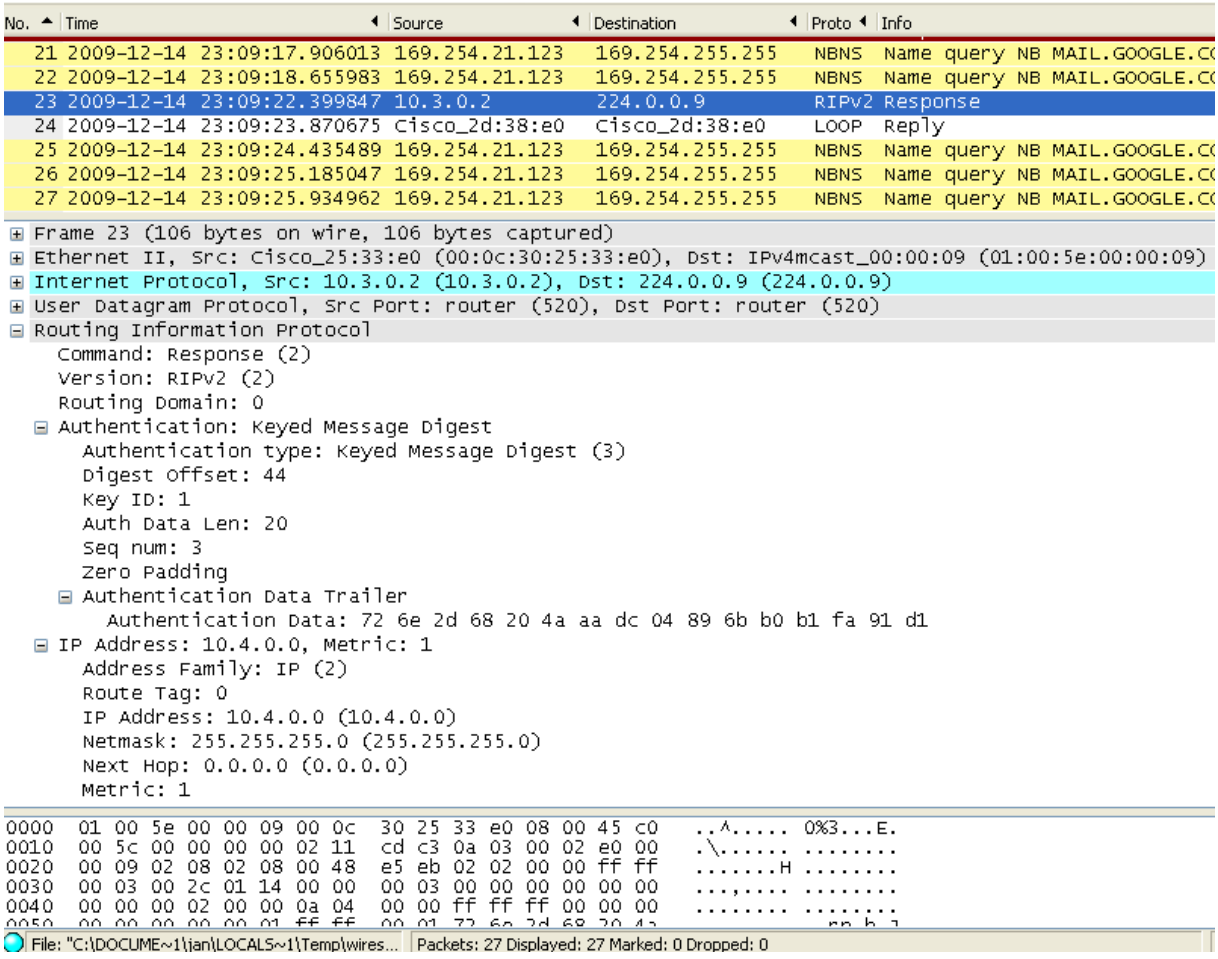
Jak widać hasło jest przesyłane w sposób całkowicie jawny i łatwy do odczytania, co czyni tę metodę uwierzytelniania mało skuteczną w praktyce.

6.2 RIPv2 – podsłuchiwanie, łamanie uwierzytelniania MD5

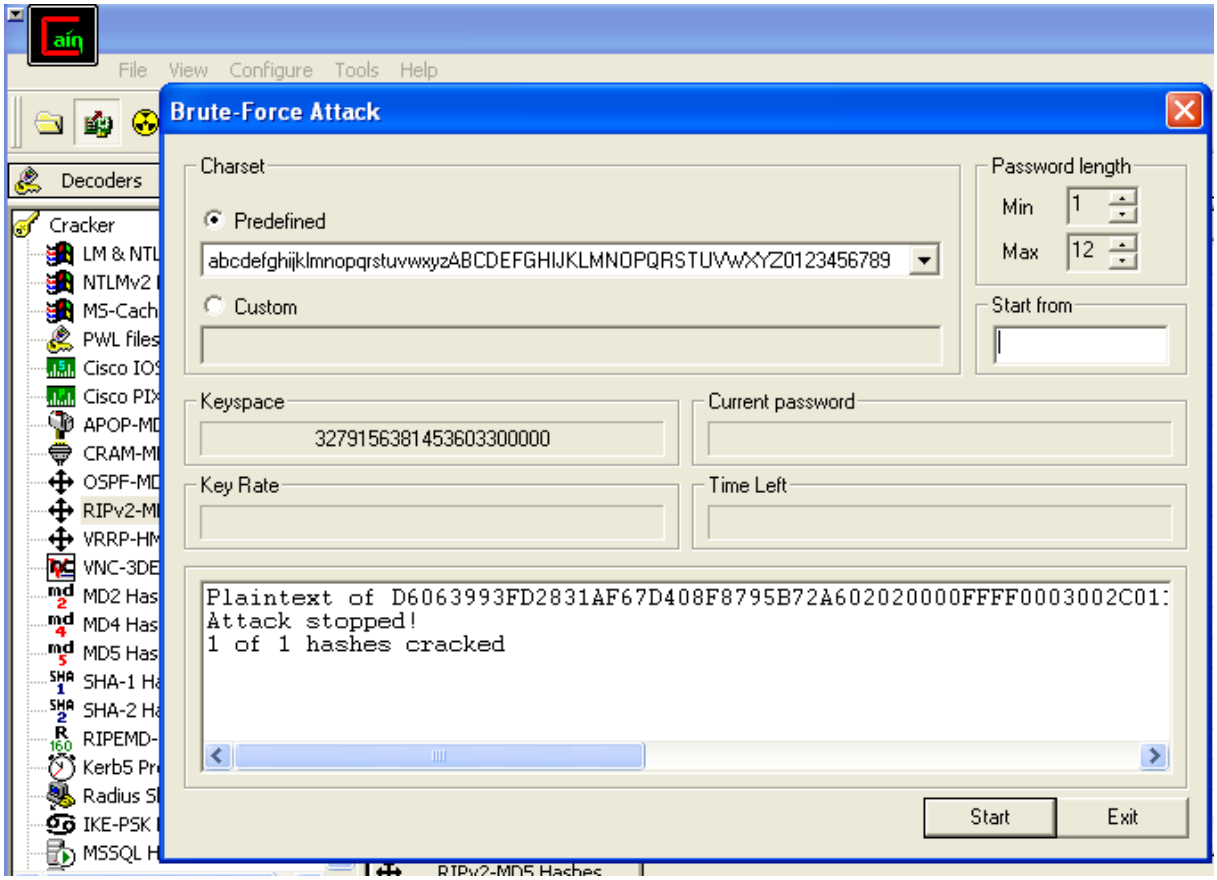
Przechwycenie klucza na podstawie którego generowane są skróty podpisujące pakiet jest zadaniem wymagającym użycia dużej mocy obliczeniowej, stosownej do poziomu skomplikowania hasła. Niestety skrót MD5 nie jest odwracalny, co oznacza, że aby odgadnąć na podstawie jakiego słowa generowany jest skrót, musimy wygenerować takie słowo a z niego skrót i porównać. W zależności od rodzaju zastosowanych znaków i długości klucza, czas potrzebny na odgadnięcie hasła może się drastycznie wydłużyć. Z tego powodu dla celów ćwiczenia należy wybrać hasło do 6 znaków składające się z małych liter.

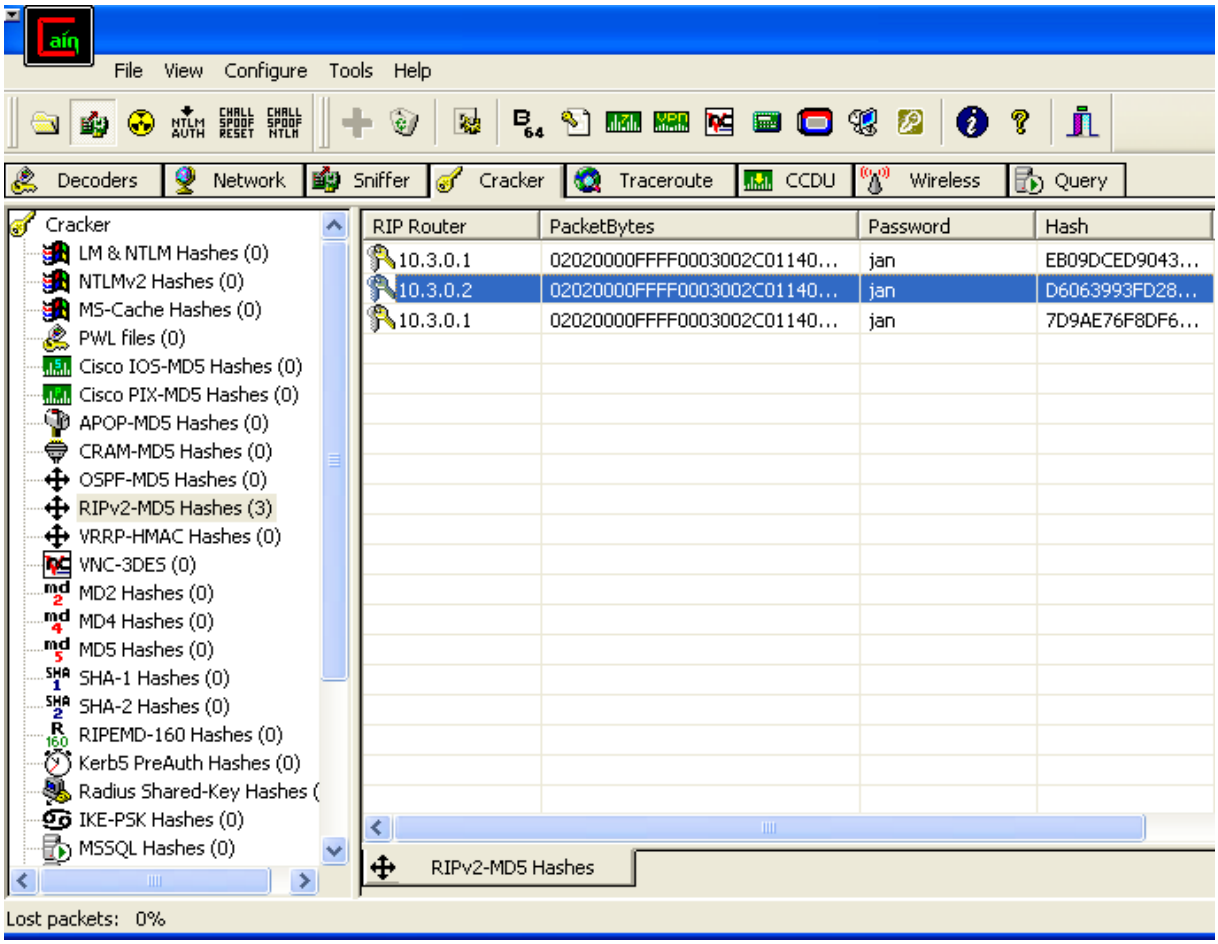
Do przechwycenia i próby odgadnięcia klucza wykorzystamy narzędzie Cain & Abel. Posiada on wbudowany mechanizm przechwytywania pakietów, ich analizowania i późniejszego łamania skrótów MD5.

1. Uruchamiamy przechwytywanie pakietów.
2. Czekamy aż przybędą pakiety protokołu RIP. Warto tutaj odczekać chwilę, aby zebrać pakiety RIP od wszystkich węzłów znajdujących się w domenie rozgłoszeniowej. Pomoże to atakującemu przeanalizować i odtworzyć topologię sieci możliwie dokładnie.



3. Po przechwyceniu żądanych pakietów przesyłamy je do Cracker. Tam wybieramy zbiór możliwych znaków oraz długość hasła. Innym wyborem może być atak słownikowy, jest on szybszy od ataku Brute-Force jednak musimy być pewni że szukana fraza znajduje się w słowniku.
4. Po odgadnięciu hasła program nas poinformuje jaka fraza jest używana do generowania skrótów.





6.3 RIPv2 – Zatrucie trasy, rozłączenie, ilość przeskoków 16

Atak polega na wstrzyknięciu pakietu UDP do atakowanego węzła z adresem prawdziwego nadawcy. Jest to łatwe do wykonania ze względu na bezpołączeniowy sposób komunikacji za pomoć protokołu UDP. Trasę którą chcemy uznać za nieosiągalną, wstrzykniemy z odległością przeskoków równą 16.

6.4 RIPv2 – Zatrucie trasy, rozłączenie, ilość przeskoków 16 – uwierzytelnianie proste

Atak polega na wstrzyknięciu pakietu UDP do atakowanego węzła z adresem prawdziwego nadawcy. Jest to łatwe do wykonania ze względu na bezpołączeniowy sposób komunikacji za pomocą protokołu UDP. Trasę którą chcemy uznać za nieosiągalną, wstrzykniemy z odległością przeskoków równą 16.

Dodatkowym utrudnieniem jest wykorzystanie uwierzytelniania pakietów z wykorzystaniem ustalonego wcześniej hasła. Jak wspomniano w 3.2.10 istnieją trzy tryby

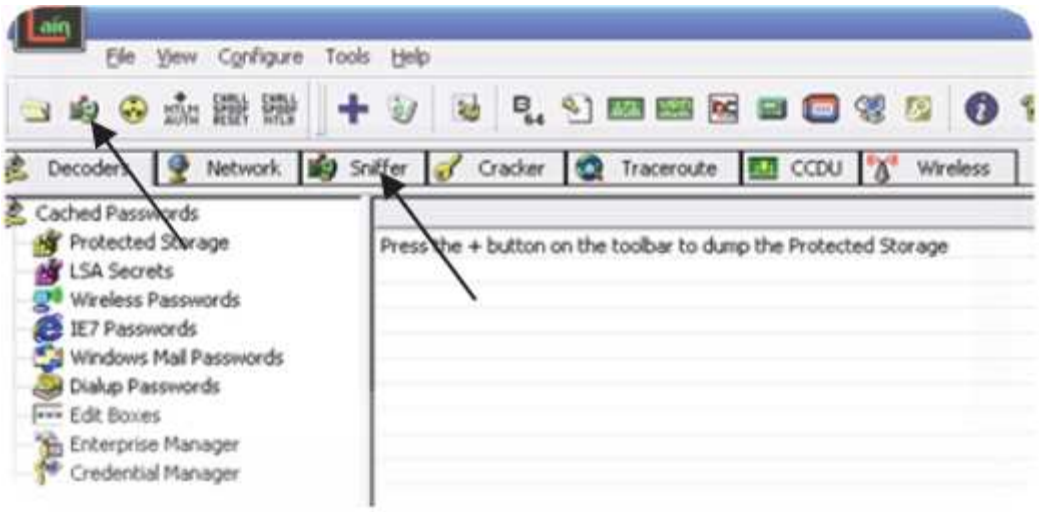
uwierzytelnienia nadawcy i ewentualnego sprawdzenia integralności pakietu. W tym ataku założymy, że komunikacja chroniona jest prostym uwierzytelnianiem. Implementacja ataku na uwierzytelnianie MD5 różni się tylko trudnością implementacji i budową pakietu, jednak schemat pozostaje ten sam.

6.5 OSPF – przechwytywanie i przełamywanie klucza uwierzytelniania MD5

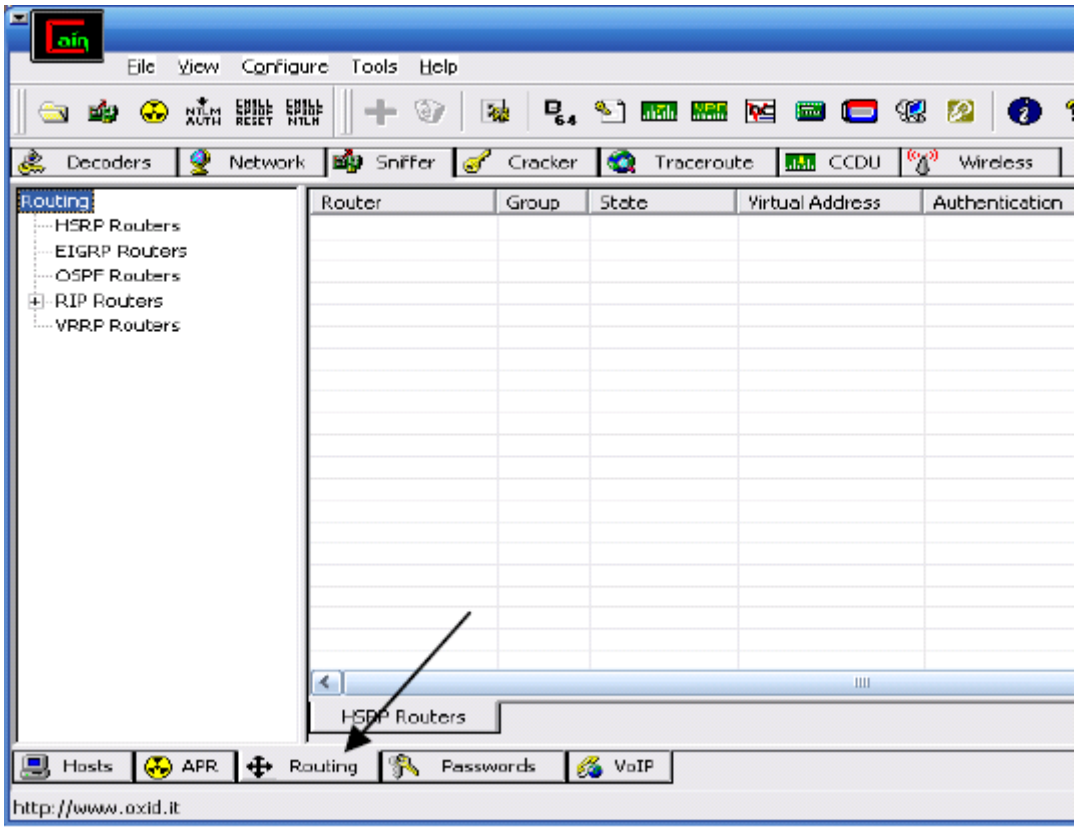
Protokół OSPF posiada trzy typy autoryzacji i uwierzytelniania swoich pakietów. Pierwszy z nich to przesyłanie danych bez żadnego hasła. Drugi to przesyłanie pakietu uwierzytelnionego hasłem w postaci jawnego tekstu. Trzeci sposób to podpisanie pakietu skrótem MD5. Pierwsze dwa sposoby z oczywistych powodów nie zapewniają dostatecznej ochrony. Skrót MD5 jest powszechnie znany jako słabe zabezpieczenie, a dodatkowo uwierzytelnia on tylko dane nagłówka OSPF, natomiast nagłówek IP nie jest nim chroniony.

W tym ćwiczeniu przełamany został najsilniejszy możliwy sposób uwierzytelniania. Schemat ataku jest prosty, należy przechwycić uwierzytelniony pakiet a następnie użyć odpowiedniego programu do złamania skrótu. Do tego celu możemy użyć domowego komputera, dedykowanych klastrów udostępnionych w sieci lub tablic z gotowymi już rozszyfrowanymi skrótami.

Uruchamiamy program Cain&Abel. Wybieramy zakładkę „Sniffer” następnie klikamy ikonkę  i wybieramy interfejs z którego chcemy podsłuchiwać ruch.



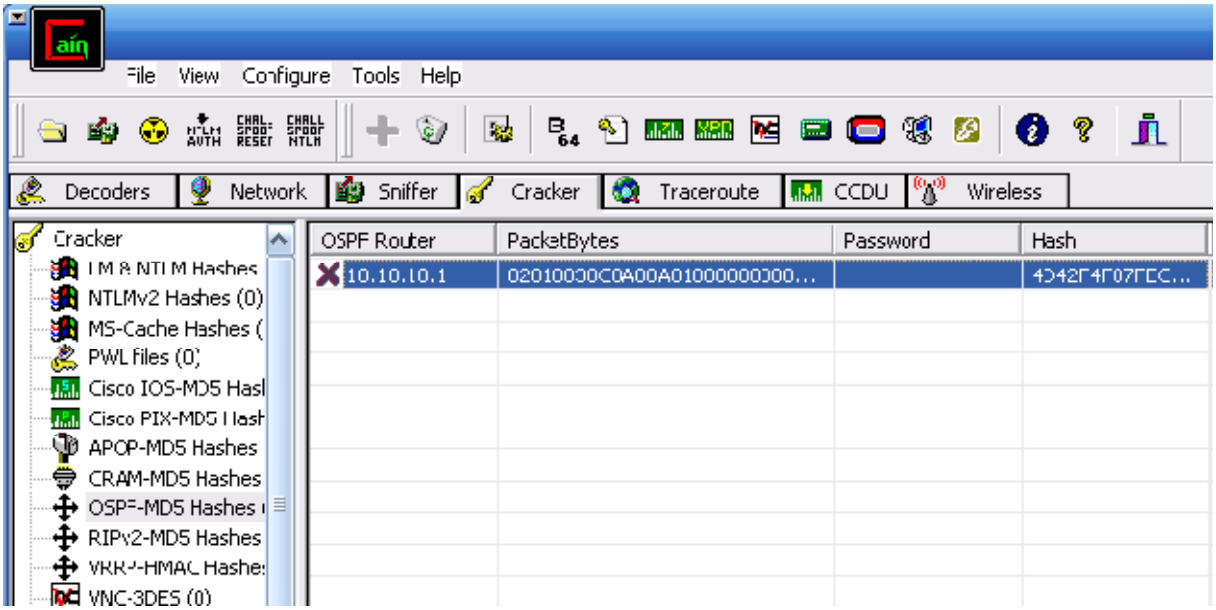
Jeżeli podsłuchiwanie już się rozpoczęło, przechodzimy do zakładki „Routing", gdzie możemy zobaczyć informacje o ruchu przechwyconym i związanym z protokołami trasowania.



Kiedy zostaną przechwycone informację dotyczące OSPF, otrzymamy wszystkie szczegóły na temat pakietu w postaci tabeli. Następnie klikając prawym klawiszem myszy w

wybrany pakiet w tabeli możemy wybrać chęć wysłania danych do „Cracker”(ang. łamacz). Bierze on pod uwagę jedynie pola Last Hash oraz PacketBytes.

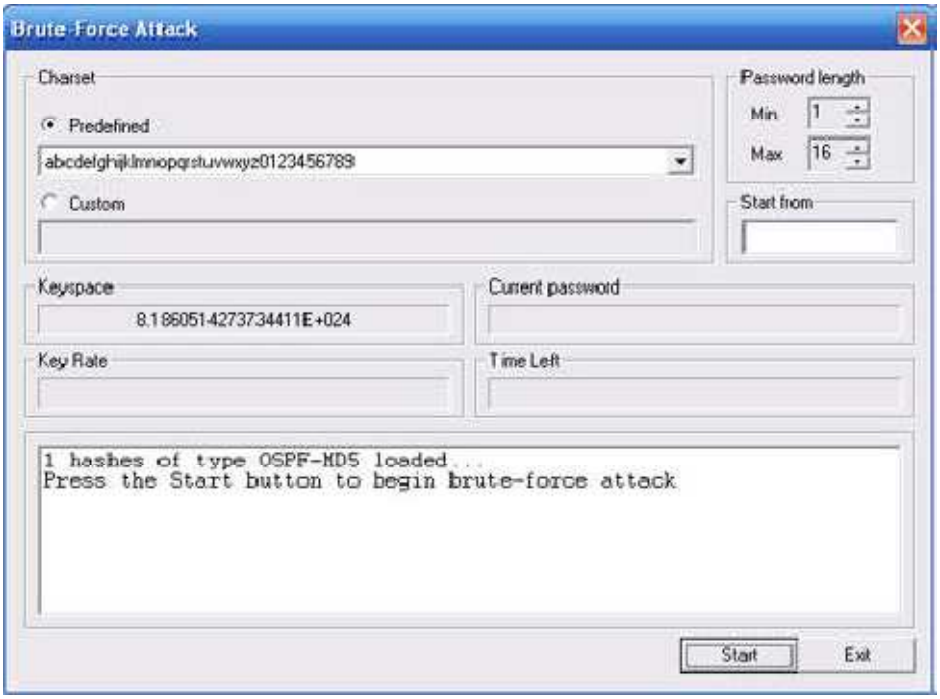
W dalszej kolejności przechodzimy do zakładki „Cracker” i wybieramy z lewego menu opcję „OSPF-MD5 Hashes”. Powinniśmy widzieć dane o pakietach, które w poprzednim kroku zostały przetransferowane do „Crackera”.



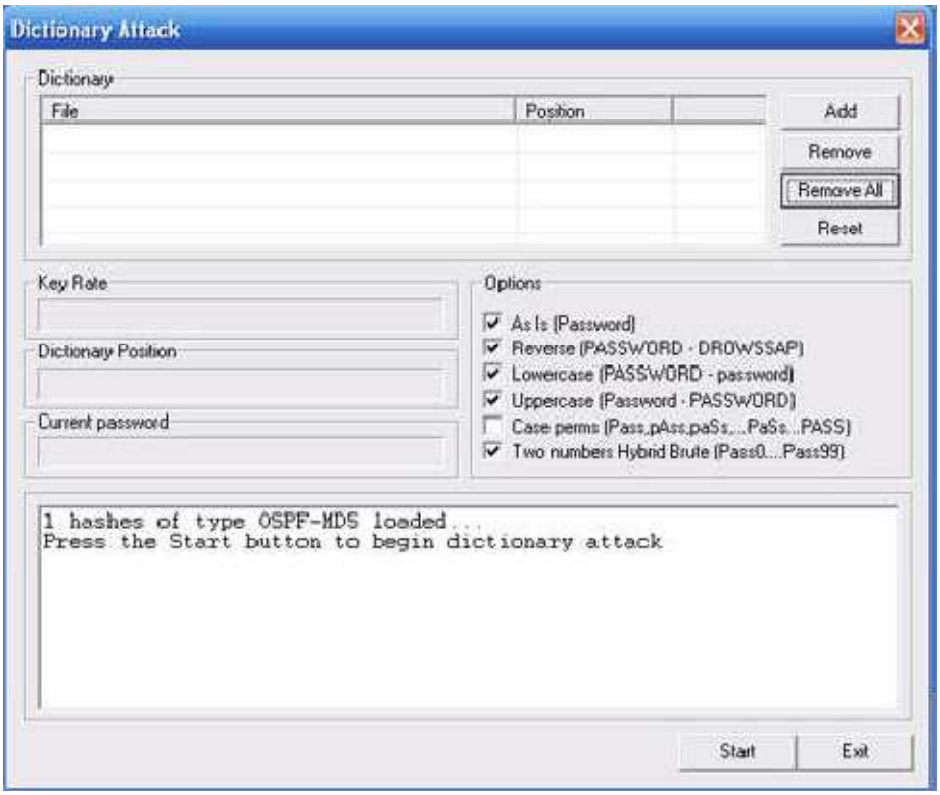
Kolejnym krokiem jest wybranie sposobu złamania skrótu. Metoda tutaj wykorzystywana to klasyczne generowanie skrótu z czystego tekstu i dopasowanie skrótów wygenerowanego i podsłuchanego. Wszystkie opcje dostępne są przez kliknięcie prawym przyciskiem myszy na wybrany wiersz. Poniżej znajdują się możliwe do wyboru sposoby:

Mamy trzy sposoby otrzymania klucza. Jeżeli podejrzewamy tekst jaki kryje się w skrócie, możemy użyć opcji „Test password”, jednak w większości przypadków nie będziemy go znali. Do wyboru wtedy mamy pozostałe dwa sposoby.

Atak „Brute-Force” spróbuje dopasować każdą możliwą kombinację z zadanego zbioru znaków. Zakres długości możliwego klucza także może zostać wybrany. Atak ten jest z reguły wolniejszy, jednak otrzymujemy większa szansę na znalezienie prawidłowego klucza generującego skrót. Poniżej znajduje się zrzut ekranu opcji możliwych do wybrania przed uruchomieniem ataku „Brute-Force”.



Inny wybór to atak słownikowy. Atak ten użyje do porównania słowa z pliku oraz wygeneruje niektóre ich mutacje. Sposób ten uważany jest za szybszy, jednak warunkiem koniecznym jest występowanie klucza w wykorzystywanym zbiorze. Poniżej znajduję się zrzut ekranu z możliwymi opcjami.



Jeżeli uruchomiliśmy któryś z wyżej wymienionych ataków, musimy poczekać aż słowo zostanie odnalezione. Sytuację taką przedstawia obraz poniżej.

Posiadając już hasło użyte do autoryzacji i sprawdzania integralności pakietu możemy go użyć do zmiany przechwyconego pakietu lub nawet do wygenerowania własnego i wstrzyknięcia go do sieci. Nowy skrót musi być wygenerowany używając znalezionego klucza.

[1] Krawetz, Neal. "Chapter 3 - Network Theory". Introduction to Network Security. Cengage Charles River Media. © 2007. Books24x7.

[2] Kozierok, Charles M.. "Chapter 5 - General OSI Reference Model Issues and Concepts". The TCP/IP Guide. No Starch Press. © 2005. Books24x7.

[3] Krutz, Ronald L., and Russell Dean Vines. "Chapter 3 - Telecommunications and Network Security". The CISSP Prep Guide: Gold Edition. John Wiley & Sons. © 2003. Books24x7.

[4] Lammle, Todd. "Chapter 6 - IP Routing". CCNA: Cisco Certified Network Associate Study Guide, Sixth Edition, (Exam 640-802). Sybex. © 2007. Books24x7.

[5] Macfarlane, James. "Chapter 5 - RIP". Network Routing Basics: Understanding IP Routing in Cisco Systems. John Wiley & Sons. © 2006. Books24x7.

[6] CCNA3 – Curriculum by Cisco, v3.1.1

[7] Macfarlane, James. "Chapter 5 - RIP". Network Routing Basics: Understanding IP Routing in Cisco Systems. John Wiley & Sons. © 2006. Books24x7.

[8] Naugle, Matt. "Chapter 2 - The Protocol Suite of TCP/IP". Illustrated TCP/IP: A Graphic Guide to the Protocol Suite. John Wiley & Sons. © 1999. Books24x7.

[9] Timm, Carl, and Wade Edwards. "Chapter 5 - OSPF Operation in a Single Area". CCNP: Building Scalable Cisco Internetworks Study Guide (Exam 642-801). Sybex. © 2004. Books24x7. <http://common.books24x7.com/book/id_7300/book.asp>

[10] http://www.cisco.com/en/US/tech/tk365/technologies_white_paper09186a0080094e9e.shtml#t5

[11] http://www.cisco.com/en/US/docs/internetworking/technology/handbook/OSPF.html

[12] RFC 2328