Leksykon: Sieci Komputerowe

Dla inżynierów oraz techników

Kacper Ostrowski

Ostrowski, K. (2025). Leksykon: Sieci Komputerowe – Dla inżynierów oraz techników (1.0). Zenodo. https://doi.org/10.5281/zenodo.15341381

Opracowanie własne

Krótka historia powstania sieci komputerowych

Początki sieci komputerowych

Historia sieci komputerowych sięga lat 60. XX wieku, kiedy to rozwój technologii komputerowej i potrzeba efektywnej komunikacji między różnymi ośrodkami badawczymi stały się priorytetem. W 1962 roku Paul Baran z RAND Corporation przedstawił koncepcję sieci rozproszonej, zdolnej do przetrwania uszkodzeń części jej struktury, co było kluczowe w kontekście zimnej wojny .

ARPANET – narodziny Internetu

W 1969 roku, dzięki finansowaniu przez Agencję Zaawansowanych Projektów Badawczych Departamentu Obrony USA (ARPA), uruchomiono ARPANET – pierwszą sieć komputerową opartą na technologii komutacji pakietów. Początkowo łączyła ona cztery ośrodki akademickie: Uniwersytet Kalifornijski w Los Angeles (UCLA), Instytut Badawczy Stanforda (SRI), Uniwersytet Kalifornijski w Santa Barbara (UCSB) oraz Uniwersytet Utah .

Rozwój i standaryzacja protokołów komunikacyjnych

W miarę rozwoju ARPANET pojawiła się potrzeba standaryzacji protokołów komunikacyjnych. W 1974 roku Vinton Cerf i Bob Kahn opublikowali specyfikację protokołu TCP/IP, który stał się fundamentem współczesnego Internetu . W 1983 roku ARPANET oficjalnie przeszedł na protokół TCP/IP, co umożliwiło dalszą ekspansję sieci i jej interoperacyjność z innymi systemami.

Powstanie sieci lokalnych (LAN) i rozległych (WAN)

Lata 70. i 80. XX wieku to okres intensywnego rozwoju sieci lokalnych (LAN) oraz rozległych (WAN). W

1973 roku Robert Metcalfe opracował Ethernet – technologię umożliwiającą efektywną komunikację w sieciach lokalnych . Jednocześnie rozwijały się sieci rozległe, łączące odległe ośrodki i umożliwiające globalną wymianę informacji.

Internet w Polsce

Do Polski Internet dotarł na początku lat 90. XX wieku. W 1991 roku Naukowa i Akademicka Sieć Komputerowa (NASK) nawiązała pierwsze połączenie z siecią EARN, co zapoczątkowało rozwój Internetu w środowisku akademickim i naukowym w kraju .

Kamienie milowe w rozwoju sieci komputerowych

Kluczowe wydarzenia w historii rozwoju sieci komputerowych

Rok	Wydarzenie
1962	Paul Baran przedstawia koncepcję sieci rozproszonej .
1969	Uruchomienie ARPANET, pierwszej sieci komputerowej opartej na komutacji pakietów .
1971	Wprowadzenie systemu poczty elektronicznej .
1973	Opracowanie technologii Ethernet przez Roberta Metcalfe’a .
1974	Publikacja specyfikacji protokołu TCP/IP przez Vintona Cerfa i Boba Kahna .
1983	ARPANET przechodzi na protokół TCP/IP, ustanawiając standard dla przyszłego Internetu .
1991	NASK nawiązuje pierwsze połączenie z siecią EARN, wprowadzając Internet do Polski .

Podsumowanie

Rozwój sieci komputerowych od lat 60. XX wieku do współczesności to proces dynamiczny, napędzany potrzebą efektywnej komunikacji i wymiany informacji. Od pionierskich prac nad ARPANET, przez standaryzację protokołów komunikacyjnych, po globalną ekspansję Internetu – każdy etap tej historii przyczynił się do ukształtowania świata, w którym obecnie żyjemy.

Historyczne technologie sieciowe

Token Ring

Token Ring to technologia sieciowa, która była szeroko stosowana w latach 80. i 90. XX wieku. Sieć Token Ring korzysta z topologii pierścienia, w której każdy węzeł (komputer lub urządzenie) jest połączony z dwoma sąsiednimi węzłami. Działa ona na zasadzie przesyłania specjalnego pakietu, zwanego tokenem, który daje urządzeniu prawo do nadawania danych.

Zastosowanie

Token Ring był wykorzystywany w dużych firmach oraz korporacjach, zwłaszcza w systemach, które wymagały wysokiej niezawodności w przesyłaniu danych. Technologie takie jak Ethernet i Wi-Fi ostatecznie wyparły Token Ring, ale przez pewien czas była to popularna opcja w sieciach lokalnych.

Działanie urządzeń

Urządzenia w sieci Token Ring są połączone w sposób szeregowy, tworząc pierścień. Token przesuwa się w jednym kierunku, a urządzenie może nadawać dane tylko, jeśli posiada token. Każde urządzenie ma swoją unikalną adresację, a dane są przesyłane w sposób sekwencyjny przez sieć, aż dotrą do docelowego urządzenia.

Sygnały

Token Ring używa sygnałów elektromagnetycznych do przesyłania danych. W zależności od wersji, Token Ring pracuje na różnych prędkościach transmisji (4 Mbps lub 16 Mbps), a dane są przesyłane za pomocą sygnałów o określonej częstotliwości.

Dial-up

Dial-up to technologia szerokopasmowa, która używa linii telefonicznych do przesyłania danych. Użytkownik musi wykonać połączenie z serwerem za pomocą modemu, który przekształca sygnały cyfrowe na analogowe i vice versa.

Zastosowanie

Dial-up był używany powszechnie do połączeń internetowych w latach 90. XX wieku. Wymagał użycia modemu, a połączenie z Internetem odbywało się poprzez zwykłą linię telefoniczną. Technologie DSL i kablowe stopniowo wyparły Dial-up.

Działanie urządzeń

Modem w technologii dial-up przekształcał dane cyfrowe na analogowe sygnały, które mogły być

przesyłane przez linię telefoniczną. Modem odbierał sygnały od dostawcy internetu i przekształcał je na sygnał cyfrowy, który mógł zostać odczytany przez komputer.

Sygnały

Dial-up używa analogowych sygnałów do przesyłania danych, najczęściej w zakresie częstotliwości od 300 Hz do 3 kHz. Sygnały te są modulowane za pomocą różnych metod, takich jak modulacja amplitudy (AM) czy częstotliwości (FM).

ThinNet

ThinNet to rodzaj technologii sieciowej Ethernet, która wykorzystuje cienki kabel koncentryczny. Jest to starsza forma Ethernetu, używana w latach 80. i 90. XX wieku.

Zastosowanie

ThinNet był popularny w małych i średnich sieciach lokalnych, gdzie oszczędność na kablach była kluczowa. Używał cienkiego kabla koncentrycznego, który był tańszy od grubszych kabli wykorzystywanych w innych technologiach Ethernet.

Działanie urządzeń

W technologii ThinNet urządzenia były połączone szeregowo za pomocą kabla koncentrycznego. Przesyłanie danych odbywało się za pomocą sygnałów elektrycznych, a urządzenia „nasłuchiwały” na wspólnej linii, aby odbierać dane.

Sygnały

Sygnały w ThinNet są przesyłane za pomocą prądów elektrycznych, a standardowa prędkość transmisji wynosiła 10 Mbps.

ThickNet

ThickNet, znany również jako 10BASE5, jest starszą wersją Ethernetu, która wykorzystywała gruby kabel koncentryczny. Był bardziej odporny na zakłócenia elektromagnetyczne niż ThinNet, ale bardziej kosztowny.

Zastosowanie

ThickNet był stosowany w większych sieciach komputerowych, w tym w sieciach akademickich i przemysłowych, gdzie wymagano długich odległości połączeń i większej niezawodności.

Działanie urządzeń

W ThickNet urządzenia były połączone z centralnym kablem koncentrycznym, przez który przesyłane były dane. W odróżnieniu od ThinNet, ThickNet wymagał stosowania specjalnych urządzeń do podłączania komputerów, takich jak tzw. transceivery.

Sygnały

Sygnały przesyłane w ThickNet były również sygnałami elektrycznymi, a transmisja odbywała się z prędkością 10 Mbps.

ISDN

ISDN (Integrated Services Digital Network) to cyfrowa sieć, która umożliwia przesyłanie danych, głosu i obrazu przez jedną linię telefoniczną.

Zastosowanie

ISDN było szeroko stosowane w latach 90. w telekomunikacji, szczególnie w połączeniach głosowych oraz w małych firmach, które potrzebowały szybszych prędkości przesyłania danych niż w technologii dial-up.

Działanie urządzeń

Urządzenia ISDN korzystają z cyfrowych linii telefonicznych, które mogą jednocześnie przesyłać głos, dane i obraz. ISDN używa kanałów B (dla danych) i kanałów D (dla sygnalizacji).

Sygnały

ISDN używa cyfrowych sygnałów o różnych przepustowościach, w zależności od konfiguracji. Standardowe prędkości to 64 kbps na kanał B, co daje 128 kbps w trybie podwójnego kanału.

DSL i VDSL (Broadband)

DSL (Digital Subscriber Line) oraz VDSL (Very High-Speed Digital Subscriber Line) to technologie szerokopasmowe, które umożliwiają przesyłanie danych przez tradycyjne linie telefoniczne, oferując wyższe prędkości niż dial-up.

Zastosowanie

DSL i VDSL są stosowane głównie do dostępu do internetu w domach i małych firmach. VDSL oferuje wyższe prędkości przesyłania danych w porównaniu do standardowego DSL.

Działanie urządzeń

Technologie te wykorzystują częstotliwości wyższe niż te używane w tradycyjnych połączeniach telefonicznych, co umożliwia jednoczesne przesyłanie głosu i danych. DSL i VDSL wymagają odpowiednich modemów do konwersji sygnałów.

Sygnały

W DSL i VDSL wykorzystywane są cyfrowe sygnały o różnych częstotliwościach, a prędkości transmisji w DSL wynoszą od 128 kbps do kilku Mbps, natomiast VDSL oferuje prędkości od 13 Mbps do 100 Mbps.

ATM (Asynchronous Transfer Mode)

ATM to technologia przesyłania danych, która była używana w szerokopasmowych sieciach, głównie w telekomunikacji i sieciach WAN.

Zastosowanie

ATM był używany w sieciach telekomunikacyjnych i dużych sieciach WAN, oferując wysoką jakość transmisji dla danych, głosu i obrazu.

Działanie urządzeń

W technologii ATM dane są przesyłane w postaci małych jednostek zwanych komórkami, które mają stałą długość 53 bajtów. ATM pozwala na skuteczne zarządzanie ruchem i zapewnia gwarantowaną jakość usług (QoS).

Sygnały

ATM używa cyfrowych sygnałów do przesyłania danych, z określoną prędkością transmisji, która w zależności od konfiguracji może wynosić od 155 Mbps do 10 Gbps.

E1/T1

E1 i T1 to standardy cyfrowych łączy telekomunikacyjnych wykorzystywanych w sieciach WAN.

Zastosowanie

E1/T1 są używane do łączenia sieci w różnych lokalizacjach, oferując transmisję danych z prędkościami odpowiednio 2 Mbps (E1) i 1.5 Mbps (T1).

Działanie urządzeń

Urządzenia E1/T1 przesyłają dane w postaci cyfrowych sygnałów przez specjalnie przygotowane łącza, umożliwiając komunikację pomiędzy różnymi punktami sieci.

Sygnały

E1/T1 używają cyfrowych sygnałów z określoną prędkością transmisji, która wynosi 1.544 Mbps w przypadku T1 i 2.048 Mbps w przypadku E1.

V.35

V.35 to standard interfejsu, który jest stosowany do przesyłania danych w sieciach WAN, w szczególności w połączeniach między urządzeniami routerów a urządzeniami sieciowymi.

Zastosowanie

V.35 jest stosowany w sieciach szerokopasmowych, gdzie wymagana jest duża prędkość przesyłania danych oraz niskie opóźnienia.

Działanie urządzeń

Urządzenia korzystające z V.35 używają cyfrowych sygnałów przesyłanych przez łącza szeregowe. Interfejs V.35 umożliwia przesyłanie danych z prędkościami od 64 kbps do 2 Mbps.

Sygnały

V.35 używa sygnałów elektrycznych o określonych napięciach i prędkościach transmisji, zapewniając wysoką jakość połączenia.

X.25

X.25 to jeden z pierwszych protokołów sieci pakietowej, opracowany w latach 70. XX wieku przez CCITT (obecnie ITU-T). Był używany głównie w publicznych sieciach transmisji danych przed popularyzacją protokołów TCP/IP i technologii Frame Relay.

Zasada działania X.25

X.25 jest protokołem warstwy sieciowej i łączy danych w modelu OSI. Jego architektura opiera się na trzech głównych warstwach:

- **Warstwa fizyczna** – zwykle korzystała z łączy szeregowych, takich jak EIA-232 (RS-232) lub V.35.
- **Warstwa łącza danych** – wykorzystywała protokół LAPB (Link Access Procedure, Balanced) będący odmianą HDLC do kontroli błędów i retransmisji.
- **Warstwa sieciowa** – obsługiwała wirtualne obwody pakietowe (*Virtual Circuits, VC*), co oznaczało, że każde połączenie wymagało zestawienia sesji przed transmisją danych.

Cechy i ograniczenia X.25

- **Potwierdzenia na każdym etapie** – każdy pakiet musiał być potwierdzony przez kolejne węzły, co zwiększało niezawodność, ale powodowało wysokie opóźnienia.
- **Przewidywana niezawodność** – wbudowane mechanizmy korekcji błędów umożliwiały działanie na niestabilnych łączach, takich jak łącza analogowe.
- **Niska przepustowość** – ograniczona do kilkudziesięciu kbps, co stało się niewystarczające w nowoczesnych zastosowaniach.

Zastosowania X.25

X.25 był powszechnie stosowany w latach 80. i 90. XX wieku w systemach wymagających stabilności i niezawodności:

- Bankowość i systemy płatności (np. terminale kart kredytowych w sieci VisaNet).
- Sieci rządowe i wojskowe, np. w administracji publicznej do transmisji danych.
- Połączenia terminalowe dla systemów mainframe, np. IBM 3270, umożliwiające zdalny dostęp do serwerów.

Następcy i obecny status

Protokół X.25 został stopniowo zastąpiony przez bardziej wydajne technologie, takie jak Frame Relay, ATM i MPLS. Obecnie jest rzadko używany, ale niektóre starsze systemy, zwłaszcza w bankowości i administracji publicznej, nadal mogą korzystać z jego wariantów.

Rodzaje topologii sieciowych

Topologia sieciowa to sposób, w jaki urządzenia w sieci są połączone ze sobą, oraz sposób, w jaki dane przepływają między nimi. Wybór odpowiedniej topologii zależy od wielu czynników, takich jak rozmiar sieci, wymagania dotyczące wydajności, niezawodności i kosztów. Poniżej przedstawiamy różne rodzaje topologii sieciowych wraz z ich zastosowaniami i przykładami.

Topologia magistrali

Topologia magistrali jest jedną z najstarszych topologii sieciowych. W tej topologii wszystkie urządzenia są podłączone do jednego wspólnego kabla (magistrali), przez który przesyłane są dane.

Zastosowanie

Topologia magistrali była szeroko stosowana w małych sieciach lokalnych, szczególnie w początkowych fazach rozwoju Ethernetu. Znalazła zastosowanie w sieciach, gdzie potrzebna była oszczędność miejsca i kosztów, np. w biurach lub lokalnych sieciach komputerowych.

Przykład

W tej topologii wszystkie urządzenia są połączone do jednej linii transmisyjnej, która działa jak kanał komunikacyjny dla całej sieci.

Topologia gwiazdy

Topologia gwiazdy jest jedną z najczęściej stosowanych w nowoczesnych sieciach komputerowych. W tej topologii wszystkie urządzenia są połączone z centralnym węzłem, zwanym koncentratorem lub switchem.

Zastosowanie

Topologia gwiazdy jest powszechnie stosowana w sieciach Ethernet, szczególnie w dużych organizacjach, gdzie centralny węzeł zapewnia łatwe zarządzanie ruchem w sieci i umożliwia łatwą diagnostykę.

Przykład

W tej topologii wszystkie urządzenia są połączone do centralnego punktu, jakim może być koncentrator, switch lub router.

Topologia pierścienia

W topologii pierścienia urządzenia są połączone w zamknięty pierścień, a dane podróżują w jednym kierunku (lub dwóch, jeśli jest to topologia podwójnego pierścienia).

Zastosowanie

Topologia pierścienia była popularna w starszych sieciach Token Ring i jest wykorzystywana w sieciach, które wymagają kontroli dostępu do medium transmisyjnego.

Przykład

Każde urządzenie w sieci jest połączone z dwoma sąsiednimi urządzeniami, tworząc zamknięty pierścień. Dane przesyłane są w jednym kierunku, a dostęp do medium uzyskuje się poprzez przechwycenie tokena.

Topologia drzewa

Topologia drzewa jest hierarchiczną kombinacją topologii gwiazdy i magistrali. W tej topologii urządzenia są podzielone na poziomy, a każdy poziom jest połączony z wyższym poziomem.

Zastosowanie

Topologia drzewa jest stosowana w dużych sieciach rozległych (WAN) i w sieciach, które wymagają dużej skalowalności i elastyczności, jak sieci w dużych firmach i kampusach.

Przykład

Sieć składa się z głównego węzła (root), który rozgałęzia się na inne węzły, tworząc strukturę przypominającą drzewo.

Topologia siatki

Topologia siatki zapewnia, że każde urządzenie w sieci jest połączone z każdym innym urządzeniem, tworząc pełną sieć połączeń.

Zastosowanie

Topologia siatki jest wykorzystywana w bardzo krytycznych systemach, które wymagają maksymalnej niezawodności i redundancji, jak w sieciach wojskowych, bankowych oraz w sieciach datacenter.

Przykład

Każde urządzenie w sieci jest połączone z innymi urządzeniami, co zapewnia alternatywne ścieżki w przypadku awarii połączenia.

Różnica między topologią fizyczną a logiczną

Topologia sieci komputerowej określa sposób organizacji urządzeń oraz metodę przesyłania danych. Można wyróżnić dwa podstawowe podejścia do klasyfikacji topologii: topologię fizyczną i logiczną. Chociaż często są one ze sobą powiązane, w rzeczywistości mogą znacząco się różnić.

Topologia fizyczna

Topologia fizyczna odnosi się do rzeczywistego układu połączeń między urządzeniami sieciowymi, w tym kabli, światłowodów i urządzeń sieciowych, takich jak przełączniki i routery. Jest to struktura materialna, która określa, jak urządzenia są rozmieszczone i połączone.

Przykłady topologii fizycznych:

- **Sieć Ethernet w biurze** – często oparta na fizycznej gwieździe, gdzie wszystkie urządzenia są podłączone do centralnego przełącznika (switcha).
- **Internet** – fizycznie składa się z rozległej sieci połączonych ze sobą węzłów (routerów), które komunikują się poprzez kable światłowodowe, łącza satelitarne i infrastrukturę naziemną. Struktura ta przypomina rozszerzoną gwiazdę.

Topologia logiczna

Topologia logiczna opisuje sposób, w jaki dane są przesyłane w sieci, niezależnie od fizycznych połączeń między urządzeniami. To abstrakcyjny model określający zasady routingu i transmisji pakietów w sieci.

Przykłady topologii logicznych:

- **Internet** – logicznie działa jako sieć w pełni połączona (*fully connected*), ponieważ każda maszyna może komunikować się z każdą inną poprzez system routerów i trasowania pakietów.
- **Sieć Ethernet w biurze** – chociaż fizycznie może być gwiazdą, logicznie często działa jako magistrala (broadcast), ponieważ każde urządzenie może odbierać pakiety przesyłane do całej sieci.

Podsumowanie różnic

- Topologia fizyczna opisuje rzeczywiste połączenia między urządzeniami, natomiast topologia logiczna definiuje przepływ danych i sposób ich transmisji.
- W wielu przypadkach topologia logiczna i fizyczna mogą się różnić – np. sieć Ethernet fizycznie może być gwiazdą, ale logicznie działa jako magistrala.
- Topologia logiczna jest często kształtowana przez protokoły sieciowe, takie jak Ethernet, TCP/IP czy MPLS, niezależnie od fizycznej struktury połączeń.

Rodzaje sieci komputerowych

Sieci komputerowe można klasyfikować na podstawie ich zasięgu geograficznego oraz funkcji, jakie pełnią. Do głównych typów należą: sieci lokalne (LAN), sieci rozległe (WAN) oraz globalna sieć Internet.

Sieci LAN

Sieć LAN (ang. *Local Area Network*) to sieć komputerowa obejmująca niewielki obszar geograficzny, taki jak jedno pomieszczenie, budynek lub kampus. Charakteryzuje się wysoką przepustowością i niskimi opóźnieniami, co umożliwia szybkie przesyłanie danych między podłączonymi urządzeniami. Typowe technologie wykorzystywane w sieciach LAN to Ethernet oraz Wi-Fi .

Schemat prostej sieci LAN z dwoma komputerami podłączonymi do switcha, routera i Internetu

Sieci WAN

Sieć WAN (ang. *Wide Area Network*) to rozległa sieć komputerowa, która łączy ze sobą sieci lokalne (LAN) znajdujące się w różnych obszarach geograficznych, często oddalonych od siebie o setki lub tysiące kilometrów. Sieci WAN umożliwiają komunikację i wymianę danych między oddziałami firm, instytucjami czy użytkownikami indywidualnymi na dużych odległościach. Przykładem sieci WAN jest Internet .

Schemat sieci WAN łączącej dwie sieci LAN za pośrednictwem routerów i Internetu

Internet

Internet to globalna sieć komputerowa, która łączy miliardy urządzeń na całym świecie, umożliwiając wymianę informacji i dostęp do różnorodnych usług, takich jak strony WWW, poczta elektroniczna czy media strumieniowe. Internet powstał z połączenia wielu sieci WAN i LAN, tworząc jednolitą strukturę komunikacyjną. Jego rozwój rozpoczął się w latach 60. XX wieku od projektu ARPANET, a obecnie stanowi nieodłączny element współczesnego życia społecznego i gospodarczego .

Hierarchiczny schemat połączeń sieci LAN, MAN, WAN prowadzących do Internetu

Ethernet

Historia powstania standardu

Ethernet został opracowany w latach 70. XX wieku przez Roberta Metcalfe’a i innych inżynierów w

firmie Xerox PARC. Pierwsza wersja standardu powstała w 1973 roku i była znana jako **Alto ALOHA Network**, która inspirowana była przez system komunikacji ALOHAnet używany na Hawajach . W 1980 roku firma Digital Equipment Corporation (DEC), Intel oraz Xerox opublikowały standard Ethernet o przepustowości 10 Mbps, który stał się podstawą późniejszych specyfikacji IEEE 802.3.

Definicja

Ethernet to technologia sieci lokalnych (LAN), która określa sposób przesyłania danych między urządzeniami przy użyciu metod dostępu do medium takich jak **CSMA/CD** (Carrier Sense Multiple Access with Collision Detection). Standard IEEE 802.3 opisuje różne warianty Ethernetu, począwszy od wersji 10 Mbps (Ethernet 10BASE-T) aż po nowoczesne wersje 400 Gbps.

Przyczyny zdefiniowania standardu Ethernet

Przed wprowadzeniem Ethernetu istniało wiele niekompatybilnych technologii sieciowych, takich jak Token Ring czy FDDI. Standaryzacja Ethernetu pozwoliła na:

- zapewnienie kompatybilności między różnymi producentami sprzętu,
- redukcję kosztów infrastruktury sieciowej,
- uproszczenie konfiguracji i obsługi sieci LAN,
- zwiększenie niezawodności i skalowalności komunikacji w sieciach komputerowych.

Organizacja danych w ramce Ethernet

Każda ramka Ethernet składa się z kilku pól, które pełnią określone funkcje. Poniżej przedstawiono przykładową ramkę Ethernet w formacie szesnastkowym, zgodną ze specyfikacją IEEE 802.3:

55 55 55 55 55 55 55 D5 AA AA AA AA AA AA BB BB BB BB BB BB 08 00
45 00 00 3C 1C 46 40 00 40 06 A6 EC
C0 A8 00 68 C0 A8 00 01
00 50 04 D2 00 00 00 00 00 00 00 00 50 02 20 00 91 7C 00 00
C2 21 C3 3D

Pola ramki:

- **55 55 55 55 55 55 55 D5** – Preambuła (7 bajtów) + Start Frame Delimiter (SFD, 1 bajt) – sygnał synchronizujący.
- **AA AA AA AA AA AA** – Adres MAC docelowy (6 bajtów) – identyfikuje odbiorcę.
- **BB BB BB BB BB BB** – Adres MAC źródłowy (6 bajtów) – adres nadawcy ramki.
- **08 00** – Typ EtherType (2 bajty) – np. IPv4 (0x0800), IPv6 (0x86DD), ARP (0x0806).
- **...** – Nagłówek IP (20 bajtów) – zawiera adresy IP nadawcy i odbiorcy.
- **...** – Dane (46–1500 bajtów) – rzeczywista treść wiadomości (np. segment TCP).
- **C2 21 C3 3D** – Suma kontrolna FCS (4 bajty) – wykrywa błędy transmisji.

Wersja Ethernet II używa pola EtherType do oznaczenia rodzaju przenoszonych danych, natomiast wersja IEEE 802.3 może zamiast tego zawierać pole długości i używać LLC (Logical Link Control) do

określenia protokołu.

Wykorzystanie Ethernetu w praktyce

Ethernet jest najczęściej stosowany w sieciach lokalnych (LAN), zarówno w środowiskach domowych, jak i korporacyjnych. Dzięki technologii **Gigabit Ethernet** i **10 Gigabit Ethernet** stał się również kluczową technologią w centrach danych oraz sieciach operatorskich. Przykładowe zastosowania:

- Połączenia komputerów w sieci firmowej i domowej.
- Łączenie serwerów w centrach danych.
- Przesyłanie danych w sieciach światłowodowych (np. Ethernet over Fiber).
- Integracja z sieciami Wi-Fi w ramach technologii **Power over Ethernet (PoE)**.

Protokół IPv4

Historia powstania standardu

Protokół IPv4 (Internet Protocol version 4) został opracowany w latach 70. XX wieku przez zespół inżynierów pod przewodnictwem Vinta Cerfa w ramach amerykańskiego projektu ARPANET. Protokół ten był częścią podstawowego zestawu technologii, które umożliwiły rozwój sieci komputerowych w Internecie. IPv4 stał się fundamentem dla globalnej sieci, a jego specyfikacja została opublikowana w 1981 roku w dokumencie RFC 791. Protokół IPv4 stał się powszechnie stosowanym rozwiązaniem w sieciach komputerowych na całym świecie.

Definicja

Protokół IPv4 jest protokołem warstwy sieciowej w modelu OSI, który umożliwia adresowanie i routowanie pakietów danych w sieci komputerowej. Jest jednym z kluczowych protokołów w Internecie, który zapewnia przekazywanie danych pomiędzy urządzeniami poprzez unikalne adresy IP. IPv4 używa 32-bitowych adresów IP, co pozwala na zaadresowanie około 4,3 miliarda unikalnych urządzeń.

Przyczyny zdefiniowania standardu IPv4

Przed wprowadzeniem protokołu IPv4 istniało wiele różnych, niekompatybilnych metod komunikacji w sieciach komputerowych. Standaryzacja IPv4 pozwoliła na:

- uproszczenie komunikacji między różnymi sieciami komputerowymi,
- zapewnienie globalnej kompatybilności urządzeń sieciowych,
- umożliwienie dynamicznego routowania i zarządzania ruchem w sieciach,
- wsparcie rozwoju Internetu, który szybko się rozrastał w latach 80. i 90.

Rodzaje adresów IPv4

IPv4 wykorzystuje różne klasy adresów oraz specjalne zakresy, które określają ich przeznaczenie. Poniżej przedstawiono główne kategorie:

Klasy adresów (podział historyczny):

- **Klasa A:**

Adresy od **0.0.0.0** do **127.255.255.255** (dla dużych sieci).
Uwaga: Zakres **127.0.0.0/8** jest zarezerwowany dla *loopback* (testowanie lokalne).

- **Klasa B:**

Adresy od **128.0.0.0** do **191.255.255.255** (dla sieci średniej wielkości).

- **Klasa C:**

Adresy od **192.0.0.0** do **223.255.255.255** (dla małych sieci).

- **Klasa D (Multicast):**

Adresy od **224.0.0.0** do **239.255.255.255** przeznaczone do transmisji multicast.

- **Klasa E (Experimental):**

Adresy od **240.0.0.0** do **255.255.255.254** wykorzystywane do celów eksperymentalnych i badawczych.

Adresy specjalne i zakresy prywatne:

- **Adres rozgłoszeniowy:**

255.255.255.255 – adres wykorzystywany do rozgłaszania pakietów w sieci lokalnej.

- **Adresy prywatne:**

Zakresy wykorzystywane w sieciach wewnętrznych (nie routowane w Internecie):

- **10.0.0.0/8** – adresy od 10.0.0.0 do 10.255.255.255.
- **172.16.0.0/12** – adresy od 172.16.0.0 do 172.31.255.255.
- **192.168.0.0/16** – adresy od 192.168.0.0 do 192.168.255.255.
- **Adres nieokreślony:**

0.0.0.0 – stosowany, gdy adres źródłowy nie jest jeszcze ustalony.

- **Adresy APIPA:**

Zakres **169.254.0.0/16** – automatyczna konfiguracja adresu (tzw. link-local) używana, gdy nie ma dostępnego serwera DHCP.

Struktura nagłówka IPv4

Nagłówek protokołu IPv4 zawiera informacje niezbędne do prawidłowego dostarczenia pakietu do celu. Poniżej przedstawiono strukturę nagłówka IPv4:

45 00 **00 3C** 1C 46
40 00 40 06 **A6 EC**
C0 A8 00 68 **C0 A8 00 01**
00 50 **04 D2** 00 00

Pola nagłówka:

- **45** – Wersja + IHL (Internet Header Length) – wersja protokołu (4) oraz długość nagłówka w 32-bitowych słowach.
- 00 – Type of Service (TOS) – wskazuje priorytet oraz jakość usługi dla pakietu.
- **00 3C** – Długość całkowita (Total Length) – łącznie z nagłówkiem i danymi, w bajtach.
- 1C 46 – Identyfikator (Identification) – identyfikuje pakiet, aby umożliwić jego ponowne złożenie.
- **40 00** – Flagi + Fragment Offset – umożliwia fragmentację pakietu.
- 40 06 – Protokół – określa, jaki protokół jest używany w warstwie wyższej (np. TCP=6, UDP=17).
- **A6 EC** – Suma kontrolna nagłówka – wykrywanie błędów w nagłówku.
- C0 A8 00 68 – Adres źródłowy (Source Address) – adres IP nadawcy.
- **C0 A8 00 01** – Adres docelowy (Destination Address) – adres IP odbiorcy.
- 00 50 – Port źródłowy – stosowane w przypadku protokołów transportowych, takich jak TCP i UDP.
- **04 D2** – Port docelowy – port, na który pakiet jest kierowany.
- 00 00 – Dane (Payload) – zawiera właściwą treść wiadomości.

Wykorzystanie IPv4 w praktyce

IPv4 stanowi fundament działania Internetu, a protokół jest szeroko stosowany w wielu scenariuszach:

- Adresowanie urządzeń w sieciach lokalnych (LAN) oraz globalnych (Internet).
- Rutowanie pakietów między różnymi sieciami.
- Zapewnienie rozwoju usług Internetowych, takich jak WWW, e-mail czy streaming.
- Używanie NAT (Network Address Translation) w celu zarządzania adresami IP w sieciach prywatnych.
- Obsługa dynamicznego przydzielania adresów IP poprzez DHCP (Dynamic Host Configuration Protocol).

Protokół IPv4, mimo swojej powszechności, ma ograniczoną liczbę dostępnych adresów, co skłoniło do rozwoju protokołu IPv6, który rozwiązuje ten problem. Jednak IPv4 nadal jest dominującym protokołem w wielu sieciach na całym świecie.

Podział sieci na podsieci w sieciach IPv4

Metoda klasowa:
Adresy IPv4 dzielone są na klasy, z których najpopularniejsze to:

- **Klasa A**: adresy od 0.0.0.0 do 127.255.255.255, domyślna maska podsieci **255.0.0.0** (lub /8).
- **Klasa B**: adresy od 128.0.0.0 do 191.255.255.255, domyślna maska podsieci **255.255.0.0** (lub /16).
- **Klasa C**: adresy od 192.0.0.0 do 223.255.255.255, domyślna maska podsieci **255.255.255.0** (lub /24).

W tej metodzie sieci są dzielone według stałych masek, co ogranicza elastyczność przy przydziale adresów. Na przykład sieć klasy C (/24) może być podzielona tylko na podsieci o stałej wielkości, co w przypadku potrzeby mniejszych lub większych segmentów może prowadzić do marnotrawstwa adresów.

Metoda VLSM (Variable Length Subnet Mask):
VLSM pozwala na stosowanie masek o zmiennej długości, dzięki czemu możliwe jest przydzielanie dokładnie tyle adresów, ile potrzeba dla danej podsieci. Dzięki temu optymalnie wykorzystuje się dostępną przestrzeń adresową. Przykłady masek i odpowiadających im liczby adresów:

- Maska **255.255.255.224** (/27) – 32 adresy, z czego 30 dostępnych dla hostów (po odliczeniu adresu sieci i rozgłoszeniowego).
- Maska **255.255.255.192** (/26) – 64 adresy, z czego 62 dostępne dla hostów.
- Maska **255.255.255.128** (/25) – 128 adresów, z czego 126 dostępnych dla hostów.

Dzięki VLSM administrator może podzielić jedną dużą sieć na podsieci o różnych rozmiarach, idealnie dopasowując je do rzeczywistych potrzeb, co zmniejsza marnotrawstwo adresów.

Podział sieci 10.42.0.0/24 na 4 równe podsieci

Dane wejściowe:

- Adres sieci: 10.42.0.0/24 (256 adresów)
- Liczba podsieci: 4

Aby podzielić sieć na 4 równe części, potrzebujemy 2 dodatkowych bitów (ponieważ $2^2 = 4$).
Nowa maska podsieci:
$24 + 2 = 26$, czyli /26.
Liczba adresów w każdej podsieci:
$2^{32-26} = 2^6 = 64$ adresów (62 hosty, gdyż 2 adresy są zarezerwowane – sieciowy i rozgłoszeniowy).

Przydział podsieci:

- Podsieć 1: **10.42.0.0/26**

Zakres: 10.42.0.0 – 10.42.0.63

- Podsieć 2: **10.42.0.64/26**

Zakres: 10.42.0.64 – 10.42.0.127

- Podsieć 3: **10.42.0.128/26**

Zakres: 10.42.0.128 – 10.42.0.191

- Podsieć 4: **10.42.0.192/26**

Zakres: 10.42.0.192 – 10.42.0.255

Podział sieci 10.42.0.0/16 metodą VLSM

Wymagania dla poszczególnych podsieci:

- A: 1000 hostów
- B: 200 hostów
- C: 200 hostów
- D: 100 hostów
- E: 100 hostów
- F: 50 hostów
- G: 50 hostów
- H: 10 hostów
- I: 2 hosty
- J: 2 hosty

Krok 1: Sortowanie wg wymagań
Kolejność przydziału (od największej do najmniejszej): A, B, C, D, E, F, G, H, I, J.

Krok 2: Wyznaczenie minimalnej maski dla każdej podsieci:

- A (1000 hostów):

Wymagane adresy $\geq 1000 + 2 = 1002$.
Najmniejsza potęga dwójki: $2^{10} = 1024$ adresów $\Rightarrow$ maska: $32 - 10 =$ **/22**.

- B (200 hostów):

Wymagane adresy ≥ 202.
$2^8 = 256$ adresów (254 użyteczne) $\Rightarrow$ maska: **/24**.

- C (200 hostów): **/24**.
- D (100 hostów):

Wymagane adresy ≥ 102.
$2^7 = 128$ adresów (126 użytecznych) $\Rightarrow$ maska: **/25**.

- E (100 hostów): **/25**.
- F (50 hostów):

Wymagane adresy ≥ 52.

$2^6 = 64$ adresów (62 użytecznych) $\Rightarrow$ maska: **/26**.

- G (50 hostów): **/26**.
- H (10 hostów):

Wymagane adresy ≥ 12.
$2^4 = 16$ adresów (14 użytecznych) $\Rightarrow$ maska: **/28**.

- I (2 hosty):

Wymagane adresy ≥ 4.
$2^2 = 4$ adresy (2 użyteczne) $\Rightarrow$ maska: **/30**.

- J (2 hosty): **/30**.

Krok 3: Przydział podsieci w obrębie 10.42.0.0/16
Przydział kolejnych podsieci (każda zaczyna się od pierwszego dostępnego adresu):

- A: **10.42.0.0/22**

Zakres: 10.42.0.0 – 10.42.3.255

- B: **10.42.4.0/24**

Zakres: 10.42.4.0 – 10.42.4.255

- C: **10.42.5.0/24**

Zakres: 10.42.5.0 – 10.42.5.255

- D: **10.42.6.0/25**

Zakres: 10.42.6.0 – 10.42.6.127

- E: **10.42.6.128/25**

Zakres: 10.42.6.128 – 10.42.6.255

- F: **10.42.7.0/26**

Zakres: 10.42.7.0 – 10.42.7.63

- G: **10.42.7.64/26**

Zakres: 10.42.7.64 – 10.42.7.127

- H: **10.42.7.128/28**

Zakres: 10.42.7.128 – 10.42.7.143

- I: **10.42.7.144/30**

Zakres: 10.42.7.144 – 10.42.7.147

- J: **10.42.7.148/30**

Zakres: 10.42.7.148 – 10.42.7.151

Podsumowanie:
Podsieci przydzielone zostały kolejno od adresu 10.42.0.0 do 10.42.7.151, co mieści się w ramach sieci 10.42.0.0/16.

Protokół IPv6

Historia powstania standardu

Protokół IPv6 (Internet Protocol version 6) został opracowany w latach 90. XX wieku przez IETF (Internet Engineering Task Force) w odpowiedzi na wyczerpywanie się dostępnych adresów IPv4. IPv6 został zaprezentowany w dokumencie RFC 2460 w grudniu 1998 roku. Głównym celem IPv6 było zapewnienie ogromnej przestrzeni adresowej, a także wprowadzenie usprawnień w zakresie bezpieczeństwa, mobilności i jakości usług. Chociaż IPv6 został zaprojektowany jako długoterminowa alternatywa dla IPv4, jego implementacja i adopcja są nadal w trakcie realizacji.

Definicja

Protokół IPv6 jest protokołem warstwy sieciowej, który umożliwia adresowanie i przesyłanie pakietów danych w globalnej sieci komputerowej, w tym w Internecie. IPv6 używa 128-bitowych adresów, co pozwala na adresowanie około 340 undecylionów unikalnych urządzeń. Oprócz tego, IPv6 wprowadza liczne usprawnienia, takie jak uproszczony nagłówek, lepsze wsparcie dla mobilności i szyfrowania danych, a także automatyczną konfigurację adresów.

Przyczyny zdefiniowania standardu IPv6

Protokół IPv6 został zaprojektowany z kilku powodów:

- Rozszerzenie przestrzeni adresowej – IPv4, z 32-bitowym adresem, miał ograniczoną liczbę dostępnych adresów. IPv6, z 128-bitowym adresem, umożliwia przypisanie unikalnego adresu praktycznie każdemu urządzeniu na świecie.
- Lepsza efektywność routingu – nagłówek IPv6 jest bardziej uproszczony w porównaniu do IPv4, co pozwala na bardziej wydajne i szybsze routowanie pakietów.
- Wsparcie dla bezpieczeństwa – w IPv6 domyślnie włączona jest obsługa IPSec, co zapewnia lepsze zabezpieczenie komunikacji.
- Wsparcie dla mobilności – IPv6 lepiej wspiera mobilność urządzeń, umożliwiając płynne przełączanie między sieciami.
- Uproszczenie konfiguracji sieci – IPv6 oferuje automatyczną konfigurację adresów (SLAAC), co upraszcza proces przydzielania adresów w sieciach.

Rodzaje adresów IPv6

Adresy IPv6 mają długość 128 bitów, co daje ogromną przestrzeń adresową. Wśród nich wyróżniamy następujące kategorie:

1. Adresy Unicast:

- **Global Unicast Addresses:**

Są to adresy publiczne routowane w Internecie. Większość adresów global unicast znajduje się w zakresie **2000::/3** (adresy zaczynają się od binarnego wzoru **001**), co czyni je analogicznymi do publicznych adresów IPv4.

- **Link-Local Addresses:**

Adresy te służą do komunikacji w obrębie jednego segmentu sieci (np. LAN) i posiadają prefiks **FE80::/10**. Są automatycznie konfigurowane na interfejsach i nie są routowane poza lokalnym łączem.

- **Unique Local Addresses (ULA):**

Przeznaczone do użytku w sieciach prywatnych, analogicznie do prywatnych adresów IPv4. Zakres ULA to **FC00::/7**, z częściej stosowanym podzakresem **FD00::/8**.

2. Adresy Multicast:

- **Multicast Addresses:**

Adresy multicast służą do przesyłania pakietów do grupy odbiorców. W IPv6 adresy multicast zaczynają się od **FF00::/8**.

3. Adresy Specjalne:

- **Adres nieokreślony (Unspecified Address):**

:: – reprezentuje brak przypisanego adresu, analogiczny do IPv4 **0.0.0.0**.

- **Adres loopback:**

::1 – używany do komunikacji wewnętrznej na danym urządzeniu (testowanie lokalne).

Zasady skróconego zapisu IPv6

Adresy IPv6 składają się z 8 hextetów (16-bitowych bloków) zapisywanych w systemie szesnastkowym i oddzielonych dwukropkami. Aby ułatwić zapis i odczyt, stosuje się następujące zasady skracania:

1. Usuwanie wiodących zer:
W każdej grupie można pominąć wiodące zera. Na przykład:

- `0db8` $\rightarrow$ `db8`
- `0001` $\rightarrow$ `1`

- `0000` $\rightarrow$ `0`

2. Zastępowanie ciągów zer podwójnym dwukropkiem (::):
Można jednorazowo zastąpić jedną lub więcej kolejnych grup zawierających same zera symbolem `::`. Na przykład:

- `2001:0db8:0000:0000:0000:ff00:0042:8329` $\rightarrow$ `2001:db8::ff00:42:8329`

Uwaga: Podwójny dwukropek może wystąpić tylko raz w jednym adresie, aby zachować jednoznaczność zapisu.

Przykłady skracania:

- `2001:0db8:0000:0001:0000:0000:0000:0001` $\rightarrow$ `2001:db8:0:1::1`
- `0000:0000:0000:0000:0000:0000:0000:0001` $\rightarrow$ `::1`
- `0000:0000:0000:0000:0000:0000:0000:0000` $\rightarrow$ `::`

Struktura nagłówka IPv6

Nagłówek protokołu IPv6 jest znacznie uproszczony w porównaniu do nagłówka IPv4. Zawiera tylko najważniejsze informacje niezbędne do przesyłania pakietu przez sieć. Poniżej przedstawiono przykładową strukturę nagłówka IPv6:

6 00 **00014**
1140 **06** 20
2001 0DB8 85A3 0000 8A2E 0370 7334
2001 0DB8 85A3 0000 8A2E 0370 7335

Pola nagłówka:

- **6** – **Wersja (Version)** – 4 bity, IPv6 = 6.
- 00 – **Traffic Class** – 8 bitów, wskazuje jakość usług.
- **00014** – **Flow Label** – 20 bitów, identyfikuje przepływ danych (w przykładzie wartość 0x00014).
- 1140 – **Payload Length** – 16 bitów, długość danych (bez nagłówka) w bajtach.
- **06** – **Next Header** – 8 bitów, wskazuje protokół warstwy wyższej (np. TCP, tu 0x06).
- 20 – **Hop Limit** – 8 bitów, maksymalna liczba przeskoków.
- **2001 0DB8 85A3 0000 8A2E 0370 7334** – **Adres źródłowy (Source Address)** – 128 bitów.
- 2001 0DB8 85A3 0000 8A2E 0370 7335 – **Adres docelowy (Destination Address)** – 128 bitów.

Wykorzystanie IPv6 w praktyce

Protokół IPv6, mimo że jego adopcja trwa, jest coraz bardziej wykorzystywany w różnych dziedzinach:

- Adresowanie urządzeń w nowoczesnych sieciach – IPv6 jest używany w nowych wdrożeniach sieci, zwłaszcza tam, gdzie IPv4 już nie wystarcza.
- IoT (Internet of Things) – ogromna przestrzeń adresowa IPv6 jest idealna do adresowania miliardów urządzeń IoT.

- Mobilność i zmiana sieci – IPv6 wspiera mobilność urządzeń, co jest szczególnie istotne w przypadku urządzeń przenośnych.
- Bezpieczeństwo i szyfrowanie – dzięki wbudowanemu wsparciu dla IPSec, IPv6 zapewnia lepsze bezpieczeństwo w komunikacji.
- Zwiększona efektywność routingu – uproszczony nagłówek IPv6 pozwala na szybsze przekazywanie pakietów przez routery.

Mimo że IPv4 nadal dominuje w Internecie, to IPv6 jest kluczowym elementem przyszłościowych sieci, zwłaszcza w kontekście rosnącej liczby urządzeń podłączonych do sieci.

Podział na podsieci w sieciach IPv6

IPv6 to protokół wykorzystujący 128-bitowe adresy, co daje ogromną przestrzeń adresową. Adres IPv6 dzieli się na kilka logicznych części, co umożliwia hierarchiczny podział sieci. Główne elementy to:

- **Globalny prefiks routingu (Global Routing Prefix)** – przydzielany przez dostawcę usług internetowych (ISP). Określa część adresu wspólną dla całej organizacji, np. 2001:0db8:1234 w prefiksie 2001:0db8:1234::/48.
- **Identyfikator podsieci (Subnet ID)** – wykorzystywany przez organizację do dzielenia przydzielonego bloku na mniejsze podsieci. Zazwyczaj wykorzystuje się 16 bitów, co przy typowym przydziale /48 daje podsieci z maską /64.
- **Identyfikator interfejsu (Interface Identifier)** – ostatnie 64 bity adresu, które identyfikują unikalny interfejs w danej podsieci.

Przykład:
Jeśli organizacji przydzielono prefiks 2001:0db8:1234::/48, to:

- **Globalny prefiks routingu:** 2001:0db8:1234
- **Identyfikator podsieci:** Można wykorzystać kolejne 16 bitów, co pozwala na utworzenie $2^{16} = 65\,536$ podsieci. Typowo wybieramy podsieci z maską /64, więc adres podsieci może mieć postać 2001:0db8:1234:xxxx::/64, gdzie xxxx to 16-bitowy identyfikator podsieci.
- **Identyfikator interfejsu:** Pozostałe 64 bity, służące do unikalnej identyfikacji interfejsu w obrębie danej podsieci.

Dzięki takiemu podziałowi, sieć IPv6 umożliwia:

- Hierarchiczne zarządzanie dużymi blokami adresowymi.
- Łatwe tworzenie wielu podsieci (do 65 536 w przykładzie z prefiksem /48).
- Automatyczną konfigurację adresów dzięki standardowemu podziałowi /64 (często stosowanemu w protokole SLAAC).

W rezultacie, podział na podsieci w IPv6 jest bardziej elastyczny i upraszcza zarządzanie adresacją w porównaniu do IPv4.

Model warstwowy ISO/OSI

Warstwa 1 - Fizyczna (ang. Physical Layer)

Zasada działania tej warstwy

Warstwa fizyczna w modelu OSI odpowiada za przesyłanie surowych danych w postaci bitów przez medium transmisyjne, takie jak kable, fale radiowe lub światłowody. Zajmuje się ustaleniem sposobu kodowania bitów na sygnały elektryczne, optyczne lub radiowe, a także definiuje parametry transmisji, takie jak napięcie, częstotliwość, oraz długość fali. Ta warstwa nie zajmuje się interpretowaniem danych, tylko zapewnia fizyczny transfer informacji pomiędzy urządzeniami.

Zasadnicze zadanie warstwy fizycznej to:

- Konwersja danych do postaci bitów i przesyłanie ich przez medium,
- Ustalanie właściwego typu medium transmisyjnego (np. miedziane kable, światłowody, fale radiowe),
- Synchronizacja czasu pomiędzy nadawcą a odbiorcą,
- Określenie parametrów transmisji, takich jak prędkość transmisji.

Protokoły wykorzystane w tej warstwie

Warstwa fizyczna nie korzysta z protokołów w tradycyjnym sensie, jak inne warstwy modelu OSI. Zamiast tego obejmuje technologie i standardy, które definiują fizyczne aspekty transmisji danych. Do najważniejszych protokołów i technologii w tej warstwie należą:

- **Ethernet (10BASE-T, 100BASE-TX, 1000BASE-T)** – definiuje zasady przesyłania bitów przez medium miedziane (kabel miedziany twisted pair).
- **Wi-Fi (IEEE 802.11)** – wykorzystuje fale radiowe do przesyłania danych w sieci bezprzewodowej.
- **Bluetooth** – stosowany do transmisji na krótkich odległościach, używa fal radiowych w paśmie ISM (2,4 GHz).
- **Gigabit Ethernet (1000BASE-T)** – rozszerza Ethernet na prędkość 1 Gbps, wykorzystując kable miedziane.
- **Fiber Optic (IEEE 802.3z, 10GBASE-SR)** – przesyła dane za pomocą światłowodów, co pozwala na bardzo szybkie i dalekozasięgowe transmisje.
- **DSL (Digital Subscriber Line)** – technologia wykorzystująca linie telefoniczne do transmisji danych z dużą prędkością.

Słowa kluczowe do zapamiętania

- **Bit** – najmniejsza jednostka informacji w warstwie fizycznej, reprezentująca stan 0 lub 1.
- **Medium transmisyjne** – nośnik, przez który przesyłane są bity, np. kabel miedziany, światłowód, fale radiowe.
- **Transmisja szeregowa** – sposób przesyłania danych, w którym bity są wysyłane jeden po drugim.

- **Modulacja** – proces zmiany właściwości sygnału w celu reprezentacji danych.
- **Częstotliwość** – liczba cykli sygnału na jednostkę czasu, ważna przy określaniu pasma transmisji.
- **Prędkość transmisji** – szybkość, z jaką dane są przesyłane, mierzona w bitach na sekundę (bps).
- **Kodowanie sygnału** – technika, która zamienia dane na postać zrozumiałą dla medium transmisyjnego, np. Manchester Encoding.

Przykład pakietu Ethernet na warstwie fizycznej

Ethernet w warstwie fizycznej jest odpowiedzialny za kodowanie bitów na sygnały elektryczne i ich przesyłanie przez medium transmisyjne. Poniżej przedstawiony jest uproszczony schemat działania Ethernetu, który ilustruje, jak bity z warstwy łącza danych są kodowane na sygnałach elektrycznych w standardzie 100BASE-TX.

W tym przypadku, bity, które zostały utworzone w warstwie łącza danych (np. w ramce Ethernet), są kodowane na sygnały elektryczne i przesyłane przez medium (np. kabel miedziany lub światłowód).

Warstwa 2 - Łącza danych (ang. Data Link Layer)

Zasada działania tej warstwy

Warstwa łącza danych odpowiedzialna jest za przesyłanie danych między dwoma bezpośrednio połączonymi urządzeniami. Zajmuje się organizowaniem bitów w ramki, które zawierają dane oraz kontrolę błędów. Przesyłanie danych odbywa się w postaci ramek, które są adresowane za pomocą adresów MAC (Media Access Control). Celem tej warstwy jest zapewnienie niezawodnego transferu danych pomiędzy urządzeniami w tej samej sieci fizycznej (np. w obrębie jednej lokalnej sieci LAN).

Zasadnicze zadania warstwy łącza danych to:

- Podział danych na ramki oraz dodawanie nagłówków i stref kontrolnych,
- Zapewnienie detekcji błędów w przesyłanych danych (np. suma kontrolna CRC),
- Sterowanie dostępem do medium (np. w Ethernet za pomocą CSMA/CD),
- Mapowanie adresów IP na adresy MAC (adresowanie fizyczne),
- Fragmentacja i de-fragmentacja danych.

Warstwa ta ma dwie główne funkcje:

- **Kontrola dostępu do medium (MAC)** – odpowiada za sposób, w jaki urządzenia w sieci uzyskują dostęp do wspólnego medium transmisyjnego.
- **Kontrola błędów (LLC)** – odpowiada za detekcję i korekcję błędów, które mogą wystąpić podczas przesyłania danych.

Protokoły wykorzystane w tej warstwie

Warstwa łącza danych wykorzystuje różne protokoły, które zarządzają dostępem do medium oraz

zapewniają niezawodność transmisji. Do najważniejszych protokołów i standardów w tej warstwie należą:

- **Ethernet (IEEE 802.3)** – protokół wykorzystywany w sieciach lokalnych (LAN), który odpowiada za tworzenie ramek z danymi i zarządzanie dostępem do medium (dzięki mechanizmowi CSMA/CD).
- **Wi-Fi (IEEE 802.11)** – standard komunikacji bezprzewodowej, który operuje na tej samej warstwie w sieciach WLAN.
- **PPP (Point-to-Point Protocol)** – wykorzystywany w połączeniach punkt-punkt, np. w dial-up oraz VPN, umożliwia autentykację i kompresję.
- **HDLC (High-Level Data Link Control)** – protokół ramkowy, który zapewnia bezbłędny transfer danych w komunikacji punkt-punkt.
- **Frame Relay** – protokół wykorzystywany w sieciach rozległych (WAN), który zapewnia efektywną transmisję danych w postaci ramek.
- **ATM (Asynchronous Transfer Mode)** – stosowany do transmisji danych o różnej prędkości (dźwięk, obraz, dane) w sieciach telekomunikacyjnych.

Słowa kluczowe do zapamiętania

- **Ramka** – podstawowa jednostka danych przesyłana w warstwie łącza danych, zawiera adresy MAC, dane oraz sumy kontrolne.
- **Adres MAC** – unikalny adres przypisany każdemu urządzeniu sieciowemu na poziomie sprzętowym, wykorzystywany do kierowania ramkami w sieci.
- **Detekcja błędów** – proces wykrywania błędów w przesyłanych danych (np. sumy kontrolne CRC).
- **CSMA/CD** – metoda dostępu do medium w sieciach Ethernet, która reguluje, kiedy urządzenie może wysyłać dane (Carrier Sense Multiple Access with Collision Detection).
- **MAC address resolution** – proces mapowania adresu IP na adres MAC za pomocą protokołu ARP (Address Resolution Protocol).
- **Flow control** – kontrolowanie przepływu danych w celu zapobiegania ich utracie lub przepełnieniu bufora.
- **Fragmentacja** – podział dużych bloków danych na mniejsze fragmenty, aby mogły być przesyłane przez medium transmisyjne.
- **LLC** – Logical Link Control, podwarstwa odpowiedzialna za zarządzanie ramkami i błędami.

Przykład ramki Ethernet w warstwie 2

Warstwa łącza danych w standardzie Ethernet (IEEE 802.3) tworzy ramki, które są przesyłane przez warstwę fizyczną. Poniżej przedstawiono strukturę ramki Ethernet, która zawiera zarówno adresy MAC, jak i inne informacje niezbędne do poprawnego przesyłania danych.

Na schemacie przedstawiona jest struktura ramki Ethernet, z podziałem na poszczególne pola:

- **Adres MAC docelowy** – 6 bajtów adresu docelowego urządzenia.
- **Adres MAC źródłowy** – 6 bajtów adresu urządzenia nadawcy.
- **EtherType** – 2 bajty wskazujące typ protokołu (np. IPv4, ARP).
- **Dane** – dane, które są przekazywane w ramce (od 46 do 1500 bajtów).
- **FCS** – suma kontrolna (Frame Check Sequence), która zapewnia wykrywanie błędów w ramce.

Warstwa 3 - Sieci (ang. Network Layer)

Zasada działania tej warstwy

Warstwa sieciowa w modelu OSI odpowiada za przesyłanie danych pomiędzy różnymi sieciami oraz ich routowanie. Jej głównym zadaniem jest zapewnienie niezawodnej komunikacji między urządzeniami, które nie muszą być bezpośrednio połączone w tej samej sieci fizycznej. Warstwa ta odpowiada za przekazywanie danych w postaci pakietów, które mogą przemieszczać się przez różne urządzenia sieciowe, takie jak routery.

Zasadnicze zadania warstwy sieciowej:

- **Adresowanie logiczne** – przypisanie adresów IP, które umożliwiają urządzeniom w sieci komunikowanie się, niezależnie od fizycznej lokalizacji.
- **Routing** – wybór najlepszej ścieżki dla przesyłanych danych w celu dotarcia do celu, przy użyciu różnych algorytmów routingu (np. OSPF, BGP).
- **Fragmentacja i de-fragmentacja** – podział dużych pakietów na mniejsze jednostki, które mogą być przesyłane przez medium o mniejszej maksymalnej wielkości.
- **Przesyłanie pakietów** – odpowiedzialność za adresowanie i przekazywanie pakietów przez urządzenia sieciowe, takie jak routery, pomiędzy różnymi segmentami sieci.

W tej warstwie dane są przekazywane w formie pakietów, które zawierają nie tylko dane, ale również informacje niezbędne do ich poprawnego przesyłania, takie jak adresy źródłowe i docelowe (np. adresy IP).

Protokoły wykorzystane w tej warstwie

Warstwa sieciowa korzysta z protokołów odpowiedzialnych za adresowanie, routing oraz przesyłanie pakietów przez różne sieci. Do najważniejszych protokołów w tej warstwie należą:

- **IP (Internet Protocol)** – podstawowy protokół warstwy sieciowej, odpowiedzialny za adresowanie logiczne oraz przesyłanie pakietów przez sieć. Istnieją dwie wersje protokołu: IPv4 oraz IPv6.
- **ICMP (Internet Control Message Protocol)** – protokół pomocniczy, który umożliwia wymianę komunikatów kontrolnych w sieci, takich jak informacje o błędach (np. komunikaty „host unreachable”).
- **ARP (Address Resolution Protocol)** – protokół wykorzystywany do mapowania adresów IP na adresy MAC w sieci lokalnej (LAN).
- **RARP (Reverse ARP)** – protokół, który umożliwia uzyskanie adresu IP na podstawie znanego adresu MAC.
- **Routing Protocols (RIP, OSPF, BGP)** – protokoły odpowiedzialne za wybór najlepszej ścieżki do przesyłania pakietów w obrębie sieci oraz między różnymi sieciami.

Słowa kluczowe do zapamiętania

- **Adres IP** – unikalny adres logiczny przypisany urządzeniom w sieci, wykorzystywany do ich identyfikacji na poziomie warstwy sieciowej.
- **Routowanie** – proces wyznaczania najlepszej trasy dla pakietów przesyłanych przez sieć, który

jest realizowany przez routery.
- **Router** – urządzenie sieciowe odpowiedzialne za przesyłanie pakietów pomiędzy różnymi sieciami, na podstawie adresów IP.
- **Pakiet** – jednostka danych, która jest przesyłana przez warstwę sieciową i zawiera dane oraz adresy źródłowe i docelowe.
- **Fragmentacja** – proces dzielenia dużych pakietów na mniejsze, aby mogły być przesyłane przez sieci z ograniczeniami na wielkość pakietu (MTU).
- **ICMP** – protokół kontrolny, który służy do przesyłania komunikatów o stanie sieci, takich jak błędy czy potwierdzenia.
- **Routing table** – tablica routingu, która zawiera informacje o dostępnych trasach do różnych sieci.

Przykład pakietu IP w warstwie 3

Pakiet IP jest podstawową jednostką przesyłania danych w warstwie sieciowej. Zawiera nie tylko dane użytkownika, ale również informacje potrzebne do przesłania pakietu przez sieć, takie jak adresy źródłowy i docelowy. Poniżej przedstawiona jest struktura pakietu IPv4.

Powyższy schemat przedstawia strukturę pakietu IPv4, w którym:

- **Wersja** – identyfikuje wersję protokołu IP (IPv4 lub IPv6).
- **Typ usługi (TOS)** – określa preferencje dotyczące jakości usług, takie jak priorytet.
- **Długość pakietu** – całkowita długość pakietu (nagłówek + dane).
- **Adres źródłowy i docelowy** – 4-bajtowe adresy IP urządzeń.
- **TTL** – czas życia pakietu, który zapobiega jego nieskończonemu krążeniu w sieci.
- **Protokół** – informuje, jaki protokół wyższej warstwy (np. TCP, UDP) jest używany w pakiecie.

Warstwa 4 - Transportowa (ang. Transport Layer)

Zasada działania tej warstwy

Warstwa transportowa jest odpowiedzialna za zapewnienie niezawodnej komunikacji między urządzeniami w sieci, poprzez zarządzanie przesyłaniem danych pomiędzy różnymi procesami lub aplikacjami działającymi na tych urządzeniach. Jej głównym zadaniem jest podzielenie danych na segmenty (w przypadku TCP i UDP) oraz kontrola przepływu i niezawodności przesyłania danych.

Zasadnicze zadania warstwy transportowej:

- **Segmentacja i reassembly (składanie)** – podział danych z aplikacji na mniejsze jednostki (segmenty) oraz ich ponowne składanie po dotarciu do odbiorcy.
- **Kontrola przepływu** – zarządzanie tempem przesyłania danych, aby zapobiec przeciążeniu odbiorcy.
- **Kontrola błędów** – wykrywanie i naprawianie błędów w przesyłanych segmentach.
- **Zarządzanie połączeniami** – inicjowanie, utrzymywanie oraz zamykanie połączeń między urządzeniami (w przypadku protokołów połączeniowych).

Warstwa transportowa decyduje również, czy połączenie ma być połączeniowe (wymagające

ustanowienia sesji) czy bezpołączeniowe. Do realizacji tych zadań warstwa ta wykorzystuje różne protokoły, w tym TCP i UDP.

Protokoły wykorzystane w tej warstwie

Warstwa transportowa wykorzystuje protokoły odpowiedzialne za segmentację danych, zarządzanie połączeniami, kontrolę przepływu oraz kontrolę błędów. Do najważniejszych protokołów w tej warstwie należą:

- **TCP (Transmission Control Protocol)** – protokół połączeniowy, który zapewnia niezawodność komunikacji, poprzez segmentację danych, retransmisję utraconych pakietów oraz kontrolę błędów. TCP zapewnia pełne potwierdzenie odbioru i kontrolę przepływu.
- **UDP (User Datagram Protocol)** – protokół bezpołączeniowy, który przesyła dane w postaci datagramów bez gwarancji niezawodności. UDP jest szybszy niż TCP, ale nie zapewnia kontroli błędów ani retransmisji.
- **SCTP (Stream Control Transmission Protocol)** – protokół połączeniowy, który łączy cechy TCP i UDP, oferując niezawodność oraz możliwość obsługi wielu strumieni danych w ramach jednego połączenia.
- **DCCP (Datagram Congestion Control Protocol)** – protokół zaprojektowany do zapewnienia kontrolowania przeciążenia sieci, szczególnie przy aplikacjach typu strumieniowego, takich jak wideo i audio.

Słowa kluczowe do zapamiętania

- **Połączenie** – sesja komunikacyjna pomiędzy dwoma urządzeniami w sieci, którą należy ustanowić przed rozpoczęciem wymiany danych (dotyczy protokołów połączeniowych takich jak TCP).
- **Segmentacja** – proces dzielenia dużych danych na mniejsze segmenty, które mogą być łatwiej przesyłane przez sieć.
- **TCP** – protokół transportowy zapewniający niezawodność i kontrolę przepływu; wykorzystywany głównie w aplikacjach wymagających pełnej niezawodności (np. HTTP, FTP).
- **UDP** – protokół transportowy, który jest szybszy, ale nie zapewnia gwarancji dostarczenia danych, często wykorzystywany w aplikacjach wymagających niskich opóźnień (np. strumieniowanie wideo, VoIP).
- **Porty** – numery portów wykorzystywane przez protokoły transportowe, które umożliwiają identyfikację aplikacji działających na urządzeniu.
- **Kontrola błędów** – mechanizmy wykrywania i naprawiania błędów w transmisji, takie jak sumy kontrolne (checksums).
- **Retransmisja** – proces ponownego wysyłania danych, które nie dotarły do odbiorcy lub zostały uszkodzone.

Przykład segmentu TCP

Segment TCP jest podstawową jednostką danych w protokole TCP. Zawiera on zarówno dane użytkownika, jak i informacje kontrolne, takie jak numery portów, numery sekwencyjne, flagi kontrolne, sumy kontrolne i inne.

Powyższy schemat przedstawia strukturę segmentu TCP, w którym:

- **Porty źródłowy i docelowy** – identyfikują aplikacje komunikujące się w ramach połączenia.
- **Numer sekwencyjny** – identyfikuje porządek przesyłanych segmentów.
- **Numer potwierdzenia** – informuje o tym, który segment został poprawnie odebrany.
- **Flagi** – różne bity kontrolne (np. SYN, ACK, FIN), które służą do kontrolowania stanu połączenia.
- **Rozmiar okna** – służy do kontroli przepływu i określa, ile danych może być jednocześnie przesyłanych.
- **Suma kontrolna** – stosowana do wykrywania błędów w segmencie TCP.
- **Wskaźnik pilności** – informuje o istnieniu danych wymagających szybszego przetworzenia.

Warstwa 5 - Sesji (ang. Session Layer)

Zasada działania tej warstwy

Warstwa sesji jest odpowiedzialna za ustanawianie, zarządzanie oraz kończenie sesji komunikacyjnych między aplikacjami na różnych urządzeniach w sieci. Sesje są logicznymi połączeniami, które pozwalają na wymianę danych pomiędzy dwoma punktami końcowymi w sposób zorganizowany i kontrolowany. Warstwa ta zapewnia synchronizację, kontrolę dialogu oraz zarządzanie błędami związanymi z wymianą danych między aplikacjami.

Zasadnicze funkcje warstwy sesji obejmują:

- **Ustanawianie sesji** – nawiązywanie połączenia między aplikacjami i umożliwienie im wymiany danych.
- **Zarządzanie sesją** – kontrolowanie otwartości, organizacji i zamykania sesji.
- **Synchronizacja** – zapewnienie, że dane będą wymieniane w odpowiednich momentach (np. przy użyciu punktów kontrolnych).
- **Zarządzanie dialogiem** – umożliwienie komunikacji w trybie pełnodupleksowym, półdupleksowym lub jednostronnym.
- **Kontrola błędów sesji** – wykrywanie błędów w wymianie danych w ramach sesji i zapewnienie ich naprawy.

Warstwa sesji nie jest odpowiedzialna za samą transmisję danych (to zadanie warstwy transportowej), ale za organizację i zarządzanie wymianą danych pomiędzy aplikacjami.

Protokoły wykorzystane w tej warstwie

Chociaż warstwa sesji nie jest szeroko stosowana w porównaniu do innych warstw OSI, kilka protokołów, które działają na tej warstwie, zostały zaprojektowane do zarządzania sesjami w aplikacjach sieciowych:

- **RPC (Remote Procedure Call)** – protokół, który pozwala na wywoływanie funkcji zdalnych na innych maszynach, zapewniając synchronizację oraz kontrolowanie sesji komunikacyjnych.
- **NetBIOS (Network Basic Input/Output System)** – protokół zapewniający usługi sesji w sieci LAN, umożliwiający wymianę informacji o sesjach między urządzeniami.
- **SMB (Server Message Block)** – protokół, który umożliwia wymianę danych między komputerami w sieci lokalnej, oferujący zarówno funkcje warstwy sesji, jak i prezentacji.

- **PPTP (Point-to-Point Tunneling Protocol)** – protokół używany do tworzenia tuneli VPN, zarządzający sesjami dla połączeń zdalnych.
- **SSH (Secure Shell)** – protokół zapewniający bezpieczne połączenie między dwoma komputerami, zarządzający sesjami w trybie interaktywnym.

Słowa kluczowe do zapamiętania

- **Sesja** – logiczne połączenie między aplikacjami w sieci, które pozwala na wymianę danych w sposób kontrolowany.
- **Synchronizacja** – proces organizowania wymiany danych w czasie rzeczywistym, np. z punktami kontrolnymi.
- **Dialog** – interakcja pomiędzy aplikacjami, która może odbywać się w trybie pełnodupleksowym, półdupleksowym lub jednostronnym.
- **Kontrola błędów sesji** – wykrywanie i zarządzanie błędami, które mogą wystąpić podczas wymiany danych w sesji.
- **NetBIOS** – protokół umożliwiający realizację sesji w sieciach lokalnych.
- **RPC** – protokół umożliwiający zdalne wywoływanie funkcji w ramach sesji.

Przykład działania warstwy sesji

Warstwa sesji zapewnia utrzymanie połączenia pomiędzy dwoma aplikacjami. Na przykład, jeśli użytkownik loguje się na serwerze za pomocą protokołu SSH, warstwa sesji jest odpowiedzialna za utrzymanie i zarządzanie tą sesją. Protokół SSH nawiązuje połączenie, zapewnia szyfrowanie i integrację sesji, kontrolując synchronizację między urządzeniami.

Warstwa 6 - Prezentacji (ang. Presentation Layer)

Zasada działania tej warstwy

Warstwa prezentacji odpowiada za odpowiednią prezentację, formatowanie oraz konwersję danych w celu umożliwienia ich zrozumienia przez aplikację użytkownika. Obejmuje to kodowanie, kompresję, szyfrowanie i dekodowanie danych, zapewniając, że informacje mogą być wymieniane pomiędzy systemami o różnych formatach. Główne zadanie tej warstwy to zapewnienie spójności danych, tak aby aplikacje mogły poprawnie interpretować i przetwarzać przychodzące informacje, niezależnie od różnic w reprezentacji danych na różnych urządzeniach.

Do głównych funkcji warstwy prezentacji należą:

- **Kodowanie i dekodowanie** – konwersja danych na format, który jest zrozumiały dla aplikacji docelowej.
- **Kompresja** – zmniejszanie objętości danych w celu optymalizacji wykorzystania zasobów (np. przepustowości).
- **Szyfrowanie** – zapewnianie poufności danych przesyłanych przez sieć, np. przy użyciu algorytmów takich jak AES.
- **Konwersja formatów danych** – umożliwienie komunikacji pomiędzy systemami używającymi różnych kodowań i formatów danych (np. Unicode, ASCII).

Warstwa prezentacji nie zajmuje się zarządzaniem sesjami ani przekazywaniem danych, lecz jest odpowiedzialna za odpowiednie przygotowanie i formatowanie danych przed ich przekazaniem do warstwy aplikacji.

Protokoły wykorzystane w tej warstwie

Warstwa prezentacji nie ma tylu protokołów przypisanych, co inne warstwy modelu OSI, ponieważ jej funkcje są często realizowane przez protokoły wyższych warstw, jak aplikacyjne. Niemniej jednak istnieją protokoły, które działają w tej warstwie lub oferują funkcje odpowiedzialne za prezentację danych:

- **MIME (Multipurpose Internet Mail Extensions)** – standard używany do rozszerzania formatów wiadomości e-mail i innych danych w Internecie, obejmujący kodowanie treści.
- **TLS/SSL (Transport Layer Security/Secure Sockets Layer)** – protokoły odpowiedzialne za szyfrowanie danych przesyłanych przez sieć, które zapewniają bezpieczeństwo połączenia, używane m.in. w HTTPS.
- **JPEG (Joint Photographic Experts Group)** – standard kompresji obrazów, który jest używany w warstwie prezentacji do przesyłania skompresowanych danych graficznych.
- **ASCII (American Standard Code for Information Interchange)** – standard kodowania tekstu, który zapewnia zgodność w przekazywaniu danych tekstowych w różnych systemach.
- **Unicode** – standard kodowania znaków, który umożliwia reprezentację tekstu z różnych języków w jednym systemie.
- **GIF (Graphics Interchange Format)** – standard kompresji obrazów bitmapowych używany do reprezentowania grafiki w sieci.

Słowa kluczowe do zapamiętania

- **Kodowanie** – proces przekształcania danych w odpowiedni format, np. ASCII na Unicode.
- **Szyfrowanie** – proces zabezpieczania danych w celu zapewnienia ich poufności, np. z wykorzystaniem TLS.
- **Kompresja** – zmniejszanie rozmiaru danych, np. w celu oszczędności przepustowości (np. kompresja JPEG).
- **Format danych** – sposób, w jaki dane są reprezentowane w pamięci lub w przesyłanej transmisji, np. JPEG, GIF, ASCII.
- **TLS/SSL** – protokoły szyfrowania, zapewniające bezpieczeństwo przesyłanych danych.
- **MIME** – protokół do rozszerzania możliwości kodowania wiadomości w Internecie.

Przykład działania warstwy prezentacji

Warstwa prezentacji może zajmować się kodowaniem danych w celu ich przesłania między aplikacjami działającymi na różnych systemach operacyjnych. Na przykład w przypadku komunikacji za pomocą protokołu HTTPS, warstwa prezentacji używa TLS do szyfrowania przesyłanych danych i zapewnia, że są one odpowiednio zakodowane i zrozumiałe dla aplikacji odbiorczej.

Warstwa 7 - Aplikacji (ang. Application Layer)

Zasada działania tej warstwy

Warstwa aplikacji jest najwyższą warstwą modelu OSI i jest odpowiedzialna za interakcję użytkownika z siecią. To właśnie na tej warstwie działają aplikacje, które umożliwiają wymianę danych pomiędzy użytkownikami lub między użytkownikiem a systemem komputerowym. Warstwa aplikacji zajmuje się realizowaniem funkcji specyficznych dla aplikacji, takich jak wysyłanie wiadomości e-mail, pobieranie plików czy korzystanie z aplikacji webowych.

Do głównych zadań warstwy aplikacji należy:

- **Interfejs użytkownika** – zapewnienie użytkownikowi dostępu do usług sieciowych, takich jak e-mail, FTP, HTTP itp.
- **Współpraca z aplikacjami** – umożliwienie aplikacjom użytkownikom nawiązania komunikacji z innymi urządzeniami w sieci.
- **Zarządzanie protokołami aplikacyjnymi** – na przykład w przypadku HTTP, SMTP czy FTP, warstwa aplikacji zarządza odpowiednią komunikacją, definiując reguły wymiany danych.
- **Umożliwienie wymiany danych** – proces wysyłania i odbierania danych w formacie zrozumiałym przez odbiorcę.

Warstwa aplikacji zapewnia najbliższy interfejs do użytkownika i urządzeń sieciowych, zatem jest to warstwa, z którą użytkownicy bezpośrednio się stykają w kontekście interakcji z Internetem i sieciami.

Protokoły wykorzystane w tej warstwie

Warstwa aplikacji używa różnych protokołów, które umożliwiają komunikację pomiędzy aplikacjami działającymi na różnych urządzeniach w sieci. W tej warstwie znajdują się protokoły wykorzystywane do przesyłania różnych typów danych, takich jak tekst, obrazy czy dźwięk, pomiędzy urządzeniami. Przykłady protokołów wykorzystywanych w warstwie aplikacji to:

- **HTTP (Hypertext Transfer Protocol)** – protokół odpowiedzialny za przesyłanie stron internetowych w sieci WWW, umożliwiający komunikację między przeglądarką a serwerem WWW.
- **SMTP (Simple Mail Transfer Protocol)** – protokół wykorzystywany do przesyłania e-maili między serwerami pocztowymi.
- **FTP (File Transfer Protocol)** – protokół umożliwiający przesyłanie plików pomiędzy komputerami w sieci.
- **DNS (Domain Name System)** – protokół służący do tłumaczenia nazw domenowych na adresy IP, umożliwiający lokalizowanie usług w Internecie.
- **POP3 (Post Office Protocol 3)** – protokół do odbierania e-maili z serwera pocztowego.
- **IMAP (Internet Message Access Protocol)** – protokół umożliwiający dostęp do wiadomości e-mail przechowywanych na serwerze.
- **Telnet** – protokół używany do zdalnego logowania się na komputerach w sieci.
- **HTTPS (Hypertext Transfer Protocol Secure)** – bezpieczna wersja protokołu HTTP, wykorzystująca szyfrowanie SSL/TLS do ochrony danych.
- **SIP (Session Initiation Protocol)** – protokół wykorzystywany w komunikacji głosowej i wideo, np. w VoIP.

Słowa kluczowe do zapamiętania

- **HTTP** – protokół odpowiedzialny za przesyłanie stron internetowych.
- **SMTP** – protokół do przesyłania e-maili.
- **FTP** – protokół do transferu plików.
- **DNS** – system tłumaczenia nazw domenowych na adresy IP.
- **POP3** – protokół odbioru poczty elektronicznej.
- **IMAP** – protokół do zarządzania pocztą e-mail.
- **HTTPS** – bezpieczna wersja HTTP, używająca SSL/TLS.
- **Telnet** – protokół zdalnego logowania.
- **SIP** – protokół do inicjowania sesji komunikacji głosowej i wideo.
- **Usługi internetowe** – różne usługi zapewniające komunikację między aplikacjami, takie jak przesyłanie plików, e-maili czy stron WWW.

Przykład działania warstwy aplikacji

W kontekście przeglądania strony internetowej, warstwa aplikacji (HTTP) umożliwia użytkownikowi wysłanie zapytania do serwera WWW. Po otrzymaniu odpowiedzi od serwera (np. pliku HTML), warstwa aplikacji przetwarza te dane i wyświetla je użytkownikowi w postaci strony internetowej. Protokół HTTPS zapewnia, że dane przesyłane między przeglądarką a serwerem są zaszyfrowane i zabezpieczone przed nieautoryzowanym dostępem.

Powyższy diagram ilustruje sposób, w jaki klient (np. przeglądarka internetowa) używa protokołu HTTP/HTTPS do komunikacji z serwerem WWW, wysyłając zapytanie i otrzymując odpowiedź w postaci strony internetowej.

Podsumowanie modelu ISO/OSI

Poniżej przedstawiono tabelę podsumowującą warstwy sieciowe:

Numer	Warstwa modelu ISO/OSI	Słowa kluczowe
Numer	**Warstwa modelu ISO/OSI**	**Słowa kluczowe**
1	Warstwa fizyczna	Kabel, Bit, Sygnalizacja, Medium transmisyjne
2	Warstwa łącza danych	Ethernet, MAC, Switch, RAMKA, Adresowanie
3	Warstwa sieciowa	IP, Routing, Adresowanie, Router, IP Fragmentation
4	Warstwa transportowa	TCP, UDP, Porty, Segmentacja, Połączenie
5	Warstwa sesji	Synchronizacja, Sesja, Token, Autoryzacja
6	Warstwa prezentacji	Kodowanie, Kompresja, Szyfrowanie, ASCII, JPEG
7	Warstwa aplikacji	HTTP, FTP, SMTP, DNS, POP3, IMAP, Telnet

Enkapsulacja

Enkapsulacja to proces, w którym dane, począwszy od warstwy aplikacji, przechodzą przez wszystkie warstwy modelu OSI (lub TCP/IP), w każdej z nich otrzymując dodatkowe informacje niezbędne do prawidłowej komunikacji w sieci. Celem enkapsulacji jest dostosowanie danych do wymogów każdej z

warstw, a także zapewnienie integralności oraz odpowiednich adresów, które pozwolą na prawidłowe przesyłanie informacji między urządzeniami w sieci.

- **Warstwa aplikacji** – na najwyższym poziomie dane są przygotowywane przez aplikację użytkownika. Mogą to być różne formaty, np. pliki tekstowe, obrazy, czy wiadomości e-mail.
- **Warstwa prezentacji** – tutaj dane mogą zostać zakodowane lub zaszyfrowane (np. SSL/TLS dla HTTP), aby zapewnić ochronę informacji.
- **Warstwa sesji** – organizuje i synchronizuje sesje komunikacyjne, utrzymując stan połączenia.
- **Warstwa transportowa** – protokoły takie jak TCP lub UDP dodają odpowiednie nagłówki, które zawierają informacje o porcie docelowym, numerze sekwencyjnym (w przypadku TCP) i innych parametrach.
- **Warstwa sieciowa** – dodawany jest adres IP, który umożliwia poprawne skierowanie danych w sieci.
- **Warstwa łącza danych** – w tej warstwie dodawane są adresy MAC urządzeń, a dane są podzielone na ramki.
- **Warstwa fizyczna** – ostatecznie dane zostają zamienione na sygnał elektryczny, optyczny lub radiowy, w zależności od medium transmisyjnego.

Proces enkapsulacji zachodzi, gdy dane przechodzą od najwyższej warstwy do warstwy fizycznej. Każda warstwa dodaje swoje nagłówki, które są wykorzystywane do właściwego przesyłania danych przez sieć. Każdy dodany nagłówek zawiera informacje, które pozwalają urządzeniom pośredniczącym w sieci (takim jak routery, przełączniki, bramy) na prawidłowe dostarczenie danych do miejsca docelowego.

Dekapsulacja

Dekapsulacja jest odwrotnym procesem do enkapsulacji i zachodzi, gdy dane są odbierane przez urządzenie docelowe. Podczas dekapsulacji każdy nagłówek dodany przez różne warstwy jest usuwany w odwrotnej kolejności, co pozwala na odzyskanie oryginalnych danych.

Proces dekapsulacji odbywa się, gdy dane przechodzą od warstwy fizycznej w górę modelu OSI:

- **Warstwa fizyczna** – odbierane są sygnały elektryczne lub optyczne i przekształcane na dane binarne.
- **Warstwa łącza danych** – dane są odbierane w postaci ramek. Nagłówek ramki, który zawiera adresy MAC i inne informacje, jest usuwany, pozostawiając jedynie dane.
- **Warstwa sieciowa** – usuwany jest nagłówek IP, który zawierał informacje o adresach IP nadawcy i odbiorcy.
- **Warstwa transportowa** – nagłówek TCP/UDP, który zawiera informacje o numerach portów i sekwencji danych, jest usuwany.
- **Warstwa sesji** – jeśli wymagane, są odbudowywane sesje komunikacyjne.
- **Warstwa prezentacji** – w razie potrzeby następuje dekodowanie, deszyfrowanie lub dekompresja danych.
- **Warstwa aplikacji** – na końcu dane są dostarczane do odpowiedniej aplikacji użytkownika, takiej jak przeglądarka internetowa, klient e-mail, czy klient FTP.

Dekapsulacja jest kluczowym procesem, ponieważ to dzięki niemu urządzenie odbierające dane może przeanalizować i odpowiednio przekazać dane do aplikacji, która miała je otrzymać.

Rodzaje mediów transmisyjnych w sieciach komputerowych

Media przewodowe

Media przewodowe to jedno z najstarszych i najczęściej wykorzystywanych rozwiązań w sieciach komputerowych. Charakteryzują się wysoką niezawodnością oraz dużą przepustowością, jednak ich instalacja wiąże się z koniecznością fizycznego układania kabli, co może stanowić ograniczenie w niektórych przypadkach.

Światłowodowe

Media światłowodowe wykorzystują światłowody do przesyłania sygnałów w postaci światła. Światłowody oferują szereg zalet, takich jak bardzo duża przepustowość, niskie straty sygnału na dużych odległościach oraz odporność na zakłócenia elektromagnetyczne.

Światłowody dzielą się na dwa główne typy:

- **Światłowody jednomodowe** – stosowane do transmisji na bardzo duże odległości (nawet kilkaset kilometrów), posiadają jedno włókno optyczne, przez które przesyłane jest światło o jednej długości fali.
- **Światłowody wielomodowe** – stosowane w krótszych odległościach, mają większą średnicę rdzenia, co pozwala na transmisję wielu modów (rodzajów) światła.

Światłowody zapewniają bardzo dużą szybkość transmisji (nawet do kilkudziesięciu gigabitów na sekundę) i są odporne na zakłócenia zewnętrzne. Są jednak droższe w instalacji niż inne media przewodowe.

Metalowe

Metalowe media transmisyjne, takie jak kable miedziane, są wykorzystywane głównie w sieciach lokalnych (LAN) oraz w połączeniach telefonicznych. Wykorzystują one przewodniki metalowe, przez które przesyłane są impulsy elektryczne. W zależności od rodzaju kabla, można wyróżnić następujące typy:

- **Kable koncentryczne** – składają się z pojedynczego przewodnika, otoczonego izolacją, ekranem oraz warstwą ochronną. Kable te były wykorzystywane głównie w starszych sieciach, jednak ze względu na ograniczoną przepustowość i podatność na zakłócenia zostały zastąpione przez kable skrętne i światłowody.
- **Kable skrętne (UTP, STP)** – najczęściej stosowane w nowoczesnych sieciach LAN. UTP (Unshielded Twisted Pair) to kable bez ekranowania, natomiast STP (Shielded Twisted Pair) mają dodatkowe ekranowanie, co zapewnia lepszą odporność na zakłócenia elektromagnetyczne. Kable skrętne oferują przepustowość do 10 Gbps na krótkich dystansach, ale ich jakość może się pogarszać w wyniku zakłóceń.

Kable metalowe, mimo że oferują niższą przepustowość niż światłowody, są łatwiejsze do instalacji i tańsze.

Media bezprzewodowe

Media bezprzewodowe stanowią alternatywę dla tradycyjnych mediów przewodowych. Przesyłanie danych odbywa się za pomocą fal radiowych, mikrofalowych, podczerwieni lub innych form promieniowania elektromagnetycznego, co pozwala na elastyczność w projektowaniu sieci. Chociaż media bezprzewodowe zapewniają mobilność, mogą mieć niższą przepustowość i większe narażenie na zakłócenia.

Wi-Fi

Wi-Fi to jedna z najpopularniejszych technologii bezprzewodowych stosowanych w sieciach lokalnych. Używa fal radiowych do przesyłania danych w paśmie 2.4 GHz lub 5 GHz (oraz nowszych wersjach w paśmie 6 GHz). Wi-Fi pozwala na tworzenie sieci LAN bez potrzeby używania kabli. Standardy Wi-Fi, takie jak 802.11n, 802.11ac, i 802.11ax, oferują różne przepustowości oraz zasięgi w zależności od wersji.

- **Wi-Fi 4 (802.11n)** – oferuje prędkości do 600 Mbps.
- **Wi-Fi 5 (802.11ac)** – umożliwia osiągnięcie prędkości do 3.5 Gbps w paśmie 5 GHz.
- **Wi-Fi 6 (802.11ax)** – poprawia przepustowość do 9.6 Gbps oraz zwiększa efektywność w sieciach zatłoczonych.

Wi-Fi jest popularnym rozwiązaniem w biurach, domach oraz na uczelniach, zapewniając wygodny dostęp do sieci.

Bluetooth

Bluetooth to technologia bezprzewodowa przeznaczona do łączenia urządzeń w krótkich odległościach (do 100 metrów). Jest stosunkowo wolniejsza niż Wi-Fi, ale charakteryzuje się niskim zużyciem energii, co sprawia, że jest idealna do łączenia urządzeń mobilnych, słuchawek, klawiatur, czy urządzeń IoT (Internet of Things). Bluetooth operuje w paśmie 2.4 GHz i jest wykorzystywany do wymiany małych ilości danych.

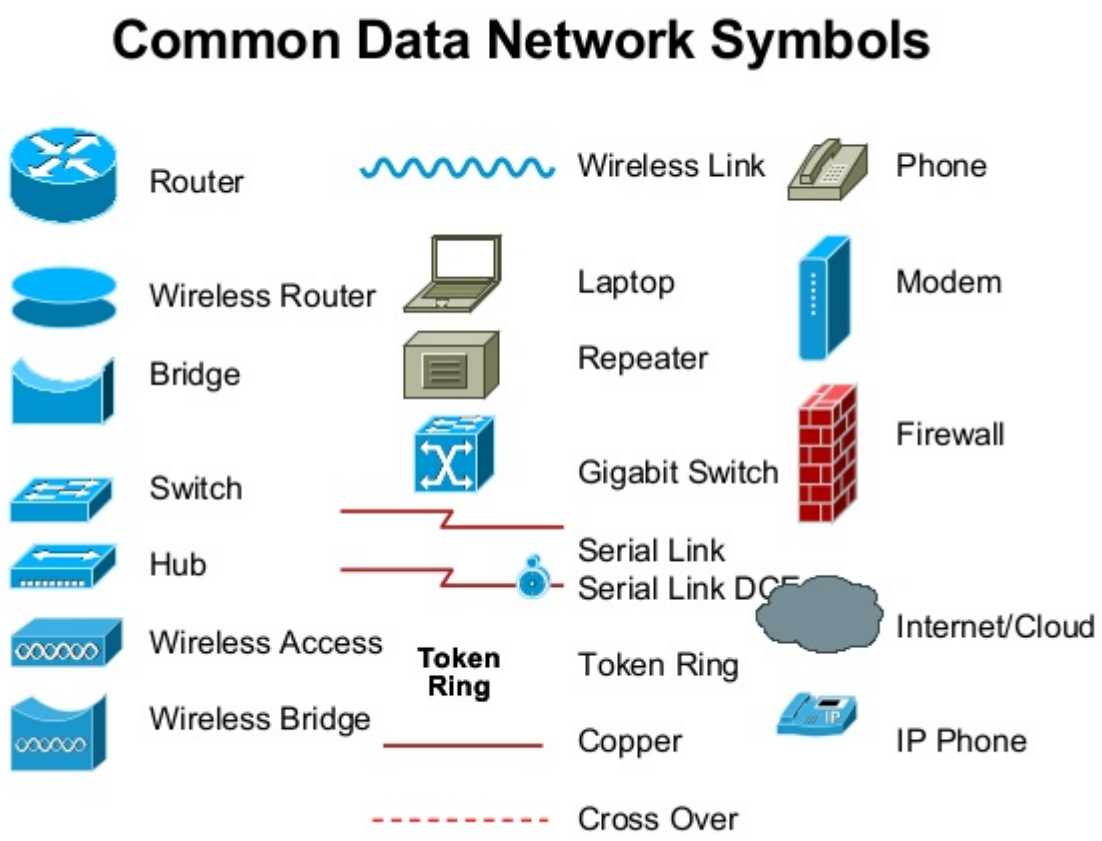
Sieci komórkowe (5G, LTE)

Sieci komórkowe zapewniają mobilną transmisję danych w technologii 3G, 4G, a obecnie 5G. 5G oferuje wyjątkowo dużą przepustowość (nawet do 10 Gbps) oraz niskie opóźnienia, co sprawia, że jest idealne do zastosowań takich jak streaming wideo w rozdzielczości 8K, autonomiczne pojazdy, czy transmisja danych z urządzeń IoT w czasie rzeczywistym. Sieci 5G operują na wyższych częstotliwościach (do 100 GHz w niektórych pasmach) i zapewniają znacznie wyższą prędkość transmisji w porównaniu do poprzednich technologii.

- **LTE (Long-Term Evolution)** – obecnie najpopularniejsza technologia mobilna, oferująca prędkości rzędu 100 Mbps do 1 Gbps w trybie 4G.

- **5G** – technologia nowej generacji, umożliwiająca prędkości powyżej 10 Gbps.

Rodzaje urządzeń w sieciach komputerowych



Zestawienie symboli urządzeń sieciowych

Koncentrator (ang. Hub)

Opis i działanie

Koncentrator (hub) to podstawowe urządzenie sieciowe, które działa na warstwie fizycznej modelu OSI. Jego główną funkcją jest rozdzielanie sygnałów przychodzących z różnych urządzeń w sieci do wszystkich innych podłączonych urządzeń. Hub działa na zasadzie „rozgłaszania” sygnałów do wszystkich portów, niezależnie od tego, do którego urządzenia sygnał jest skierowany.

Zastosowanie

Koncentratory były popularne w starszych sieciach Ethernetowych, ale zostały w dużej mierze zastąpione przez przełączniki, ponieważ huby nie potrafią efektywnie zarządzać ruchem w sieci, co

może prowadzić do dużych opóźnień oraz kolizji. Niemniej jednak w prostych i małych sieciach lokalnych huby mogą być nadal wykorzystywane.

Przełącznik (ang. Switch)

Opis i działanie

Przełącznik (switch) to bardziej zaawansowane urządzenie niż hub, działające na warstwie 2 modelu OSI (warstwa łącza danych). Switch, w przeciwieństwie do hubu, nie rozsyła sygnału do wszystkich portów, ale kieruje go tylko do odpowiedniego urządzenia w sieci. Przełącznik używa tablicy MAC (Media Access Control), aby śledzić, które urządzenie jest podłączone do którego portu i w ten sposób przesyła dane bezpośrednio do odpowiednich urządzeń.

Zastosowanie

Przełączniki są podstawowym urządzeniem w nowoczesnych sieciach LAN. Umożliwiają one bardziej efektywne zarządzanie ruchem sieciowym i eliminują problem kolizji, co pozwala na uzyskanie wyższych prędkości transmisji w sieci. Przełączniki mogą obsługiwać różne technologie, w tym gigabitowe oraz 10-gigabitowe Ethernet.

Most przewodowy lub bezprzewodowy (ang. Wireless or wired bridge)

Opis i działanie

Most (bridge) to urządzenie, które łączy dwie oddzielne sieci, umożliwiając im komunikację. Może to być most przewodowy, który łączy dwa segmenty sieci przewodowych, lub most bezprzewodowy, który łączy dwie sieci bezprzewodowe. Mosty działają na warstwie 2 modelu OSI i są odpowiedzialne za przekazywanie danych między różnymi segmentami sieci.

Zastosowanie

Mosty są stosowane w celu rozszerzenia zasięgu sieci, gdy jeden segment sieci nie obejmuje całego obszaru wymagającego pokrycia. Mosty są użyteczne w sytuacjach, gdy trzeba połączyć dwie lokalizacje w ramach jednej sieci, np. w dużych biurowcach, gdzie segmenty sieci muszą zostać połączone.

Ruter (ang. Router)

Opis i działanie

Ruter (router) to urządzenie działające na warstwie 3 modelu OSI, czyli warstwie sieciowej. Jego zadaniem jest przekazywanie pakietów danych między różnymi sieciami, w tym pomiędzy sieciami lokalnymi (LAN) a globalnym Internetem. Router podejmuje decyzje o trasie pakietów, wykorzystując tablice routingu i protokoły routingu (np. OSPF, BGP).

Zastosowanie

Routery są kluczowym elementem infrastruktury sieciowej, umożliwiającym komunikację między różnymi sieciami. Są one używane w domowych, biurowych oraz korporacyjnych sieciach, a także stanowią element infrastruktury internetu. Routery zapewniają także funkcje zabezpieczeń, takie jak zapora ogniowa (firewall), a także przydzielanie adresów IP za pomocą DHCP.

Urządzenie końcowe (ang. End device)

Opis i działanie

Urządzenia końcowe to wszystkie urządzenia, które korzystają z sieci komputerowej do komunikacji. Są to np. komputery, smartfony, drukarki, kamery IP czy inne urządzenia podłączone do sieci. Urządzenia końcowe mogą działać na różnych warstwach modelu OSI, zależnie od ich roli w sieci.

Zastosowanie

Urządzenia końcowe są kluczowe w każdej sieci komputerowej, ponieważ to one są źródłem i odbiorcą danych w komunikacji sieciowej. Są one wykorzystywane w praktycznie każdej sieci LAN, WAN, Wi-Fi oraz innych typach sieci.

Serwer (ang. Server)

Opis i działanie

Serwer to urządzenie, które zapewnia usługi i zasoby innym urządzeniom w sieci (tzw. klientom). Działa na różnych warstwach modelu OSI, w zależności od typu usługi, jaką świadczy. Serwery mogą obsługiwać różne rodzaje usług, takie jak udostępnianie plików, obsługa baz danych, poczty elektronicznej, aplikacji czy stron internetowych.

Zastosowanie

Serwery stanowią fundament infrastruktury IT w firmach, instytucjach edukacyjnych, administracji publicznej oraz wielu innych. Używane są do przechowywania danych, zapewniania dostępu do

aplikacji i usług online, a także do zarządzania ruchem sieciowym. W kontekście internetu serwery webowe oraz serwery baz danych są kluczowe w świadczeniu usług online.

Bezprzewodowy punkt dostępowy (ang. Wireless access point)

Opis i działanie

Bezprzewodowy punkt dostępowy (Access Point, AP) to urządzenie, które umożliwia urządzeniom bezprzewodowym (np. laptopom, smartfonom) połączenie się z siecią przewodową. Punkt dostępowy działa na warstwie 2 modelu OSI i pełni rolę mostu, który przekłada dane między urządzeniami bezprzewodowymi a siecią przewodową.

Zastosowanie

Punkty dostępowe są stosowane w sieciach Wi-Fi, gdzie umożliwiają urządzeniom bezprzewodowym dostęp do zasobów sieci LAN. Są powszechnie wykorzystywane w biurach, domach, na uczelniach oraz w przestrzeniach publicznych, takich jak kawiarnie czy lotniska, gdzie dostęp do internetu jest udostępniany użytkownikom bezprzewodowo.

Hybrydowe urządzenia sieciowe

Routery do małego biura lub domu (ang. Small office home office routers, SOHO routers)

Opis i działanie Routery SOHO (Small Office Home Office) to urządzenia przeznaczone do użytku w małych biurach lub domowych sieciach komputerowych. Oprócz podstawowej funkcji routingu, czyli przekazywania pakietów między różnymi sieciami (np. między siecią lokalną a internetem), routery SOHO często oferują dodatkowe funkcje, takie jak wbudowany firewall, router Wi-Fi, funkcje QoS (Quality of Service) oraz DHCP (Dynamic Host Configuration Protocol).

Routery SOHO są urządzeniami wszechstronnymi, które integrują w sobie wiele funkcji, aby uprościć konfigurację sieci i zarządzanie nią w małych organizacjach.

Zastosowanie Routery SOHO znajdują zastosowanie głównie w małych biurach oraz domach. Służą do zapewnienia dostępu do internetu, zarządzania ruchem w sieci, udostępniania zasobów sieciowych oraz zapewnienia bezpieczeństwa (np. przez zaporę ogniową). Popularne modele takich urządzeń to np. routery Wi-Fi, które umożliwiają bezprzewodowy dostęp do sieci lokalnej.

Firewalle nowej generacji (ang. Next-Gen Firewall)

Opis i działanie Firewalle nowej generacji (NGFW, Next-Generation Firewalls) to nowoczesne urządzenia zabezpieczające, które łączą tradycyjne funkcje firewalla z dodatkowymi mechanizmami,

takimi jak inspekcja aplikacji, zapobieganie włamaniom (IPS/IDS), analiza w czasie rzeczywistym oraz kontrola dostępu do zasobów na podstawie użytkownika, aplikacji czy urządzenia.

NGFW działają na warstwie 3-7 modelu OSI, co pozwala na głębszą analizę pakietów, identyfikację aplikacji oraz ich ruchu, a także wykrywanie i blokowanie zagrożeń w czasie rzeczywistym.

Zastosowanie Firewalle nowej generacji są stosowane w dużych organizacjach, aby chronić zasoby sieciowe przed zagrożeniami z zewnątrz, takimi jak ataki DDoS, złośliwe oprogramowanie czy nieautoryzowane próby dostępu. NGFW są także wykorzystywane w małych biurach i domowych sieciach, aby zapewnić wyższy poziom ochrony i lepszą kontrolę nad ruchem sieciowym.

Analizatory ruchu sieciowego (ang. Network traffic analyzers)

Opis i działanie Analizatory ruchu sieciowego to narzędzia służące do monitorowania, przechwytywania i analizy pakietów danych w sieci. Dzięki analizatorom ruchu sieciowego administratorzy mogą śledzić aktywność w sieci, identyfikować problemy z wydajnością, monitorować przepustowość oraz wykrywać niepożądany ruch, taki jak ataki typu man-in-the-middle czy złośliwe oprogramowanie.

Analizatory ruchu sieciowego mogą pracować w trybie pasywnym (gdzie jedynie obserwują i przechwytują dane) lub aktywnym (gdzie także generują pakiety testowe w celu diagnozy problemów w sieci). Popularnym narzędziem tego typu jest Wireshark.

Zastosowanie Analizatory ruchu sieciowego są niezbędnym narzędziem w pracy administratorów sieci. Używa się ich do monitorowania wydajności sieci, wykrywania problemów z przepustowością, diagnostyki sieciowej, wykrywania nieautoryzowanego dostępu oraz analizowania danych w celu identyfikacji potencjalnych zagrożeń. Często stosowane są w dużych przedsiębiorstwach, centrach danych oraz w instytucjach zajmujących się bezpieczeństwem IT.

Bezprzewodowe sieci siatkowe (ang. Wireless mesh networks)

Opis i działanie Bezprzewodowe sieci siatkowe (WMS, Wireless Mesh Networks) to typ sieci bezprzewodowych, w których urządzenia (np. routery, punkty dostępowe) tworzą dynamiczną, samokonfigurowalną sieć, w której każdy węzeł (urządzenie) może pełnić rolę zarówno nadawcy, jak i odbiorcy sygnału. Sieć siatkowa umożliwia przekazywanie danych poprzez inne urządzenia, co zwiększa zasięg sieci i pozwala na redundancję.

W przeciwieństwie do tradycyjnych sieci bezprzewodowych, gdzie dane muszą przechodzić przez jeden punkt dostępu, w sieci siatkowej dane mogą podróżować wieloma drogami, co zapewnia większą odporność na awarie i zapewnia lepszą wydajność.

Zastosowanie Bezprzewodowe sieci siatkowe są idealne do stosowania w rozległych obszarach, takich jak duże biura, kampusy uniwersyteckie, osiedla mieszkaniowe czy przestrzenie publiczne. Ze względu na elastyczność, odporność na zakłócenia oraz możliwość samonaprawy, są one wykorzystywane w miejscach, gdzie nie ma łatwego dostępu do infrastruktury przewodowej, a także tam, gdzie wymagany jest duży zasięg sieci.

Przemysłowe przełączniki (ang. Industrial switches)

Opis i działanie Przemysłowe przełączniki to specjalistyczne urządzenia sieciowe zaprojektowane do pracy w trudnych warunkach, takich jak wysokie temperatury, wilgotność, wibracje czy zapylenie. Działają na tej samej zasadzie co standardowe przełączniki, ale charakteryzują się większą niezawodnością, trwałością i odpornością na warunki przemysłowe.

Przemysłowe przełączniki często oferują dodatkowe funkcje, takie jak możliwość zarządzania siecią (przełączniki zarządzalne) oraz funkcje redundancji, które zapewniają ciągłość pracy sieci w przypadku awarii jednego z urządzeń.

Zastosowanie Przemysłowe przełączniki są wykorzystywane w zastosowaniach, gdzie wymagana jest niezawodność i odporność na trudne warunki otoczenia. Przykładowe miejsca ich zastosowania to fabryki, linie produkcyjne, systemy monitoringu wizyjnego w przemyśle, infrastruktura krytyczna, czy energetyka.

Sieci VLAN 802.1Q

Definicja sieci VLAN 802.1Q

VLAN (Virtual Local Area Network) to logicznie wydzielona sieć w ramach fizycznej infrastruktury, która pozwala na grupowanie urządzeń znajdujących się w różnych lokalizacjach w jedną spójną sieć. Standard 802.1Q jest jedną z najpopularniejszych metod tagowania ramek Ethernetowych, które umożliwiają segregację ruchu w sieciach LAN, tworząc wirtualne podsieci.

Standard 802.1Q, opracowany przez IEEE, pozwala na przesyłanie ramek Ethernet z dodanym tagiem VLAN, co umożliwia odróżnienie różnych wirtualnych sieci na tym samym medium fizycznym. Tagi VLAN są dodawane do ramki Ethernet, pozwalając przełącznikom na rozróżnianie, do której VLAN należy konkretna ramka. Tagowanie odbywa się poprzez dodanie specjalnego pola do ramki, które zawiera identyfikator VLAN (VID - VLAN Identifier), który wskazuje do jakiej sieci wirtualnej należy dana ramka.

Struktura ramki 802.1Q

W standardzie 802.1Q ramka Ethernetowa jest rozszerzona o dodatkowe pole, które zawiera informacje o VLAN. Tagi VLAN są umieszczane pomiędzy polem adresu MAC a polem typu (ethernet type). Struktura ramki z tagiem 802.1Q wygląda następująco:

- **Adres MAC docelowy** (6 bajtów)
- **Adres MAC źródłowy** (6 bajtów)
- **Typ / Długość** (2 bajty)
- **Tag VLAN 802.1Q** (4 bajty):
    - **Tag Priority** (3 bity) – określa priorytet ruchu w sieci.
    - **ID VLAN** (12 bitów) – identyfikator VLAN, który identyfikuje wirtualną sieć (0–4095).
    - **CFI (Canonical Format Indicator)** (1 bit) – wskazuje, czy ramka jest w formacie

kanonicznym.
- **Dane** (zależne od ramki)
- **Suma kontrolna** (4 bajty)

Tag VLAN 802.1Q zapewnia możliwość identyfikacji sieci VLAN w ramach jednej fizycznej sieci, umożliwiając na przykład izolowanie ruchu różnych grup użytkowników, aplikacji czy działów organizacji.

Zastosowania sieci VLAN

Sieci VLAN znajdują szerokie zastosowanie w różnych środowiskach sieciowych. Oto niektóre z głównych zastosowań:

- **Izolacja ruchu**: Sieci VLAN pozwalają na izolowanie ruchu pomiędzy różnymi grupami użytkowników lub urządzeń, co zwiększa bezpieczeństwo i zarządzalność siecią.
- **Segmentacja sieci**: Dzięki VLAN możliwe jest podzielenie jednej fizycznej sieci na mniejsze, logiczne podsieci. Może to pomóc w organizowaniu infrastruktury, jak na przykład oddzielanie ruchu pracowników biurowych od ruchu z urządzeń IoT.
- **Optymalizacja wykorzystania pasma**: VLAN pozwala na zarządzanie ruchem, zmniejszając obciążenie w ramach dużych sieci i przyspieszając transmisję między urządzeniami w tej samej VLAN.
- **Bezpieczeństwo**: Sieci VLAN mogą ograniczać dostęp do wrażliwych zasobów, takich jak serwery, przez umożliwienie komunikacji tylko urządzeniom z odpowiednich VLAN.
- **Zarządzanie ruchem**: VLAN pozwalają na przypisanie różnych poziomów priorytetów do różnych typów ruchu, co umożliwia lepsze zarządzanie jakością usług (QoS).

Zalety stosowania sieci VLAN

Stosowanie sieci VLAN niesie ze sobą wiele korzyści, które przyczyniają się do lepszej organizacji oraz efektywności sieci:

- **Zwiększenie bezpieczeństwa**: VLAN pozwalają na oddzielenie różnych grup użytkowników i urządzeń, co utrudnia nieautoryzowany dostęp do wrażliwych zasobów. Na przykład, w firmie dział IT może być izolowany od reszty użytkowników, co zapewnia dodatkową warstwę ochrony.
- **Prostsze zarządzanie siecią**: Dzięki VLAN administratorzy mogą łatwiej zarządzać dużymi sieciami, przypisując użytkowników i urządzenia do odpowiednich podsieci w zależności od ich potrzeb. Ułatwia to również dodawanie nowych urządzeń do sieci bez konieczności przekształcania fizycznej infrastruktury.
- **Redukcja kosztów**: VLAN umożliwia podział jednej fizycznej sieci na wiele logicznych, co pozwala na zmniejszenie kosztów związanych z tworzeniem nowych fizycznych połączeń sieciowych.
- **Zwiększenie wydajności**: Zmniejszenie liczby urządzeń w jednej sieci VLAN redukuje rozgłoszenia, co zmniejsza obciążenie sieci i poprawia wydajność.
- **Lepsza kontrola przepustowości**: VLAN umożliwiają przydzielanie odpowiednich priorytetów do różnych typów ruchu, co umożliwia efektywne zarządzanie pasmem i zapewnia, że aplikacje o wysokim priorytecie (np. VoIP) będą miały odpowiednią jakość usług.

Przypadki zastosowania sieci VLAN w środowisku sieciowym

Izolacja ruchu między działami firmy

W organizacjach, które posiadają różne działy (np. IT, HR, finansowy), VLAN może być użyty do oddzielenia ruchu pomiędzy tymi działami, co zapewnia lepsze bezpieczeństwo i kontrolę. Na przykład, dział IT może mieć dostęp do wszystkich zasobów sieciowych, podczas gdy dział HR może mieć dostęp tylko do swoich zasobów, a ruch między tymi działami będzie izolowany.

Podział sieci na strefy bezpieczeństwa

W większych organizacjach VLAN mogą być używane do stworzenia stref bezpieczeństwa, w których np. urządzenia IoT, systemy monitoringu lub serwery bazy danych będą izolowane od reszty sieci. Takie podejście minimalizuje ryzyko ataków na wrażliwe urządzenia.

Oddzielanie sieci gości od sieci wewnętrznej

Wiele firm oferuje dostęp do sieci gościom (np. dla odwiedzających). VLAN może być wykorzystany do stworzenia oddzielnej sieci gości, zapewniając tym samym izolację ruchu gości od zasobów wewnętrznych firmy.

Przypisanie priorytetów do aplikacji

W sieciach VLAN można stosować QoS (Quality of Service), aby przypisać wyższy priorytet dla krytycznych aplikacji, takich jak komunikacja głosowa (VoIP). Dzięki temu głosowe połączenia IP mają zapewnioną odpowiednią jakość, nawet w przypadku dużego obciążenia sieci.

Rozdzielenie ruchu sieciowego w kampusach

W środowisku kampusowym VLAN mogą być używane do podziału różnych typów ruchu – np. ruchu administracyjnego, danych i multimediów – na różne sieci wirtualne. Dzięki temu można zminimalizować zakłócenia i poprawić wydajność.

Podsumowanie

Sieci VLAN oparte na standardzie 802.1Q są istotnym narzędziem w nowoczesnych sieciach komputerowych. Dzięki możliwości tworzenia logicznych podsieci na jednej fizycznej infrastrukturze, VLAN pozwalają na efektywne zarządzanie ruchem sieciowym, zapewnienie bezpieczeństwa i optymalizację wydajności. VLAN oferują liczne korzyści, takie jak izolacja ruchu, łatwiejsze zarządzanie siecią, redukcja kosztów oraz lepsza kontrola nad przepustowością, co czyni je niezastąpionym rozwiązaniem w dużych organizacjach i zaawansowanych środowiskach sieciowych.

Konfiguracja VLAN na przełącznikach Cisco Catalyst oraz trunk

W tej sekcji przedstawimy przykładową konfigurację VLANów na dwóch przełącznikach Cisco Catalyst oraz trunku między nimi. Trunk umożliwia przesyłanie ruchu z wielu VLANów przez jedno połączenie fizyczne.

Założenia

W tym przykładzie przyjmujemy następującą konfigurację:

- Przełączniki: Switch1 i Switch2 (Cisco Catalyst)
- VLAN 10 – Działy sprzedaży
- VLAN 20 – Działy IT
- VLAN 30 – Działy HR
- Interfejs trunk: Gi0/1 na Switch1 i Gi0/1 na Switch2

Kroki konfiguracji

Tworzenie VLANów na Switch1

Na pierwszym przełączniku (Switch1) należy utworzyć VLANy oraz przypisać interfejsy do odpowiednich VLANów. Poniżej znajduje się przykładowa konfiguracja:

```
Switch1# configure terminal
Switch1(config)# vlan 10
Switch1(config-vlan)# name Sales
Switch1(config-vlan)# exit
Switch1(config)# vlan 20
Switch1(config-vlan)# name IT
Switch1(config-vlan)# exit
Switch1(config)# vlan 30
Switch1(config-vlan)# name HR
Switch1(config-vlan)# exit
```

Powyższa konfiguracja tworzy trzy VLANy: VLAN 10 (Sales), VLAN 20 (IT), oraz VLAN 30 (HR) na Switch1.

Przypisanie portów do VLANów na Switch1

Teraz przypisujemy interfejsy do odpowiednich VLANów. Załóżmy, że porty FastEthernet 0/1, 0/2 i 0/3 są przypisane do VLANów 10, 20 i 30.

```
Switch1(config)# interface range fa0/1 - 3
Switch1(config-if-range)# switchport mode access
Switch1(config-if-range)# switchport access vlan 10
Switch1(config-if-range)# exit
```

Ponadto, przypisujemy porty FastEthernet 0/4 i 0/5 do VLAN 20 oraz VLAN 30:

```
Switch1(config)# interface fa0/4
Switch1(config-if)# switchport mode access
Switch1(config-if)# switchport access vlan 20
Switch1(config-if)# exit
Switch1(config)# interface fa0/5
Switch1(config-if)# switchport mode access
Switch1(config-if)# switchport access vlan 30
Switch1(config-if)# exit
```

Konfiguracja trunku na Switch1

Aby umożliwić przesyłanie ruchu z wielu VLANów przez jedno połączenie, musimy skonfigurować port trunkowy na interfejsie Gi0/1 na Switch1:

```
Switch1(config)# interface gigabitethernet0/1
Switch1(config-if)# switchport mode trunk
Switch1(config-if)# switchport trunk encapsulation dot1q
Switch1(config-if)# exit
```

Konfiguracja VLANów na Switch2

Na drugim przełączniku (Switch2) tworzymy te same VLANy:

```
Switch2# configure terminal
Switch2(config)# vlan 10
Switch2(config-vlan)# name Sales
Switch2(config-vlan)# exit
Switch2(config)# vlan 20
Switch2(config-vlan)# name IT
Switch2(config-vlan)# exit
Switch2(config)# vlan 30
Switch2(config-vlan)# name HR
Switch2(config-vlan)# exit
```

Przypisanie portów do VLANów na Switch2

Na Switch2 przypisujemy porty do odpowiednich VLANów. Załóżmy, że porty FastEthernet 0/1, 0/2 i 0/3 są przypisane do VLANów 10, 20 i 30.

```
Switch2(config)# interface range fa0/1 - 3
Switch2(config-if-range)# switchport mode access
Switch2(config-if-range)# switchport access vlan 10
Switch2(config-if-range)# exit
```

Podobnie, przypisujemy porty FastEthernet 0/4 i 0/5 do VLAN 20 oraz VLAN 30:

```
Switch2(config)# interface fa0/4
Switch2(config-if)# switchport mode access
Switch2(config-if)# switchport access vlan 20
Switch2(config-if)# exit
Switch2(config)# interface fa0/5
Switch2(config-if)# switchport mode access
Switch2(config-if)# switchport access vlan 30
Switch2(config-if)# exit
```

Konfiguracja trunku na Switch2

Na Switch2 również musimy skonfigurować trunk na interfejsie Gi0/1:

```
Switch2(config)# interface gigabitethernet0/1
Switch2(config-if)# switchport mode trunk
Switch2(config-if)# switchport trunk encapsulation dot1q
Switch2(config-if)# exit
```

Testowanie konfiguracji

Po zakończeniu konfiguracji VLANów i trunku na obu przełącznikach, możemy przeprowadzić testy, aby upewnić się, że sieć działa poprawnie.

1. **Sprawdzenie VLANów**: Aby sprawdzić, czy VLANy zostały poprawnie skonfigurowane na przełączniku, używamy komendy:

```
Switch# show vlan brief
```

2. **Testowanie połączenia między VLANami**: Aby sprawdzić komunikację między urządzeniami w różnych VLANach, należy użyć komendy `ping` z urządzeń w różnych VLANach. Komunikacja między VLANami wymaga konfiguracji routingu między VLANami (np. poprzez router lub Layer 3 switch).

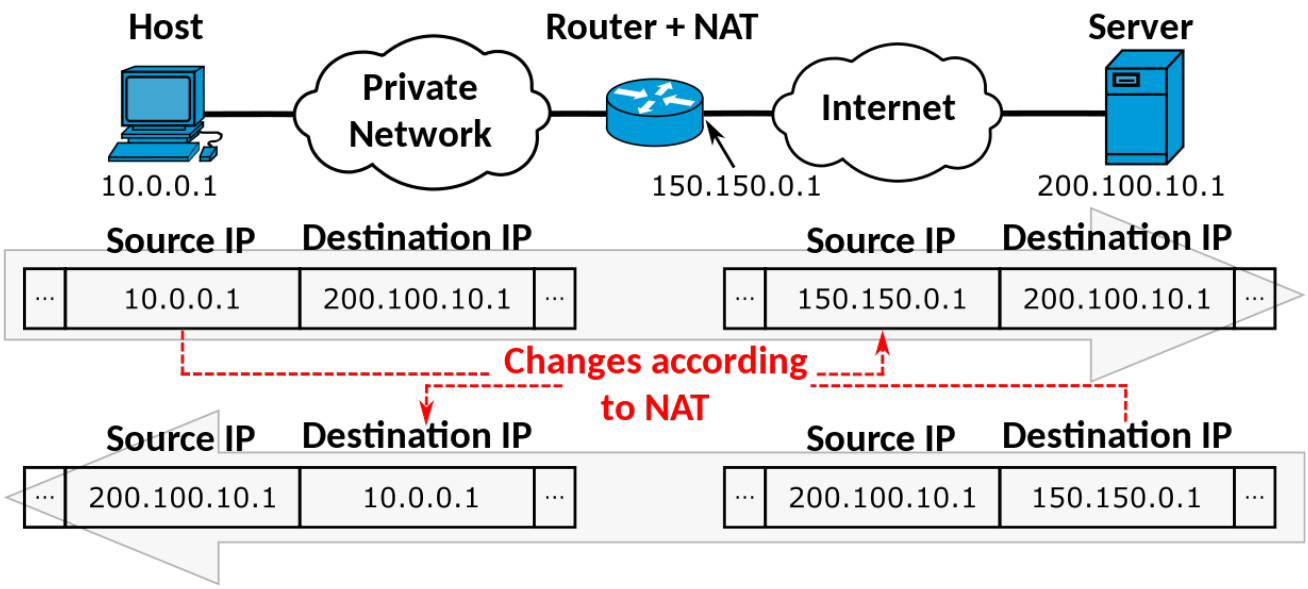
3. **Sprawdzenie stanu trunku**: Aby sprawdzić stan trunku i upewnić się, że interfejs trunk działa poprawnie, używamy komendy:

```
Switch# show interface gigabitEthernet0/1 trunk
```

Podsumowanie

W tej sekcji omówiliśmy, jak skonfigurować VLANy oraz trunk między dwoma przełącznikami Cisco Catalyst. Dzięki zastosowaniu trunku, ruch z wielu VLANów jest przesyłany przez jedno połączenie, co upraszcza zarządzanie siecią i zwiększa jej skalowalność. Prawidłowa konfiguracja VLANów umożliwia izolację ruchu, poprawia bezpieczeństwo oraz zarządzanie ruchem w sieci.

Translacja adresów w sieciach IP



Przedstawienie translacji SNAT pomiędzy siecią WAN i siecią LAN

Translacja adresów w sieciach IP (ang. Address Translation) jest techniką umożliwiającą mapowanie adresów IP z jednej przestrzeni adresowej na inną. Jest szczególnie przydatna w zarządzaniu adresacją sieciową, zapewnianiu bezpieczeństwa oraz umożliwianiu komunikacji między różnymi sieciami. Istnieje kilka metod translacji adresów, z których najczęściej wykorzystywane to: SNAT (Source Network Address Translation), DNAT (Destination Network Address Translation) oraz PAT (Port Address Translation).

SNAT (Source Network Address Translation)

SNAT jest procesem translacji adresu źródłowego w pakietach IP. Wykorzystywany jest głównie w scenariuszach, gdy urządzenia wewnętrzne (np. komputery w sieci lokalnej) chcą komunikować się z zewnętrzną siecią (np. Internetem), ale nie mają publicznych adresów IP. Dzięki SNAT, pakiety wychodzące z sieci wewnętrznej są zmieniane tak, by wyglądały, jakby pochodziły od urządzenia z publicznym adresem IP. Typowym przykładem jest router pełniący rolę bramy domyślnej, który zamienia prywatne adresy IP na jeden publiczny adres.

Praktyczne zastosowania SNAT:

- Łączenie urządzeń wewnętrznych z Internetem przy użyciu pojedynczego publicznego adresu IP.
- Maskowanie prawdziwego adresu źródłowego w celu ochrony prywatności wewnętrznych urządzeń.
- Optymalizacja wykorzystania publicznych adresów IP w przypadku ograniczonej liczby dostępnych zasobów.

DNAT (Destination Network Address Translation)

DNAT to proces translacji adresu docelowego w pakietach IP. Jest wykorzystywany, gdy pakiety przychodzące do sieci muszą być skierowane do odpowiednich urządzeń wewnętrznych, mimo iż posiadają one prywatne adresy IP. Najczęściej stosowane w scenariuszach, w których serwery wewnętrzne muszą być dostępne z Internetu, np. serwer WWW, serwer FTP, czy serwer pocztowy. Przy pomocy DNAT, router lub zapora sieciowa zmienia adresy docelowe pakietów, przekierowując je do odpowiednich serwerów wewnętrznych.

Praktyczne zastosowania DNAT:

- Udostępnianie serwerów wewnętrznych (np. serwery WWW, FTP) w Internecie.
- Kierowanie ruchu przychodzącego na różne serwery w zależności od portu docelowego (np. HTTP na jeden serwer, FTP na inny).
- Umożliwienie komunikacji z urządzeniami posiadającymi prywatne adresy IP w środowiskach zewnętrznych.

PAT (Port Address Translation)

PAT, znane również jako *NAT z translacją portów*, to technika, w której wiele urządzeń wewnętrznych korzysta z jednego publicznego adresu IP, ale różne połączenia wychodzące są identyfikowane na podstawie numeru portu. Jest to najbardziej popularna forma NAT, stosowana w sytuacjach, gdy w sieci lokalnej znajduje się wiele urządzeń, ale tylko jeden publiczny adres IP. Każde połączenie wychodzące z sieci wewnętrznej jest przypisane do unikalnego numeru portu, co pozwala na zróżnicowanie wielu połączeń wychodzących z jednego adresu IP.

Praktyczne zastosowania PAT:

- Umożliwienie wielu urządzeniom w sieci lokalnej korzystania z jednego publicznego adresu IP.
- Efektywne zarządzanie ograniczoną liczbą publicznych adresów IP.
- Użycie w domowych routerach, gdzie wiele urządzeń korzysta z internetu poprzez jeden publiczny adres IP.

DMZ (Demilitarized Zone)

DMZ, czyli *strefa zdemilitaryzowana*, to segment sieci, który znajduje się pomiędzy zewnętrzną siecią (np. Internetem) a wewnętrzną siecią korporacyjną. Jest to specjalnie wydzielona strefa, w której znajdują się serwery, które muszą być dostępne zarówno z sieci wewnętrznej, jak i z Internetu, np. serwery WWW, FTP, czy pocztowe. Celem DMZ jest zwiększenie bezpieczeństwa, ponieważ pozwala na oddzielenie serwerów, które muszą być dostępne z zewnątrz, od wewnętrznej sieci, chroniąc ją przed bezpośrednim dostępem z Internetu.

Praktyczne zastosowania DMZ:

- Izolowanie serwerów dostępnych z Internetu (np. serwery WWW, serwery pocztowe) od wewnętrznej sieci firmowej.
- Zwiększenie bezpieczeństwa organizacji poprzez segmentację ruchu sieciowego.
- Zapewnienie kontroli dostępu do wrażliwych zasobów, ograniczając dostęp do serwerów wewnętrznych z zewnątrz.

Podsumowanie

Translacja adresów w sieciach IP stanowi niezbędny element w zarządzaniu ruchem sieciowym, zapewnianiu bezpieczeństwa oraz efektywnym wykorzystaniu dostępnych zasobów adresowych. Techniki takie jak SNAT, DNAT, PAT oraz koncepcja DMZ pozwalają na skuteczną kontrolę dostępu oraz zarządzanie komunikacją między różnymi segmentami sieci. Poprzez zastosowanie tych technologii, możliwe jest zarówno zapewnienie prywatności urządzeń wewnętrznych, jak i udostępnienie zasobów sieciowych na zewnątrz w sposób bezpieczny i kontrolowany.

Protokoły warstwy aplikacji

Protokół HTTP

Protokół HTTP (HyperText Transfer Protocol) jest protokołem komunikacyjnym, który umożliwia przesyłanie danych pomiędzy klientem a serwerem w internecie. HTTP działa w oparciu o model żądanie-odpowiedź, gdzie klient (zwykle przeglądarka) wysyła zapytanie, a serwer odpowiada danymi, najczęściej w postaci stron HTML.

Metoda GET

Metoda GET jest jedną z najczęściej stosowanych metod w protokole HTTP. Służy do przesyłania żądań od klienta do serwera, w celu uzyskania danych. Dane są przesyłane w adresie URL, co oznacza, że ich długość jest ograniczona. GET jest metodą bezpieczną i idempotentną, co oznacza, że wielokrotne wysłanie tego samego żądania nie powinno zmieniać stanu serwera.

Metoda POST

Metoda POST jest używana do przesyłania danych do serwera, zazwyczaj w celu zapisania tych danych lub wykonania operacji zmieniającej stan serwera. W przeciwieństwie do GET, dane są przesyłane w ciele żądania, co pozwala na przesyłanie większej ilości danych. Metoda POST nie jest idempotentna, ponieważ wielokrotne wysłanie tego samego żądania może zmienić stan serwera.

Poczta elektroniczna

Poczta elektroniczna (email) jest jednym z najstarszych i najczęściej używanych sposobów komunikacji w Internecie. Protokóły związane z pocztą elektroniczną to m.in. SMTP (Simple Mail Transfer Protocol), POP3 (Post Office Protocol 3) oraz IMAP (Internet Message Access Protocol). SMTP jest używany do wysyłania wiadomości, natomiast POP3 i IMAP służą do odbierania i przechowywania wiadomości na serwerze.

Protokół FTP

Protokół FTP (File Transfer Protocol) służy do przesyłania plików pomiędzy komputerami w sieci. Umożliwia zarówno pobieranie, jak i wysyłanie plików na serwer. FTP działa w modelu klient-serwer i oferuje różne tryby, takie jak aktywny i pasywny, które różnią się sposobem nawiązywania połączenia. FTP może używać uwierzytelniania przy pomocy loginu i hasła, ale jest również dostępny w wersji anonimowej.

Protokół SSH

Protokół SSH (Secure Shell) jest używany do bezpiecznego zdalnego logowania i wykonywania poleceń na zdalnym serwerze. Zapewnia poufność i integralność przesyłanych danych, wykorzystując kryptografię. SSH jest szeroko stosowany do administracji systemami, zdalnego dostępu do serwerów oraz przesyłania plików (wraz z protokołem SFTP).

Protokół DNS

Protokół DNS (Domain Name System) jest odpowiedzialny za tłumaczenie nazw domenowych na adresy IP. Kiedy użytkownik wpisuje nazwę domeny w przeglądarkę, zapytanie jest wysyłane do serwera DNS, który odpowiada odpowiednim adresem IP. DNS umożliwia korzystanie z łatwych do zapamiętania nazw, zamiast musieć posługiwać się numerami IP.

Protokół DHCP

Protokół DHCP (Dynamic Host Configuration Protocol) umożliwia automatyczne przydzielanie adresów IP urządzeniom w sieci. DHCP pozwala na centralne zarządzanie adresami IP, dzięki czemu urządzenia mogą otrzymywać konfigurację sieciową bez konieczności ręcznego wprowadzania tych ustawień. Protokół ten jest powszechnie stosowany w sieciach lokalnych.

Zestawienie protokołów

Zestawienie protokołów warstwy aplikacji z portami

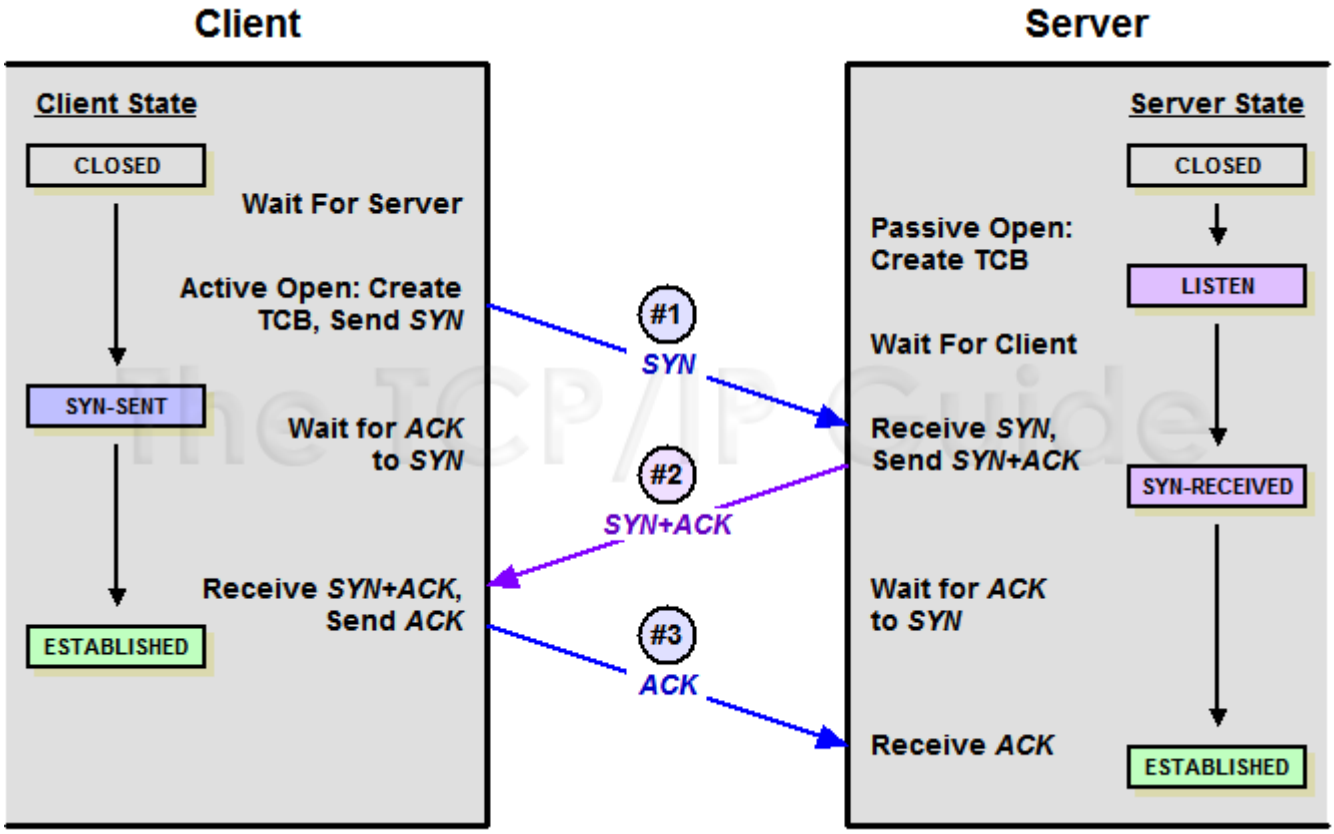
Protokół	Opis	Port
HTTP	Protokół komunikacyjny w internecie do przesyłania stron www	80
HTTPS	Protokół HTTP z szyfrowaniem SSL/TLS, zapewniający bezpieczne połączenie	443
FTP	Protokół transferu plików, umożliwiający przesyłanie plików między komputerami	21
SFTP	Bezpieczny protokół FTP, wykorzystujący SSH do szyfrowania transmisji	22
SSH	Protokół zdalnego logowania, zapewniający bezpieczny dostęp do systemów	22
SMTP	Protokół wysyłania poczty elektronicznej	25
POP3	Protokół odbierania poczty z serwera, pobierając ją na urządzenie użytkownika	110
IMAP	Protokół odbierania poczty elektronicznej z serwera, umożliwiający synchronizację	143
DNS	Protokół tłumaczenia nazw domenowych na adresy IP	53
DHCP	Protokół przydzielania adresów IP w sieci lokalnej	67 (serwer), 68 (klient)

Protokoły warstwy transportowej

Nagłówek TCP

Nagłówek TCP (Transmission Control Protocol) zawiera informacje niezbędne do nawiązania i utrzymania połączenia, jak również do zapewnienia niezawodności transmisji danych. Składa się z różnych pól, takich jak numer portu źródłowego, numer portu docelowego, numer sekwencyjny, numer potwierdzenia, długość nagłówka, flagi, okno odbiorcze oraz suma kontrolna. Nagłówek TCP może być rozszerzany o dodatkowe opcje, takie jak np. czas życia połączenia.

Uzgadnianie trój-etapowe (ang. three way handshake)



Wizualizacja uzgadniania trój-etapowego

Uzgadnianie trój-etapowe jest procesem wykorzystywanym w protokole TCP do nawiązywania połączenia między klientem a serwerem. Proces ten składa się z trzech kroków:

1. **SYN**: Klient wysyła pakiet SYN (synchronizacja) do serwera, informując go o chęci nawiązania połączenia.
2. **SYN-ACK**: Serwer odpowiada pakietem SYN-ACK (synchronizacja-acknowledgment), potwierdzając otrzymanie pakietu SYN.
3. **ACK**: Klient wysyła pakiet ACK (acknowledgment), potwierdzając otrzymanie pakietu SYN-ACK i finalizując proces uzgadniania.

Po zakończeniu tej procedury połączenie jest nawiązane i można rozpocząć przesyłanie danych.

Okno TCP

Okno TCP (ang. TCP Window) to mechanizm, który pozwala na kontrolowanie ilości danych, które mogą być przesyłane bez oczekiwania na potwierdzenie. Jest to część algorytmu sterowania przepływem w TCP. Okno jest dynamicznie dostosowywane w zależności od warunków sieciowych i obciążenia, umożliwiając optymalizację przesyłania danych. Wartość okna wskazuje, ile danych może być wysłanych przez nadawcę przed oczekiwaniem na potwierdzenie odbioru.

Protokół UDP

Protokół UDP (User Datagram Protocol) jest protokołem warstwy transportowej, który zapewnia

niestrukturalne, bezpołączeniowe przesyłanie danych. W przeciwieństwie do TCP, UDP nie gwarantuje dostarczenia pakietów, nie zapewnia kontroli błędów ani mechanizmów retransmisji. Dzięki temu jest szybszy i mniej obciążający, ale mniej niezawodny. UDP jest wykorzystywane w aplikacjach, gdzie szybkość jest bardziej istotna niż niezawodność, np. w transmisjach strumieniowych czy grach online.

Polecenie NETSTAT

Polecenie NETSTAT (Network Statistics) jest narzędziem diagnostycznym, które pozwala na wyświetlanie informacji o połączeniach sieciowych, statystykach protokołów, tabeli routingu i interfejsach sieciowych. Może być używane do monitorowania aktywnych połączeń TCP/UDP, sprawdzania stanu gniazd sieciowych, a także do analizy problemów związanych z siecią. Przykładowe użycie:

- `netstat -a` – wyświetla wszystkie aktywne połączenia i porty nasłuchujące.
- `netstat -t` – pokazuje tylko połączenia TCP.
- `netstat -u` – pokazuje tylko połączenia UDP.
- `netstat -n` – wyświetla adresy IP i numery portów w formacie numerycznym.

Rodzaje routingu w sieciach IPv4

Routing w sieciach IPv4 polega na określaniu optymalnych ścieżek przesyłu pakietów między sieciami. W praktyce wyróżniamy dwa główne typy routingu: **routing statyczny** oraz **routing dynamiczny**.

Routing statyczny

Routing statyczny polega na ręcznym konfigurowaniu tras przez administratora sieci. Każda trasa jest wpisywana do tablicy routingu i pozostaje niezmieniona, dopóki administrator jej nie zmodyfikuje.

Charakterystyka i zalety:

- **Prostota konfiguracji:** W małych lub prostych sieciach, gdzie topologia jest stała, ręczne wpisywanie tras jest łatwe do zarządzania.
- **Brak narzutu protokołu:** Statyczne trasy nie generują dodatkowego ruchu związanego z wymianą informacji między routerami.
- **Pełna kontrola:** Administrator precyzyjnie określa, przez które łącza mają przechodzić pakiety, co może być przydatne w przypadku specyficznych wymagań dotyczących bezpieczeństwa lub polityki routingu.

Wady:

- **Brak automatycznej adaptacji:** W przypadku awarii łącza lub zmiany topologii, trasy nie są aktualizowane automatycznie – niezbędna jest interwencja administratora.
- **Niska skalowalność:** W dużych sieciach ręczne zarządzanie wieloma trasami jest pracochłonne i podatne na błędy.

Przykład zastosowania:

W niewielkiej sieci biurowej lub w środowisku testowym, gdzie topologia rzadko ulega zmianie, routing statyczny jest wystarczający. Przykładowa konfiguracja na routerze Cisco:

```
ip route 192.168.2.0 255.255.255.0 192.168.1.2
```

Tutaj ruch do sieci 192.168.2.0/24 jest kierowany przez następny skok o adresie 192.168.1.2.

Routing dynamiczny

Routing dynamiczny wykorzystuje specjalne protokoły do automatycznej wymiany informacji o topologii sieci między routerami. Dzięki temu każdy router samodzielnie buduje i aktualizuje swoją tablicę routingu, reagując na zmiany w sieci.

Charakterystyka i zalety:

- **Automatyczna aktualizacja:** Routery komunikują się między sobą, wymieniając informacje o dostępnych trasach. W razie awarii łącza lub zmiany topologii, trasy są aktualizowane automatycznie.
- **Skalowalność:** Dynamiczne protokoły są lepiej przystosowane do dużych, złożonych sieci, gdzie ręczne zarządzanie trasami byłoby niepraktyczne.
- **Optymalizacja tras:** Protokoły dynamicznego routingu (np. OSPF, EIGRP) potrafią wyznaczać najkrótsze lub najbardziej efektywne ścieżki na podstawie różnych metryk (koszt, przepustowość, opóźnienie).

Wady:

- **Narzut sieciowy:** Wymiana informacji między routerami generuje dodatkowy ruch, co może wpływać na zasoby (CPU, pamięć).
- **Czas konwergencji:** Po zmianach w topologii sieci może wystąpić chwilowy okres, podczas którego tablice routingu nie są zsynchronizowane, co może prowadzić do krótkotrwałych pętli routingu lub utraty pakietów.

Przykłady protokołów dynamicznego routingu:

- **RIP (Routing Information Protocol):** Protokół oparty na algorytmie wektora odległości. Używa liczby przeskoków (hop count) jako miary odległości, z maksymalnym limitem 15 przeskoków. Ze względu na prostotę jest stosowany w małych sieciach, lecz jego ograniczenia (wolna konwergencja, ograniczenie liczby przeskoków) czynią go mniej odpowiednim dla większych sieci.
- **OSPF (Open Shortest Path First):** Protokół typu link-state, który wykorzystuje algorytm Dijkstry do wyznaczania najkrótszych ścieżek. OSPF umożliwia podział sieci na obszary (areas), co poprawia skalowalność oraz szybkość konwergencji. Znajduje zastosowanie w dużych sieciach korporacyjnych.
- **EIGRP (Enhanced Interior Gateway Routing Protocol):** Protokół hybrydowy (proprietary, opracowany przez Cisco), który łączy cechy protokołów wektora odległości i stanu łącza. Charakteryzuje się szybką konwergencją i efektywnym wykorzystaniem pasma, co czyni go popularnym w sieciach opartych na sprzęcie Cisco.
- **BGP (Border Gateway Protocol):** Protokół typu path vector używany głównie do routingu między autonomicznymi systemami (AS) w Internecie. Umożliwia zarządzanie politykami routingu oraz obsługę bardzo rozległych sieci, stanowiąc kręgosłup globalnego Internetu.

Przykład zastosowania:
W dużych korporacyjnych sieciach wewnętrznych często stosuje się OSPF lub EIGRP, które dynamicznie reagują na zmiany w infrastrukturze, automatycznie aktualizując tablice routingu. Natomiast w przypadku łączenia sieci między różnymi dostawcami Internetu wykorzystywany jest BGP, który umożliwia wymianę tras na poziomie globalnym i zarządzanie politykami routingu między AS.

Podsumowanie:
Wybór pomiędzy routingiem statycznym a dynamicznym zależy od wielkości, złożoności i wymagań danej sieci. Routing statyczny oferuje prostotę i pełną kontrolę, lecz nie reaguje automatycznie na zmiany w sieci. Routing dynamiczny, choć bardziej złożony i obciążający zasoby, umożliwia automatyczną adaptację do zmian topologii, co jest kluczowe w dużych i dynamicznych środowiskach sieciowych.

RIP (Routing Information Protocol)

RIP jest jednym z najstarszych protokołów routingu dynamicznego, opartym na algorytmie wektora odległości. Jego główne cechy to:

- **Metryka:** Liczba przeskoków (hop count) – trasa jest wybierana na podstawie najmniejszej liczby przeskoków, przy czym maksymalna dozwolona wartość wynosi 15 (wartość 16 oznacza trasę niedostępną).
- **Aktualizacje:** Routery wysyłają pełne tablice routingu co 30 sekund. Aktualizacje te mogą być również wywołane zdarzeniami (tzw. aktualizacje wyzwolone) w przypadku wykrycia zmian.
- **Mechanizmy zapobiegające pętlom:** RIP stosuje techniki takie jak *split horizon* (nie wysyła informacji o trasie z powrotem na interfejs, z którego została odebrana), *poison reverse* (ogłasza trasę jako niedostępną) oraz *hold-down timers* (opóźnia przyjmowanie nowych informacji o trasach), aby ograniczyć powstawanie pętli routingu.
- **Zastosowanie:** Ze względu na prostotę i niewielki narzut obliczeniowy, RIP nadaje się do małych i prostych sieci, jednak ograniczenie liczby przeskoków oraz wolna konwergencja sprawiają, że nie jest optymalny dla dużych środowisk.

OSPF (Open Shortest Path First)

OSPF to protokół routingu stanu łącza (link-state), który działa wewnątrz jednego systemu autonomicznego. Jego działanie opiera się na następujących zasadach:

- **Link-State:** Każdy router zbiera informacje o stanie swoich interfejsów oraz o sąsiadach. Te informacje są rozsyłane w postaci *Link State Advertisements* (LSA) do wszystkich routerów w obrębie danego obszaru.
- **Algorytm Dijkstry:** Po otrzymaniu LSAs każdy router buduje kompletną mapę topologii sieci i za pomocą algorytmu Dijkstry wyznacza najkrótsze ścieżki do wszystkich węzłów.
- **Hierarchia i obszary:** OSPF umożliwia podział sieci na obszary (areas), co pozwala ograniczyć rozmiar bazy LSAs i zwiększa skalowalność całej sieci. Obszar 0 (Backbone) pełni rolę rdzenia, do którego muszą być przyłączone wszystkie inne obszary.
- **Optymalizacja transmisji:** W sieciach typu broadcast (np. Ethernet) OSPF wyznacza routera wyznaczonego (Designated Router, DR) oraz zapasowego (Backup Designated Router, BDR) w celu ograniczenia liczby wymienianych komunikatów.

- **Zastosowanie:** OSPF jest szeroko stosowany w dużych sieciach korporacyjnych i kampusowych, gdzie wymagana jest szybka konwergencja oraz efektywne zarządzanie rozległą topologią.

EIGRP (Enhanced Interior Gateway Routing Protocol)

EIGRP to zaawansowany protokół wektora odległości, często określany jako protokół hybrydowy, ponieważ łączy cechy protokołów wektora odległości i stanu łącza. Jego działanie charakteryzuje się:

- **Algorytm DUAL:** EIGRP wykorzystuje *Diffusing Update Algorithm* (DUAL) do obliczania najlepszych tras oraz szybkiego reagowania na zmiany topologii, co pozwala na minimalizację czasu konwergencji.
- **Metryka złożona:** Obliczenie metryki w EIGRP opiera się na kilku parametrach, takich jak przepustowość, opóźnienie, niezawodność, obciążenie oraz czas transmisji. Pozwala to na bardziej precyzyjne określenie jakości trasy.
- **Wymiana aktualizacji:** EIGRP wysyła jedynie przyrostowe aktualizacje (zmiany w tablicy routingu) zamiast pełnych tablic, co zmniejsza narzut sieciowy.
- **Szybka konwergencja:** Dzięki mechanizmowi DUAL oraz możliwości szybkiego wyliczania tras zapasowych, EIGRP osiąga bardzo krótki czas konwergencji.
- **Zastosowanie:** EIGRP jest popularny w sieciach opartych na urządzeniach Cisco, gdzie liczy się zarówno szybkość reakcji na zmiany, jak i efektywne wykorzystanie zasobów.

BGP (Border Gateway Protocol)

BGP jest protokołem typu *path vector* i stanowi podstawowy mechanizm routingu między systemami autonomicznymi (AS) w Internecie. Jego działanie opiera się na następujących zasadach:

- **Path Vector:** BGP nie korzysta z tradycyjnej metryki, lecz utrzymuje pełną ścieżkę (lista AS, przez które trasa przebiega) dla każdej dostępnej trasy. Dzięki temu możliwe jest wdrażanie złożonych polityk routingu.
- **Transport TCP:** Komunikacja między routerami BGP odbywa się poprzez sesje TCP (port 179), co zapewnia niezawodny przesył komunikatów.
- **Atrybuty tras:** W procesie wyboru najlepszej ścieżki BGP uwzględnia szereg atrybutów, takich jak AS-PATH, NEXT-HOP, LOCAL-PREFERENCE, MED (Multi Exit Discriminator) oraz inne. Te atrybuty pozwalają na szczegółowe sterowanie ruchem oraz implementację polityk routingu.
- **Stabilność i skalowalność:** BGP został zaprojektowany z myślą o bardzo dużej skali, dlatego stosuje mechanizmy ograniczające fluktuacje tras oraz zapewniające stabilność globalnego routingu.
- **Zastosowanie:** BGP jest kluczowy w łączeniu różnych systemów autonomicznych, dlatego wykorzystywany jest przez dostawców usług internetowych (ISP) oraz duże organizacje do wymiany informacji routingu na poziomie globalnym.

Konfiguracja routingu statycznego i

dynamicznego na routerach Cisco

W tej sekcji przedstawimy konfigurację różnych rodzajów routingu pomiędzy dwoma routerami Cisco. Omówimy konfigurację routingu statycznego oraz dynamicznego z wykorzystaniem protokołów RIP, OSPF, EIGRP i BGP.

Routing statyczny

Routing statyczny jest najprostszym rodzajem routingu, w którym administrator ręcznie definiuje trasy w tablicy routingu. Jest to rozwiązanie używane w małych sieciach, gdzie zmiana topologii jest rzadkością.

Konfiguracja routingu statycznego pomiędzy dwoma routerami

Załóżmy, że mamy dwa routery (Router1 i Router2) z następującą topologią sieciową:

- Router1: IP 192.168.1.1/24, interfejs 192.168.1.1 łączący Router1 z Router2
- Router2: IP 192.168.1.2/24, interfejs 192.168.1.2 łączący Router2 z Router1
- Router2 ma także sieć 192.168.2.0/24, która jest dostępna tylko przez Router2.

Aby router1 mógł wysyłać ruch do sieci 192.168.2.0/24, musi mieć dodaną trasę statyczną:

```
Router1# configure terminal
Router1(config)# ip route 192.168.2.0 255.255.255.0 192.168.1.2
```

Ta komenda dodaje trasę statyczną na Router1, wskazując, że ruch do sieci 192.168.2.0/24 powinien być kierowany przez Router2 (adres 192.168.1.2).

Aby Router2 mógł wysyłać ruch do sieci 192.168.1.0/24, musi dodać trasę statyczną w następujący sposób:

```
Router2# configure terminal
Router2(config)# ip route 192.168.1.0 255.255.255.0 192.168.1.1
```

Routing dynamiczny

Routing dynamiczny pozwala routerom na automatyczne wymienianie informacji o trasach. Istnieje wiele protokołów routingu dynamicznego, które mogą być używane w sieci, a najczęściej wykorzystywane to RIP, OSPF, EIGRP i BGP.

Routing z protokołem RIP

RIP (Routing Information Protocol) to jeden z najstarszych protokołów routingu, który jest oparty na odległości. RIP używa liczby skoków (hop count) jako metryki, z maksymalną liczbą skoków wynoszącą 15.

Konfiguracja RIP na Router1 i Router2

```
Router1# configure terminal
Router1(config)# router rip
Router1(config-router)# version 2
Router1(config-router)# network 192.168.1.0
Router1(config-router)# network 192.168.2.0
Router1(config-router)# exit
```

Na Router2, konfiguracja RIP będzie wyglądać podobnie:

```
Router2# configure terminal
Router2(config)# router rip
Router2(config-router)# version 2
Router2(config-router)# network 192.168.1.0
Router2(config-router)# network 192.168.2.0
Router2(config-router)# exit
```

Po konfiguracji RIP routery będą wymieniały informacje o trasach, automatycznie ucząc się o dostępnych sieciach.

Routing z protokołem OSPF

OSPF (Open Shortest Path First) jest protokołem routingu stanu łącza, który jest bardziej zaawansowany i skalowalny niż RIP. OSPF używa algorytmu Dijkstry do obliczania najkrótszej ścieżki do celu.

Konfiguracja OSPF na Router1 i Router2

Aby skonfigurować OSPF, musimy zdefiniować proces OSPF oraz przypisać sieci do odpowiednich obszarów OSPF.

Na Router1:

```
Router1# configure terminal
Router1(config)# router ospf 1
Router1(config-router)# network 192.168.1.0 0.0.0.255 area 0
Router1(config-router)# network 192.168.2.0 0.0.0.255 area 0
Router1(config-router)# exit
```

Na Router2:

```
Router2# configure terminal
Router2(config)# router ospf 1
Router2(config-router)# network 192.168.1.0 0.0.0.255 area 0
Router2(config-router)# network 192.168.2.0 0.0.0.255 area 0
Router2(config-router)# exit
```

Po tej konfiguracji oba routery będą wymieniały informacje o trasach przy użyciu protokołu OSPF.

Routing z protokołem EIGRP

EIGRP (Enhanced Interior Gateway Routing Protocol) to protokół routingu firmy Cisco, który łączy cechy protokołów wektora odległości i stanu łącza. EIGRP jest bardziej efektywny niż RIP i ma lepszą skalowalność.

Konfiguracja EIGRP na Router1 i Router2

Na Router1:

```
Router1# configure terminal
Router1(config)# router eigrp 100
Router1(config-router)# network 192.168.1.0 0.0.0.255
Router1(config-router)# network 192.168.2.0 0.0.0.255
Router1(config-router)# exit
```

Na Router2:

```
Router2# configure terminal
Router2(config)# router eigrp 100
Router2(config-router)# network 192.168.1.0 0.0.0.255
Router2(config-router)# network 192.168.2.0 0.0.0.255
Router2(config-router)# exit
```

Po skonfigurowaniu EIGRP routery wymienią informacje o trasach i będą wybierały najoptymalniejszą ścieżkę do docelowej sieci.

Routing z protokołem BGP

BGP (Border Gateway Protocol) jest protokołem routingu międzydomenowego (EGP), który jest wykorzystywany głównie w Internecie do wymiany tras pomiędzy różnymi systemami autonomicznymi.

Konfiguracja BGP na Router1 i Router2

Załóżmy, że Router1 jest w systemie autonomicznym (AS) 100, a Router2 w AS 200.

Na Router1:

```
Router1# configure terminal
Router1(config)# router bgp 100
Router1(config-router)# neighbor 192.168.1.2 remote-as 200
Router1(config-router)# network 192.168.1.0 mask 255.255.255.0
Router1(config-router)# exit
```

Na Router2:

```
Router2# configure terminal
Router2(config)# router bgp 200
Router2(config-router)# neighbor 192.168.1.1 remote-as 100
Router2(config-router)# network 192.168.2.0 mask 255.255.255.0
Router2(config-router)# exit
```

W przypadku BGP, trasy są wymieniane pomiędzy różnymi systemami autonomicznymi, co jest szczególnie przydatne w dużych sieciach takich jak Internet.

Podsumowanie

W tej sekcji zaprezentowaliśmy przykłady konfiguracji różnych typów routingu na routerach Cisco, w tym:

- Routing statyczny
- Routing dynamiczny z protokołami RIP, OSPF, EIGRP i BGP

Każdy z tych protokołów ma swoje zastosowanie w różnych scenariuszach, a wybór odpowiedniego protokołu zależy od rozmiaru sieci, wymagań dotyczących wydajności i skalowalności, a także specyfiki topologii sieciowej.

Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP) to protokół sieciowy stosowany do automatycznego przydzielania adresów IP oraz innych parametrów konfiguracyjnych hostom w sieci. Dzięki DHCP eliminowana jest potrzeba ręcznej konfiguracji adresów IP na każdym urządzeniu.

Konfiguracja DHCP na systemie Linux

Na systemie Linux, popularnym serwerem DHCP jest `isc-dhcp-server`. Aby go skonfigurować, należy wykonać następujące kroki:

1. Instalacja serwera DHCP:

```
sudo apt update
```

```
sudo apt install isc-dhcp-server
```

2. Konfiguracja pliku `/etc/dhcp/dhcpd.conf`:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.100 192.168.1.200;
    option routers 192.168.1.1;
    option domain-name-servers 8.8.8.8, 8.8.4.4;
    option broadcast-address 192.168.1.255;
    default-lease-time 600;
    max-lease-time 7200;
}
```

3. Restart usługi DHCP:

```
sudo systemctl restart isc-dhcp-server
```

Konfiguracja DHCP na routerach Cisco

Na routerach Cisco DHCP konfiguruje się w trybie konfiguracji globalnej. Przykładowa konfiguracja:

```
Router(config)# ip dhcp excluded-address 192.168.1.1 192.168.1.50
Router(config)# ip dhcp pool LAN
Router(dhcp-config)# network 192.168.1.0 255.255.255.0
Router(dhcp-config)# default-router 192.168.1.1
Router(dhcp-config)# dns-server 8.8.8.8 8.8.4.4
Router(dhcp-config)# lease 7
```

Po zakończeniu konfiguracji można sprawdzić przydzielone adresy poleceniem:

```
Router# show ip dhcp binding
```

Komunikacja w protokole DHCP

Proces przydzielania adresu IP przez serwer DHCP składa się z czterech głównych etapów:

- **DHCPDISCOVER** - Klient wysyła rozgłoszeniową wiadomość w celu znalezienia dostępnych serwerów DHCP.
- **DHCPOFFER** - Serwer DHCP odpowiada, oferując dostępny adres IP oraz inne parametry konfiguracyjne.
- **DHCPREQUEST** - Klient wybiera jedną z ofert i wysyła prośbę o przypisanie adresu IP.
- **DHCPACK** - Serwer potwierdza przydzielenie adresu, a klient może go używać.

Cały proces przebiega w modelu klient-serwer, a komunikacja opiera się na pakietach UDP: port 67 (serwer) i port 68 (klient).

System DNS (ang. domain name system)

Podstawy działania DNS

DNS (Domain Name System) jest systemem, który tłumaczy ludzkie nazwy domenowe, takie jak `www.example.com`, na adresy IP, które są wykorzystywane przez urządzenia w sieci. W skrócie, DNS działa jak książka telefoniczna dla internetu, umożliwiając urządzeniom odnalezienie serwisów internetowych na podstawie nazw domenowych.

Podstawowe elementy działania DNS obejmują:

- **Rekordy DNS**: Są to informacje przechowywane w bazach danych serwerów DNS, które określają, jakie adresy IP są przypisane do danej nazwy domenowej.
- **Serwery DNS**: Serwery odpowiedzialne za przechowywanie rekordów i rozwiązywanie zapytań DNS. Istnieje kilka typów serwerów DNS, w tym serwery główne (root), serwery nazw (nameservers) oraz serwery podrzędne.
- **Zapytania DNS**: Kiedy urządzenie (np. komputer) chce uzyskać adres IP powiązany z określoną nazwą domeny, wysyła zapytanie do serwera DNS. Odpowiedź może zawierać pełny adres IP lub wskazówki do dalszych zapytań.

Rodzaje rekordów DNS

Rekordy DNS są kluczowe dla prawidłowego funkcjonowania systemu nazw domenowych. Istnieje wiele typów rekordów, z których każdy pełni określoną funkcję.

Rekord A (Address Record)

Rekord typu A (Address) mapuje nazwę domeny na adres IPv4. Jest to najczęściej używany rekord w DNS, który pozwala na przekształcenie nazw domenowych na numery IP.

```
    example.com.    IN    A    192.168.1.1
```

Wyjaśnienie: Rekord ten mówi, że domena `example.com` jest przypisana do adresu IPv4 `192.168.1.1`.

Rekord AAAA (IPv6 Address Record)

Rekord typu AAAA mapuje nazwę domeny na adres IPv6. Jest to odpowiednik rekordu A w systemie IPv6.

```
    example.com.    IN    AAAA    2001:0db8:85a3:0000:0000:8a2e:0370:7334
```

Wyjaśnienie: Rekord ten przypisuje domenę `example.com` do adresu IPv6 `2001:0db8:85a3:0000:0000:8a2e:0370:7334`.

Rekord MX (Mail Exchange Record)

Rekord typu MX jest używany do wskazania serwera poczty elektronicznej odpowiedzialnego za obsługę wiadomości dla danej domeny. Zawiera również priorytet, który określa preferencje dla serwerów pocztowych.

```
    example.com.    IN    MX    10 mail.example.com.
```

Wyjaśnienie: Rekord MX wskazuje, że serwer poczty dla domeny `example.com` to `mail.example.com` i ma priorytet 10.

Rekord CNAME (Canonical Name Record)

Rekord typu CNAME jest aliasem dla innej domeny. Zamiast bezpośrednio wskazywać adres IP, rekord CNAME przekierowuje zapytanie DNS do innej domeny, która zawiera rekord A lub AAAA.

```
    www.example.com.    IN    CNAME    example.com.
```

Wyjaśnienie: Rekord CNAME wskazuje, że `www.example.com` jest aliasem dla `example.com`, co oznacza, że zapytanie DNS o `www.example.com` zostanie przekierowane do `example.com`.

Rekord NS (Name Server Record)

Rekord typu NS wskazuje na serwery nazw odpowiedzialne za zarządzanie rekordami DNS danej domeny.

```
    example.com.    IN    NS    ns1.example.com.
```

Wyjaśnienie: Rekord NS mówi, że serwer `ns1.example.com` jest odpowiedzialny za zarządzanie rekordami DNS dla domeny `example.com`.

Rekord PTR (Pointer Record)

Rekord typu PTR jest używany do odwrotnego mapowania adresów IP na nazwy domenowe. Stosowany głównie w przypadku odwrotnych zapytań DNS, aby uzyskać nazwę hosta na podstawie adresu IP.

```
    1.168.192.in-addr.arpa.    IN    PTR    example.com.
```

Wyjaśnienie: Rekord PTR mówi, że adres IP `192.168.1.0` jest powiązany z nazwą domeny `example.com`.

Rekord SOA (Start of Authority Record)

Rekord typu SOA określa podstawowe informacje o strefie DNS, takie jak główny serwer nazw, kontakt do administratora strefy oraz informacje o czasach aktualizacji.

```
example.com.    IN    SOA    ns1.example.com. admin.example.com. (
2022020501 ; serial
3600       ; refresh
1800       ; retry
1209600    ; expire
86400 )    ; minimum TTL
```

Wyjaśnienie: Rekord SOA zawiera informacje o głównym serwerze nazw (`ns1.example.com`) oraz dane administratora (`admin.example.com`).

Rodzaje zapytań DNS

Zapytań DNS można dokonywać na kilka różnych sposobów, zależnie od tego, jak system zarządza zapytaniem:

Rekursywne zapytanie DNS

Rekursywne zapytanie DNS oznacza, że serwer DNS jest odpowiedzialny za całkowite rozwiązanie zapytania. W przypadku, gdy serwer DNS nie ma odpowiedzi w swojej pamięci podręcznej, podejmuje on wyszukiwanie rekordów w innych serwerach DNS, aż znajdzie odpowiedź lub stwierdzi, że domena nie istnieje.

Nie-rekursywne zapytanie DNS

Nie-rekursywne zapytanie DNS oznacza, że serwer DNS odpowiada tylko, jeśli ma odpowiedź w swojej bazie danych lub pamięci podręcznej. Jeśli nie ma odpowiedzi, nie podejmuje dalszych prób.

Podsumowanie

DNS jest podstawowym mechanizmem, który umożliwia tłumaczenie nazw domenowych na adresy IP, umożliwiając urządzeniom komunikację w sieci. Rekordy DNS są kluczowymi elementami tego procesu i obejmują różne typy, takie jak A, AAAA, MX, CNAME, NS, PTR i SOA, które pełnią różne funkcje w zależności od potrzeby. Zrozumienie działania DNS i jego rekordów jest niezbędne do prawidłowego zarządzania usługami internetowymi oraz diagnozowania problemów z połączeniami sieciowymi.

Budowanie zasad w firewallach

Firewall (zaporą sieciowa) jest urządzeniem lub oprogramowaniem, które monitoruje i kontroluje ruch

sieciowy przychodzący i wychodzący z sieci komputerowej. Firewall działa na zasadzie zestawu reguł (zasad), które określają, który ruch jest dozwolony, a który zablokowany. Budowanie zasad w firewallu polega na definiowaniu reguł, które pozwalają na kontrolowanie dostępu do zasobów sieciowych w oparciu o różne kryteria, takie jak adresy IP, porty, protokoły, czas, czy inne atrybuty pakietów.

Zasady w firewallach mogą być budowane na kilku poziomach:

- **Adres IP**: Określenie, który ruch jest dozwolony w zależności od adresu źródłowego lub docelowego pakietu.
- **Porty**: Reguły mogą bazować na numerach portów, co pozwala na kontrolowanie dostępu do określonych usług, takich jak HTTP (port 80) czy SSH (port 22).
- **Protokół**: Określenie, jaki protokół (np. TCP, UDP, ICMP) jest dozwolony w danym ruchu.
- **Stan połączenia**: Budowanie zasad na podstawie stanu połączenia, np. pozwalanie na odpowiedź tylko dla połączeń ustanowionych.
- **Interfejs sieciowy**: Określenie, z jakiego interfejsu sieciowego ma być przyjmowany lub wysyłany ruch.

W systemie Linux najpopularniejszym narzędziem do zarządzania firewallami jest ‘iptables‘. ‘iptables‘ pozwala na tworzenie i zarządzanie regułami w oparciu o różne kryteria.

Przykłady konfiguracji firewalla za pomocą iptables

Poniżej przedstawiono kilka podstawowych przykładów konfiguracji firewalla przy użyciu narzędzia ‘iptables‘ w systemie Linux:

Akceptowanie ruchu na określonym porcie (np. HTTP na porcie 80)

Aby zezwolić na ruch przychodzący na porcie 80 (HTTP), można dodać następującą regułę:

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

Wyjaśnienie:

- `-A INPUT`: Dodaj regułę do łańcucha `INPUT`, który odpowiada za ruch przychodzący.
- `-p tcp`: Określa protokół TCP.
- `–dport 80`: Oznacza port docelowy, którym jest port 80 (HTTP).
- `-j ACCEPT`: Oznacza, że ruch powinien zostać zaakceptowany.

Odrzucanie ruchu z określonego adresu IP

Aby zablokować ruch przychodzący z określonego adresu IP (np. 192.168.1.100), używamy następującej reguły:

```
sudo iptables -A INPUT -s 192.168.1.100 -j DROP
```

Wyjaśnienie:

- `-s 192.168.1.100`: Określa adres IP źródłowy, który chcemy zablokować.
- `-j DROP`: Ruch z tego adresu zostanie odrzucony.

Zezwolenie na ruch tylko z określonego interfejsu

Aby zezwolić na ruch przychodzący tylko przez określony interfejs sieciowy (np. `eth0`), używamy następującej reguły:

```
sudo iptables -A INPUT -i eth0 -j ACCEPT
```

Wyjaśnienie:

- `-i eth0`: Określa interfejs sieciowy `eth0`.
- `-j ACCEPT`: Akceptuje ruch przychodzący przez interfejs `eth0`.

Blokowanie dostępu do portu SSH (port 22) z zewnątrz

Aby zablokować dostęp do portu 22 (SSH) z zewnętrznych adresów IP, można użyć następującej reguły:

```
sudo iptables -A INPUT -p tcp --dport 22 -i eth0 -s 0.0.0.0/0 -j DROP
```

Wyjaśnienie:

- `-p tcp`: Określa protokół TCP.
- `–dport 22`: Określa port docelowy, którym jest port 22 (SSH).
- `-i eth0`: Określa interfejs sieciowy.
- `-s 0.0.0.0/0`: Oznacza, że reguła dotyczy wszystkich adresów źródłowych.
- `-j DROP`: Blokuje ruch na porcie 22.

Zezwolenie na połączenia wychodzące na wszystkich portach

Aby umożliwić połączenia wychodzące na wszystkich portach z maszyny:

```
sudo iptables -A OUTPUT -j ACCEPT
```

Wyjaśnienie:

- `-A OUTPUT`: Dodaje regułę do łańcucha `OUTPUT`, który odpowiada za ruch wychodzący.
- `-j ACCEPT`: Zezwala na wszelkie połączenia wychodzące.

Podsumowanie

Budowanie zasad w firewallach jest kluczowe dla zapewnienia bezpieczeństwa sieci komputerowych. Dzięki odpowiedniej konfiguracji ‘iptables‘ w systemie Linux można precyzyjnie kontrolować dostęp do zasobów sieciowych. Tworzenie zasad może obejmować blokowanie określonych adresów IP,

zarządzanie dostępem do usług na podstawie portów, protokołów oraz interfejsów sieciowych. Poprzez odpowiednią konfigurację, administratorzy mogą skutecznie chronić swoje sieci przed nieautoryzowanym dostępem.

Przykład szyfrowania HELLO WORLD

W tej sekcji przedstawimy krok po kroku przykład szyfrowania ciągu znaków HELLO WORLD za pomocą trzech różnych algorytmów kryptograficznych: AES, SHA oraz RSA. Każdy z tych algorytmów ma różne zastosowanie i różne mechanizmy szyfrowania, więc w naszym przykładzie zaprezentujemy operacje matematyczne, które są wykonywane na tym ciągu znaków.

Szyfrowanie AES (Advanced Encryption Standard)

AES to algorytm szyfrowania symetrycznego, co oznacza, że ten sam klucz jest używany zarówno do szyfrowania, jak i deszyfrowania danych. Aby zacząć, przekształcimy ciąg znaków HELLO WORLD na odpowiednią formę binarną, a następnie użyjemy algorytmu AES do zaszyfrowania danych.

Kroki szyfrowania AES

1. **Konwersja tekstu na dane binarne:** Ciąg znaków „HELLO WORLD” ma długość 11 znaków. Wartości ASCII dla każdego znaku są:

- H = 72
- E = 69
- L = 76
- L = 76
- O = 79
- (spacja) = 32
- W = 87
- O = 79
- R = 82
- L = 76
- D = 68

Następnie konwertujemy każdy z tych znaków na ich binarną reprezentację, tworząc ciąg bitów.

2. **Wybór klucza szyfrującego:** Załóżmy, że używamy klucza 128-bitowego:
Klucz = 0x2b7e151628aed2a6abf7158809cf4f3c (klucz 128-bitowy)

3. **Szyfrowanie danych za pomocą AES:** AES dzieli dane na bloki 128-bitowe i wykonuje operacje na każdym bloku w kilku rundach. Po kilku rundach operacji, takich jak SubBytes, ShiftRows, MixColumns i AddRoundKey, wynikowy zaszyfrowany tekst (ciphertext) będzie wyglądał mniej więcej tak: $$\text{Ciphertext} = 0x3ad77bb40d7a3660a89ecaf32466ef97$$

Podsumowanie szyfrowania AES:

AES zapewnia bezpieczeństwo poprzez szyfrowanie bloków danych za pomocą różnych operacji, w tym podmiany bajtów (SubBytes), przesunięcia wierszy (ShiftRows), mieszania kolumn (MixColumns) i dodania klucza rundy (AddRoundKey). W tym przykładzie zaszyfrowaliśmy ciąg „HELLO WORLD” za pomocą klucza 128-bitowego, uzyskując zaszyfrowany ciąg znaków.

Funkcja skrótu SHA (Secure Hash Algorithm)

SHA jest funkcją skrótu, która przekształca dane wejściowe w skrót o stałej długości. SHA-256, będący częścią rodziny SHA-2, generuje 256-bitowy skrót z dowolnych danych wejściowych.

Kroki generowania skrótu SHA-256

1. **Konwersja tekstu na dane binarne:** Ponownie zaczynamy od ciągu HELLO WORLD. Każdy znak jest reprezentowany przez swój kod ASCII, a następnie konwertowany na format binarny.

2. **Obliczanie skrótu SHA-256:** Następnie przekazujemy dane wejściowe do algorytmu SHA-256, który wykonuje operacje takie jak rozkładanie danych na bloki, dodawanie paddingu oraz wykonywanie rund obliczeniowych z użyciem funkcji mieszania i permutacji. Po zakończeniu procesu obliczania skrótu otrzymujemy wynik:
SHA-256 Hash = 0xA591A6D40BF420404A011733CFB7B190D62C65BF0
BCDA6C1C3F2A78C89E2C86E

Podsumowanie funkcji SHA:

Funkcja SHA-256 przekształca dane wejściowe HELLO WORLD w skrót o stałej długości 256 bitów. Główną rolą funkcji skrótu jest zapewnienie integralności danych — każda zmiana w danych wejściowych prowadzi do zupełnie innego skrótu, co pozwala na detekcję wszelkich manipulacji.

Szyfrowanie RSA

RSA jest algorytmem kryptografii asymetrycznej, co oznacza, że używa się dwóch kluczy: publicznego (do szyfrowania) i prywatnego (do deszyfrowania). RSA opiera się na matematyce faktoryzacji liczb pierwszych i jest szeroko stosowany do bezpiecznej wymiany kluczy.

Kroki szyfrowania RSA

1. **Generowanie kluczy:** W pierwszym kroku generujemy dwie duże liczby pierwsze p i q oraz obliczamy ich iloczyn $n = p \times q$. Z tego n obliczamy funkcję Eulera $\varphi(n)$, a następnie wybieramy publiczny wykładnik e, który jest względnie pierwszy względem $\varphi(n)$. Na koniec obliczamy prywatny wykładnik d, który spełnia równanie: $$e \times d \equiv 1 \pmod{\varphi(n)}$$

2. **Szyfrowanie wiadomości:** Ciąg „HELLO WORLD” jest najpierw przekształcany na liczbę, reprezentującą dane w formacie numerycznym (np. za pomocą kodowania ASCII). Następnie szyfrujemy wiadomość za pomocą klucza publicznego e i n z algorytmem RSA: $$C = M^e \mod n$$ gdzie M to wiadomość w postaci liczby, a C to zaszyfrowana wiadomość.

3. **Deszyfrowanie wiadomości:** Ostatecznie, używając klucza prywatnego d, deszyfrujemy wiadomość za pomocą algorytmu RSA: $$M = C^d \mod n$$ Wynikiem tego procesu jest odzyskanie oryginalnej wiadomości „HELLO WORLD”.

Podsumowanie szyfrowania RSA:

W procesie szyfrowania RSA, „HELLO WORLD” jest przekształcane w liczbę, która jest następnie szyfrowana za pomocą klucza publicznego. Dzięki właściwościom kryptografii asymetrycznej, tylko osoba posiadająca odpowiedni klucz prywatny jest w stanie odszyfrować tę wiadomość.

Podsumowanie

W tej sekcji zaprezentowaliśmy przykład szyfrowania ciągu znaków „HELLO WORLD” przy użyciu trzech różnych algorytmów kryptograficznych: AES, SHA oraz RSA. Każdy z tych algorytmów operuje na innych zasadach matematycznych i jest stosowany w różnych kontekstach kryptograficznych, zapewniając poufność, integralność i autentyczność danych.

SSH i szyfrowanie asymetryczne

SSH (Secure Shell) to protokół używany do bezpiecznej komunikacji w sieciach komputerowych, szczególnie w celu zdalnego logowania się do systemów oraz wykonywania poleceń na serwerach. SSH zapewnia poufność, integralność danych oraz autentyczność połączenia. W tej sekcji wyjaśnimy, jak działa SSH oraz jak w kontekście tego protokołu wykorzystuje się szyfrowanie asymetryczne.

Zasada działania SSH

SSH działa na zasadzie wymiany danych pomiędzy klientem a serwerem. Kiedy klient łączy się z serwerem, następuje wymiana kluczy publicznych, dzięki którym możliwe jest autentykowanie tożsamości oraz ustanowienie bezpiecznego połączenia. SSH może korzystać z kilku mechanizmów autentykacji, z których najpopularniejsze to:

- **Autentykacja na podstawie hasła** – klient wprowadza hasło w celu uwierzytelnienia.
- **Autentykacja na podstawie kluczy publicznych** – klient i serwer wymieniają klucze publiczne, aby zweryfikować tożsamość.

Po wymianie kluczy, nawiązywane jest szyfrowane połączenie z użyciem algorytmów takich jak AES, DES lub 3DES, co zapewnia poufność przesyłanych danych.

Szyfrowanie asymetryczne

Szyfrowanie asymetryczne to typ szyfrowania, w którym używa się dwóch różnych kluczy: publicznego i prywatnego. Klucz publiczny służy do szyfrowania wiadomości, a klucz prywatny – do jej odszyfrowywania. Klucze te są matematycznie powiązane, ale nie można ich obliczyć na podstawie siebie nawzajem.

Główne zasady szyfrowania asymetrycznego:

- Klucz publiczny może być udostępniany wszystkim, natomiast klucz prywatny musi być zachowany w tajemnicy.
- Wiadomość zaszyfrowana kluczem publicznym może być odszyfrowana tylko za pomocą odpowiadającego mu klucza prywatnego.
- Klucz prywatny może być użyty do podpisywania wiadomości, a klucz publiczny – do weryfikacji podpisu.

Matematyka szyfrowania asymetrycznego

Szyfrowanie asymetryczne opiera się na problemach matematycznych, które są łatwe do wykonania w jedną stronę, ale bardzo trudne do odwrócenia bez posiadania odpowiednich informacji (np. klucza prywatnego). Przykładem jest algorytm RSA, który opiera się na trudności faktoryzacji dużych liczb pierwszych.
Algorytm RSA:

1. Generowanie kluczy:
    - Wybieramy dwie duże liczby pierwsze p i q.
    - Obliczamy $n = p \cdot q$ – jest to moduł, który będzie używany do szyfrowania i odszyfrowywania.
    - Obliczamy funkcję Eulera $\varphi(n) = (p - 1) \cdot (q - 1)$.
    - Wybieramy liczbę e, która jest względnie pierwsza z $\varphi(n)$, a następnie obliczamy d, które jest odwrotnością e modulo $\varphi(n)$, tzn. $e \cdot d \equiv 1 \ (\text{mod} \ \varphi(n))$.
2. Klucze:
    - Klucz publiczny to para (e, n).
    - Klucz prywatny to para (d, n).
3. Szyfrowanie:
    - Aby zaszyfrować wiadomość m, obliczamy $c = m^e \ (\text{mod} \ n)$, gdzie m to liczba odpowiadająca tekstowi.
4. Odszyfrowywanie:
    - Aby odszyfrować wiadomość, obliczamy $m = c^d \ (\text{mod} \ n)$.

Matematyka stojąca za RSA sprawia, że faktoryzacja dużych liczb jest obliczeniowo bardzo trudna, a tym samym niemożliwe jest łatwe obliczenie klucza prywatnego d na podstawie klucza publicznego e i n, co stanowi podstawę bezpieczeństwa szyfrowania asymetrycznego.

Importowanie kluczy na serwer SSH na Ubuntu

Aby skonfigurować autentykację za pomocą kluczy publicznych na serwerze SSH, należy wykonać następujące kroki:

Generowanie pary kluczy na kliencie

Na komputerze klienckim używamy narzędzia ‘ssh-keygen‘, aby wygenerować parę kluczy (klucz publiczny i prywatny).

```
ssh-keygen -t rsa -b 2048
```

Po wykonaniu tego polecenia zostaną utworzone dwa pliki:

- `/.ssh/id_rsa` – klucz prywatny (należy go chronić).
- `/.ssh/id_rsa.pub` – klucz publiczny (można go udostępnić).

Przesyłanie klucza publicznego na serwer

Aby zaimportować klucz publiczny na serwerze, używamy polecenia ‘ssh-copy-id‘. Dzięki temu klucz publiczny zostanie dodany do pliku autoryzacji na serwerze.

```
ssh-copy-id user@server_ip
```

Po wykonaniu tego polecenia, klucz publiczny zostanie dodany do pliku `/.ssh/authorized_keys` na serwerze, co umożliwi logowanie się do serwera bez potrzeby podawania hasła.

Ręczne kopiowanie klucza publicznego

Alternatywnie, można ręcznie skopiować zawartość pliku klucza publicznego `/.ssh/id_rsa.pub` i wkleić ją do pliku `/.ssh/authorized_keys` na serwerze.

```
cat ~/.ssh/id_rsa.pub | ssh user@server_ip "mkdir -p ~/.ssh &&
cat >> ~/.ssh/authorized_keys"
```

Zmiana uprawnień na serwerze

Po dodaniu klucza publicznego, upewnij się, że pliki i katalogi mają odpowiednie uprawnienia:

```
chmod 700 ~/.ssh
chmod 600 ~/.ssh/authorized_keys
```

Podsumowanie

SSH jest kluczowym protokołem w zabezpieczonej komunikacji zdalnej, który wykorzystuje szyfrowanie asymetryczne w celu zapewnienia poufności i autentyczności. Szyfrowanie asymetryczne

opiera się na dwóch kluczach, publicznym i prywatnym, z których tylko prywatny klucz może odszyfrować dane zaszyfrowane kluczem publicznym. Dodatkowo, konfiguracja SSH na Ubuntu umożliwia bezpieczne logowanie za pomocą kluczy publicznych, co eliminuje konieczność używania haseł i zwiększa bezpieczeństwo połączeń.

IPsec i jego zastosowanie z tunelami L2TP

Co to jest IPsec?

IPsec (Internet Protocol Security) to zestaw protokołów kryptograficznych, który zapewnia bezpieczną transmisję danych w sieci IP. IPsec działa na warstwie sieciowej modelu OSI i zapewnia ochronę danych na poziomie całego ruchu IP, co umożliwia zabezpieczenie komunikacji w sieciach publicznych, takich jak Internet.

Celem IPsec jest zapewnienie:

- **Poufności (Encryption)** – poprzez szyfrowanie danych przesyłanych w sieci.
- **Integralności danych (Integrity)** – zapewnienie, że dane nie zostały zmodyfikowane w trakcie transmisji.
- **Autentyczności (Authentication)** – potwierdzenie tożsamości nadawcy i odbiorcy.
- **Odporności na powtórzenie (Anti-Replay)** – zabezpieczenie przed próbami powtórnego nadania tych samych danych.

IPsec działa poprzez zastosowanie różnych protokołów i mechanizmów kryptograficznych, takich jak szyfrowanie, klucze publiczne, oraz różne metody uwierzytelniania.

Jak działa IPsec?

IPsec oferuje dwie główne funkcje, które można stosować razem lub oddzielnie:

1. **Transport Mode** – tylko dane są szyfrowane, a nagłówki IP pozostają nietknięte.
2. **Tunnel Mode** – całe pakiety IP są szyfrowane, w tym nagłówki, co tworzy nowy nagłówek IP.

Transport Mode

W trybie transportowym IPsec szyfruje jedynie dane użytkownika, pozostawiając nagłówki IP bez zmian. Z tego trybu korzystają zazwyczaj połączenia typu point-to-point, takie jak połączenia VPN między dwoma urządzeniami.

Tunnel Mode

Tryb tunelowania jest szeroko stosowany w połączeniach VPN, gdzie całość pakietu (w tym nagłówki) jest szyfrowana, co zapewnia pełne zabezpieczenie całego ruchu sieciowego. Pakiet otrzymuje nowy

nagłówek IP, który zawiera adresy bramki VPN.

Matematyka i inżynieria stojąca za IPsec

IPsec używa różnych algorytmów kryptograficznych, w tym algorytmów szyfrowania, takich jak AES (Advanced Encryption Standard) oraz algorytmów haszujących, takich jak SHA (Secure Hash Algorithm). Kluczowym mechanizmem w IPsec jest stosowanie kluczy symetrycznych i asymetrycznych do zapewnienia poufności i autentyczności danych.

Algorytm AES

Algorytm AES, który jest jednym z najczęściej stosowanych w IPsec, jest algorytmem szyfrowania blokowego. AES działa na blokach danych o długości 128 bitów, używając kluczy o długości 128, 192 lub 256 bitów.

Szyfrowanie AES opiera się na następujących operacjach:

1. **SubBytes** – podmiana bajtów w tabeli S-Box.
2. **ShiftRows** – przesunięcie wierszy w tabeli stanu.
3. **MixColumns** – mieszanie kolumn w tabeli stanu.
4. **AddRoundKey** – dodanie (XOR) klucza rundy do stanu.

AES działa w określonej liczbie rund w zależności od długości klucza:

- 10 rund dla klucza 128-bitowego,
- 12 rund dla klucza 192-bitowego,
- 14 rund dla klucza 256-bitowego.

Algorytm SHA

Algorytmy SHA, takie jak SHA-1 i SHA-2, są używane w IPsec do zapewnienia integralności danych. SHA działa poprzez obliczenie skrótu (hash) wiadomości, który jest unikalnym odzwierciedleniem danych. Najczęściej wykorzystywaną wersją jest SHA-256, która generuje skrót o długości 256 bitów.

Zastosowanie IPsec z tunelami L2TP

L2TP (Layer 2 Tunneling Protocol) to protokół, który samodzielnie nie zapewnia szyfrowania ani autentyczności, ale jest często używany w połączeniu z IPsec do zapewnienia bezpiecznego tunelowania danych w sieci. L2TP jest protokołem, który tworzy tunel między dwoma punktami w sieci i umożliwia przesyłanie pakietów IP przez ten tunel.

Zasada działania tunelu L2TP z IPsec

Połączenie L2TP/IPsec działa na zasadzie:

1. Tworzenie tunelu L2TP, który działa na warstwie 2 (łącze danych) i umożliwia przekazywanie danych przez połączenie sieciowe.
2. Szyfrowanie danych i autentykacja za pomocą IPsec, który działa na warstwie 3 (sieciowa) i zapewnia bezpieczeństwo przesyłanych informacji.

Taki tunel jest bezpieczny, ponieważ:

- L2TP tworzy tunel, przez który dane są przesyłane.
- IPsec zapewnia poufność (szyfrowanie), integralność i autentyczność danych.

Przykład konfiguracji tunelu L2TP/IPsec

Aby skonfigurować tunel L2TP z IPsec, musisz wykonać następujące kroki:

1. Skonfiguruj serwer VPN, aby obsługiwał protokół L2TP.
2. Skonfiguruj serwer VPN, aby używał IPsec do szyfrowania danych.
3. Wymień klucze szyfrowania (np. Pre-Shared Key - PSK) pomiędzy klientem i serwerem.
4. Skonfiguruj klienta, aby łączył się z serwerem VPN używając L2TP/IPsec.

Zastosowanie IPsec i L2TP w praktyce

Połączenie L2TP/IPsec jest szeroko stosowane w:

- Zdalnych połączeniach VPN (Virtual Private Network), gdzie użytkownicy mogą bezpiecznie łączyć się z siecią firmową.
- Wirtualnych prywatnych sieciach (VPN) w sieciach mobilnych, zapewniających ochronę danych podczas korzystania z niezabezpieczonych sieci publicznych (np. Wi-Fi).
- W sieciach korporacyjnych, w celu połączenia oddziałów firmy w bezpieczny sposób przez publiczny Internet.

Podsumowanie

IPsec i L2TP stanowią silne połączenie, które zapewnia bezpieczeństwo transmisji danych w sieciach komputerowych. IPsec chroni dane za pomocą zaawansowanego szyfrowania i autentykacji, natomiast L2TP umożliwia tworzenie bezpiecznych tuneli. Razem te protokoły oferują wysoką jakość ochrony danych w komunikacji sieciowej, co czyni je idealnymi rozwiązaniami w połączeniach VPN i innych aplikacjach wymagających bezpiecznego tunelowania.

Tunelowanie VPN

Tunelowanie VPN (Virtual Private Network) to technika pozwalająca na tworzenie bezpiecznych połączeń między odległymi sieciami lub użytkownikami poprzez sieć publiczną, taką jak internet. Tunele VPN umożliwiają przesyłanie danych w sposób zaszyfrowany oraz zapewniają dodatkowe funkcje, takie jak ochrona prywatności i integralność transmisji.

Porównanie różnych tuneli sieciowych

Tunel	Typ ruchu	Opis	Zastosowanie
IPIP	IPv4 w IPv4	Kapsułkowanie pakietów IPv4 w innym pakiecie IPv4.	Łączenie sieci IPv4 przez internet.
6to4	IPv6 w IPv4	Automatyczny tunel do przesyłania IPv6 przez IPv4.	Migracja do IPv6, tunelowanie przez NAT.
IPIPv6	IPv4 w IPv6	Przesyłanie ruchu IPv4 w sieci IPv6.	Integracja IPv4 w infrastrukturze IPv6.
EoIPv6	Ethernet w IPv6	Tunelowanie ramek Ethernet przez IPv6.	Łączenie sieci warstwy 2 przez IPv6.
EoIP	Ethernet w IPv4	Tunelowanie ramek Ethernet przez IPv4.	Łączenie zdalnych sieci LAN.
GRE6	Różne protokoły w IPv6	Enkapsulacja różnych protokołów w IPv6.	Łączenie sieci w topologiach IPv6.
GRE	Różne protokoły w IPv4	Elastyczny tunel dla wielu typów ruchu.	Dynamiczne VPN, tunelowanie multicastu.
PPP	Różne (PPP)	Protokół punkt-punkt, używany do połączeń szeregowych.	Połączenia modemowe, dial-up.
PPTP	PPP w GRE	Stary, łatwy do skonfigurowania tunel VPN.	VPN w systemach Windows.
SSTP	PPP w HTTPS	VPN przez TLS, odporny na blokowanie.	Zastosowania w systemach Windows.
L2TP	Ethernet/PPP w UDP	Tunelowanie warstwy 2, często z IPSec.	VPN bez szyfrowania (z IPSec).
OVPN	IP w TLS	Szyfrowany tunel VPN w protokole TLS.	Bezpieczny VPN w wielu platformach.
PPPoE	PPP w Ethernet	PPP w sieciach Ethernet.	Dostęp do internetu u ISP.

Charakterystyka tunelowania VPN

VPN realizuje swoje funkcje poprzez enkapsulację pakietów w inny protokół tunelowy, co pozwala na ich przesyłanie przez sieć publiczną. Wyróżnia się następujące cechy tunelowania:

- **Poufność** – dane przesyłane przez tunel są szyfrowane, co zapobiega ich przechwyceniu.
- **Integralność** – zabezpieczenie przed modyfikacją przesyłanych pakietów.
- **Autentykacja** – uwierzytelnienie użytkowników i urządzeń, aby uniemożliwić dostęp nieuprawnionym podmiotom.

Poniżej przedstawiono najpopularniejsze technologie tunelowania VPN wraz z przykładową konfiguracją na urządzeniach MikroTik.

Tunele IP

Prosty tunel oparty na kapsułkowaniu pakietów IPv4 wewnątrz innych pakietów IPv4. Używany do łączenia sieci w różnych lokalizacjach.

Konfiguracja na MikroTik:

```
/interface ipip add name=ipip-tunnel remote-address=192.168.1.1
local-address=192.168.2.1
/ip address add address=10.10.10.1/30 interface=ipip-tunnel
```

6to4

Tunel przeznaczony do przesyłania IPv6 przez sieci IPv4. Jest to metoda automatyczna, stosowana głównie przy przejściu z IPv4 na IPv6.

Konfiguracja na MikroTik:

```
/interface 6to4 add local-address=192.168.1.1 name=6to4-tunnel
/ipv6 address add address=2002:c0a8:0101::1/64 interface=6to4-tunnel
```

IPIPv6

Tunel służący do enkapsulacji pakietów IPv4 w IPv6.

Konfiguracja na MikroTik:

```
/interface ipip6 add name=ipipv6-tunnel remote-address=2001:db8::2
local-address=2001:db8::1
/ip address add address=10.10.10.1/30 interface=ipipv6-tunnel
```

EoIPv6

Ethernet over IPv6 – pozwala na przesyłanie ramek Ethernet przez IPv6.

Konfiguracja na MikroTik:

```
/interface eoip add name=eoipv6-tunnel remote-address=2001:db8::2
local-address=2001:db8::1
```

EoIP

Ethernet over IP – tunel przenoszący ramki Ethernetowe poprzez IPv4.

Konfiguracja na MikroTik:

```
/interface eoip add remote-address=192.168.1.1 tunnel-id=0
/ip address add address=10.10.10.1/30 interface=eoip-tunnel
```

GRE6

Generic Routing Encapsulation dla IPv6, używane do tunelowania różnych protokołów przez IPv6.

Konfiguracja na MikroTik:

```
/interface gre6 add name=gre6-tunnel remote-address=2001:db8::2
local-address=2001:db8::1
/ip address add address=10.10.10.1/30 interface=gre6-tunnel
```

GRE

Podstawowy tunel GRE dla IPv4.

Konfiguracja na MikroTik:

```
/interface gre add name=gre-tunnel remote-address=192.168.1.1
local-address=192.168.2.1
/ip address add address=10.10.10.1/30 interface=gre-tunnel
```

PPP

Protokół Point-to-Point, używany głównie w połączeniach modemowych.

Konfiguracja na MikroTik:

```
/interface pppoe-client add interface=ether1 name=pppoe-out1
user=myuser password=mypass
```

PPTP

Point-to-Point Tunneling Protocol, klasyczny protokół VPN.

Konfiguracja na MikroTik:

```
/interface pptp-client add connect-to=192.168.1.1 user=myuser
password=mypass name=pptp-client
```

SSTP

Secure Socket Tunneling Protocol, stosowany w systemach Windows.

Konfiguracja na MikroTik:

```
/interface sstp-client add connect-to=192.168.1.1 user=myuser
password=mypass certificate=none
```

L2TP

Layer 2 Tunneling Protocol, często używany z IPSec.

Konfiguracja na MikroTik:

```
/interface l2tp-client add connect-to=192.168.1.1 user=myuser
password=mypass name=l2tp-client
```

OVPN

OpenVPN – popularny i bezpieczny protokół VPN.

Konfiguracja na MikroTik:

```
/interface ovpn-client add connect-to=192.168.1.1 user=myuser
password=mypass name=ovpn-client
```

PPPoE

Point-to-Point Protocol over Ethernet, używany w dostępie do internetu.

Konfiguracja na MikroTik:

```
/interface pppoe-client add interface=ether1 user=myuser
password=mypass name=pppoe-out1
```

Praktyczna implementacja VPN

W tej sekcji przedstawimy sposób konfiguracji różnych typów tuneli VPN na urządzeniach MikroTik, w tym OpenVPN (OVPN), PPTP oraz L2TP. Konfiguracja obejmuje zarówno ustawienia serwera, jak i klienta, umożliwiając bezpieczną komunikację między zdalnymi lokalizacjami i użytkownikami.

Konfiguracja OpenVPN (OVPN)

OpenVPN jest jednym z najbezpieczniejszych protokołów VPN, który używa protokołu SSL/TLS do szyfrowania i autentykacji.

Konfiguracja serwera OpenVPN na MikroTik

Aby skonfigurować serwer OpenVPN na MikroTik, wykonaj następujące kroki:

- Utwórz certyfikat serwera i klienta. Certyfikaty są wymagane do ustanowienia bezpiecznego połączenia.
- Włącz usługę OpenVPN i skonfiguruj odpowiednie parametry.

Przykład konfiguracji:

```
    /certificate add name=server-cert common-name=server
    /certificate sign server-cert
    /interface ovpn-server server set enabled=yes certificate=server-cert
    /ppp profile add name=ovpn-profile local-address=10.10.10.1
    remote-address=10.10.10.2
    /ppp secret add name=ovpn-user password=ovpn-password profile=ovpn-
profile
```

Konfiguracja klienta OpenVPN na MikroTik

Na urządzeniu klienta wprowadź następujące polecenia:

```
    /interface ovpn-client add connect-to=server_ip user=ovpn-user
    password=ovpn-password certificate=client-cert
```

Warto dodać, że na urządzeniu klienckim również musisz zaimportować odpowiedni certyfikat.

Konfiguracja PPTP

PPTP (Point-to-Point Tunneling Protocol) to starszy protokół VPN, który jest prosty do skonfigurowania, ale mniej bezpieczny niż inne opcje.

Konfiguracja serwera PPTP na MikroTik

Aby skonfigurować serwer PPTP:

- Włącz usługę PPTP i skonfiguruj profil PPP.
- Skonfiguruj użytkowników VPN.

Przykład konfiguracji serwera PPTP:

```
    /interface pptp-server server set enabled=yes
    /ppp profile add name=pptp-profile local-address=192.168.1.1
    remote-address=192.168.1.2
    /ppp secret add name=pptp-user password=pptp-password
    profile=pptp-profile
```

Konfiguracja klienta PPTP na MikroTik

Na urządzeniu klienckim wprowadź następujące polecenia:

```
    /interface pptp-client add connect-to=server_ip user=pptp-user
    password=pptp-password
```

Konfiguracja L2TP

L2TP (Layer 2 Tunneling Protocol) to protokół VPN często używany z IPSec dla zwiększenia bezpieczeństwa. L2TP zapewnia lepsze zabezpieczenia niż PPTP, a także oferuje lepszą wydajność.

Konfiguracja serwera L2TP na MikroTik

Aby skonfigurować serwer L2TP:

- Włącz usługę L2TP i skonfiguruj odpowiednie profile.
- Skonfiguruj IPSec dla dodatkowego bezpieczeństwa.

Przykład konfiguracji serwera L2TP:

```
    /interface l2tp-server server set enabled=yes ipsec-secret=my-secret
    use-ipsec=yes
    /ppp profile add name=l2tp-profile local-address=192.168.10.1
    remote-address=192.168.10.2
    /ppp secret add name=l2tp-user password=l2tp-password profile=l2tp-
profile
    /ip ipsec peer add address=server_ip secret=my-secret
```

Konfiguracja klienta L2TP na MikroTik

Na urządzeniu klienckim wprowadź następujące polecenia:

```
    /interface l2tp-client add connect-to=server_ip user=l2tp-user
    password=l2tp-password use-ipsec=yes ipsec-secret=my-secret
```

Podsumowanie

Wszystkie trzy protokoły VPN (OpenVPN, PPTP, L2TP) mają swoje zalety i wady:

- **OpenVPN** – Najbezpieczniejszy, ale wymaga konfiguracji certyfikatów.
- **PPTP** – Łatwy w konfiguracji, ale mniej bezpieczny.
- **L2TP** – Bardziej bezpieczny niż PPTP, zwłaszcza gdy jest używany z IPSec.

Każdy z tych protokołów jest odpowiedni do różnych zastosowań. OpenVPN jest zalecany do środowisk, gdzie bezpieczeństwo jest kluczowe, natomiast PPTP może być stosowane tam, gdzie łatwość konfiguracji ma większe znaczenie. L2TP z IPSec to dobry kompromis pomiędzy bezpieczeństwem a wydajnością.

Przydatne polecenia do diagnostyki sieciowej w systemach operacyjnych Linux i Windows

Diagnostyka sieciowa jest kluczowym elementem w utrzymaniu i rozwiązywaniu problemów z połączeniami sieciowymi. W systemach operacyjnych Linux i Windows istnieje wiele narzędzi i poleceń, które pozwalają na monitorowanie stanu sieci, testowanie połączeń oraz rozwiązywanie problemów związanych z konfiguracją i łącznością.

Diagnostyka w systemie Linux

W systemie Linux dostępnych jest wiele narzędzi diagnostycznych, które pomagają w analizie ruchu sieciowego, sprawdzaniu konfiguracji i diagnozowaniu problemów z połączeniami sieciowymi.

ping

Polecenie `ping` służy do sprawdzania, czy dany host w sieci jest osiągalny. Wysyła pakiety ICMP Echo Request do zdalnego hosta, a ten odpowiada pakietami Echo Reply.

```
ping <adres IP lub nazwa hosta>
```

Przykład:

```
ping 8.8.8.8
```

Wyjaśnienie: To polecenie sprawdza, czy serwer Google DNS (8.8.8.8) jest osiągalny.

ifconfig

Polecenie `ifconfig` pozwala na wyświetlenie informacji o interfejsach sieciowych w systemie, w tym adresach IP, maskach podsieci, statystykach ruchu i błędach.

```
ifconfig
```

Przykład:

```
ifconfig eth0
```

Wyjaśnienie: Wyświetla szczegółowe informacje o interfejsie `eth0`.

traceroute

Polecenie `traceroute` służy do śledzenia trasy, jaką pakiety pokonują od lokalnego hosta do docelowego adresu w sieci. Pokazuje kolejne przeskoki (hopy) i czas odpowiedzi z każdego z nich.

```
traceroute <adres IP lub nazwa hosta>
```

Przykład:

```
traceroute www.google.com
```

Wyjaśnienie: Pokazuje trasę, jaką pakiety pokonują do serwera Google.

netstat

Polecenie `netstat` umożliwia wyświetlanie informacji o połączeniach sieciowych, tabelach routingu, statystykach interfejsów oraz innych aspektach konfiguracji sieci.

```
netstat -tuln
```

Wyjaśnienie: Wyświetla listę wszystkich nasłuchujących portów (`-tuln`) w systemie.

nslookup

Polecenie `nslookup` pozwala na wykonywanie zapytań DNS (Domain Name System), aby uzyskać informacje o domenach i ich przypisanych adresach IP.

```
nslookup <domena>
```

Przykład:

```
nslookup www.google.com
```

Wyjaśnienie: Wyświetla adres IP przypisany do domeny `www.google.com`.

Diagnostyka w systemie Windows

W systemie Windows dostępnych jest również wiele narzędzi diagnostycznych, które pozwalają na monitorowanie połączeń sieciowych, sprawdzanie konfiguracji i rozwiązywanie problemów z dostępem do sieci.

ping

Podobnie jak w systemie Linux, polecenie `ping` w Windows służy do sprawdzania dostępności hostów w sieci.

```
ping <adres IP lub nazwa hosta>
```

Przykład:

```
ping 8.8.8.8
```

Wyjaśnienie: Sprawdza dostępność serwera Google DNS.

ipconfig

Polecenie `ipconfig` w Windows służy do wyświetlania konfiguracji interfejsów sieciowych, takich jak adresy IP, maski podsieci, brama domyślna, serwery DNS.

```
ipconfig
```

Przykład:

```
ipconfig /all
```

Wyjaśnienie: Wyświetla pełne informacje o konfiguracji sieciowej wszystkich interfejsów w systemie.

tracert

Polecenie `tracert` jest odpowiednikiem `traceroute` w systemie Windows. Pokazuje trasę pakietów od źródła do celu, z uwzględnieniem każdego przeskoku.

```
tracert <adres IP lub nazwa hosta>
```

Przykład:

```
tracert www.google.com
```

Wyjaśnienie: Pokazuje trasę do serwera Google.

netstat

Podobnie jak w systemie Linux, polecenie `netstat` w Windows służy do wyświetlania połączeń sieciowych oraz informacji o portach nasłuchujących w systemie.

```
netstat -ano
```

Wyjaśnienie: Wyświetla wszystkie połączenia i porty nasłuchujące (`-ano`) oraz identyfikatory procesów.

nslookup

Polecenie `nslookup` w systemie Windows działa identycznie jak w Linuxie i pozwala na zapytania DNS w celu uzyskania informacji o adresach IP związanych z nazwami domen.

```
nslookup <domena>
```

Przykład:

```
nslookup www.google.com
```

Wyjaśnienie: Wykonuje zapytanie DNS dla domeny `www.google.com`.

Podsumowanie

Diagnostyka sieciowa jest kluczowym elementem w utrzymaniu i rozwiązywaniu problemów związanych z połączeniami sieciowymi. W systemach operacyjnych Linux i Windows dostępne są liczne narzędzia, które pozwalają na szybkie wykrywanie i diagnozowanie problemów. Narzędzia takie jak `ping`, `traceroute`, `netstat`, `nslookup`, czy `ipconfig` stanowią podstawowe wyposażenie administratorów sieciowych, umożliwiając monitorowanie, analizowanie oraz rozwiązywanie problemów z połączeniami sieciowymi.

Sieci Światłowodowe

Sieci światłowodowe stanowią podstawę nowoczesnych systemów komunikacyjnych, oferując szybki i niezawodny transfer danych na dużą odległość. W tej sekcji omówimy konstrukcję światłowodu, zasady jego działania, rodzaje oraz zastosowanie w sieciach komputerowych i dostępowych.

Konstrukcja światłowodu

Światłowód jest przewodem optycznym, który służy do przesyłania sygnałów świetlnych. Składa się z dwóch podstawowych elementów: rdzenia i płaszcza.

Współczynnik załamania

Współczynnik załamania materiału w rdzeniu i płaszczu światłowodu jest kluczowy dla efektywnego przesyłania sygnału świetlnego. Rdzeń ma wyższy współczynnik załamania niż płaszcz, co pozwala na propagację światła przez odbicia wewnętrzne.

Zasada odbicia

Światłowód działa na zasadzie całkowitego wewnętrznego odbicia. Światło, które wpada pod odpowiednim kątem na granicę pomiędzy rdzeniem a płaszczem, jest całkowicie odbite z powrotem do rdzenia, dzięki czemu sygnał jest kierowany przez światłowód.

Rodzaje światłowodów

Światłowody dzielą się na kilka typów w zależności od konstrukcji rdzenia i płaszcza oraz sposobu rozchodzenia się światła w rdzeniu.

- **Światłowody jednordzeniowe (SM - Single Mode)**: Światło przemieszcza się tylko po jednej ścieżce w rdzeniu, co pozwala na przesyłanie sygnałów na bardzo dużą odległość bez utraty jakości. Współczynnik załamania w rdzeniu i płaszczu różni się znacznie.
- **Światłowody wielomodowe (MM - Multi Mode)**: Światło przemieszcza się wieloma ścieżkami w rdzeniu, co umożliwia krótsze odległości przesyłania. Współczynnik załamania w rdzeniu i płaszczu jest mniej różny niż w przypadku SM.

Gradientowe światłowody

Światłowody gradientowe to rodzaj światłowodów, w których współczynnik załamania zmienia się stopniowo wzdłuż rdzenia. Można je podzielić na:

- **Światłowody gradientowe stopniowe**: Współczynnik załamania zmienia się w sposób skokowy w różnych warstwach rdzenia.
- **Światłowody gradientowe ciągłe**: Współczynnik załamania zmienia się w sposób ciągły wzdłuż całego rdzenia.

Zasada działania światłowodu

Podstawową zasadą działania światłowodu jest **całkowite wewnętrzne odbicie** (TIR - Total Internal Reflection). Sygnał świetlny, który wpada na granicę rdzenia i płaszcza pod odpowiednim kątem, jest całkowicie odbity z powrotem do rdzenia, co pozwala na jego przesyłanie przez długie odległości.

Stożek akceptacji

Każdy światłowód ma określony kąt akceptacji, który definiuje, w jakim zakresie kątów może wchodzić światło, aby zostało ono skutecznie załadowane do rdzenia. Kąt ten tworzy tzw. stożek akceptacji, który jest zależny od współczynnika załamania rdzenia i płaszcza.

Rodzaje światłowodów

Single Mode (SM)

Światłowody jednordzeniowe są zaprojektowane tak, aby światło przechodziło tylko jednym modzie, czyli jednym promieniu wzdłuż rdzenia. Stosowane są w sieciach o dużej przepustowości, gdzie potrzebna jest transmisja na duże odległości (do kilkuset kilometrów).

Multi Mode (MM)

Światłowody wielomodowe pozwalają na przesyłanie wielu modów światła w różnych kierunkach w rdzeniu. Stosowane są w sieciach lokalnych (LAN) i na krótszych odległościach.

Modulacje oraz okna transmisyjne

Modulacje

W światłowodach stosuje się różne rodzaje modulacji, w tym:

- **Modulacja amplitudy (AM)**: Zmienia amplitudę sygnału świetlnego.
- **Modulacja częstotliwości (FM)**: Zmienia częstotliwość sygnału świetlnego.
- **Modulacja fazy (PM)**: Zmienia fazę sygnału świetlnego.

Okna transmisyjne

Światłowody wykorzystują różne długości fal światła, z których każda jest optymalna do transmisji w określonych oknach transmisyjnych. Najczęściej stosowane to:

- **Okno 850 nm (krótkozasięgowe)**: Używane w światłowodach MM.
- **Okno 1310 nm (średni zasięg)**: Używane w światłowodach SM.
- **Okno 1550 nm (długozasięgowe)**: Używane w długozasięgowych transmisjach światłowodowych.

Zastosowanie w sieciach komputerowych

W sieciach komputerowych światłowody stosowane są do łączenia urządzeń o dużej przepustowości, takich jak przełączniki, routery czy serwery. Przykładem są moduły SFP (Small Form-factor Pluggable), SFP+ (szybsza wersja SFP) oraz QSFP+ (Quad Small Form-factor Pluggable), które pozwalają na szybkie połączenia optyczne w sieciach 10G, 40G i 100G.

- **SFP**: Moduł optyczny, który jest wykorzystywany w transceiverach do przesyłania danych w sieciach 1G, 10G.
- **SFP+**: Szybsza wersja SFP, która obsługuje przepustowość do 10 Gbps.
- **QSFP+**: Moduł wykorzystywany w połączeniach o wysokiej przepustowości, takich jak 40G i 100G.

Zastosowanie w sieciach dostępowych

Światłowody są również wykorzystywane w sieciach dostępowych, takich jak FTTH (Fiber To The Home) oraz GPON (Gigabit Passive Optical Network), które pozwalają na dostarczanie usług szerokopasmowego internetu do użytkowników końcowych.

- **FTTH (Fiber To The Home)**: Światłowód prowadzi bezpośrednio do domu użytkownika, zapewniając bardzo wysoką prędkość internetu.
- **GPON (Gigabit Passive Optical Network)**: Technologia optyczna, która umożliwia przesyłanie danych do wielu użytkowników za pomocą jednej linii światłowodowej.

Bilans łącza światłowodowego

Bilans łącza światłowodowego to analiza strat sygnału w czasie transmisji przez światłowód. Straty te wynikają z różnych czynników, takich jak:

- Straty tłumienia (absorpcyjne, rozpraszające).
- Straty spowodowane załamaniem i odbiciem.
- Straty spowodowane uszkodzeniem kabla lub jego nieprawidłowym ułożeniem.

Aby zapewnić odpowiednią jakość transmisji, bilans łącza musi być kontrolowany i dostosowywany w zależności od długości kabla, rodzaju światłowodu i zastosowanej modulacji.

Przykładowa konfiguracja wkładki SFP na switchu Cisco

Wkładki SFP (Small Form-factor Pluggable) są wykorzystywane w switchach Cisco do zapewnienia połączeń optycznych o dużej przepustowości. Moduły SFP są wymienne i umożliwiają dostosowanie portów switcha do różnych rodzajów medium transmisyjnego, takich jak światłowód lub miedź. Konfiguracja wkładki SFP w switchu Cisco jest stosunkowo prosta, jednak wymaga pewnych kroków i uwagi, aby zapewnić prawidłowe działanie.

Instalacja wkładki SFP

1. **Zainstalowanie wkładki SFP**: Aby zainstalować wkładkę SFP, należy umieścić ją w odpowiednim porcie na switchu. Wkładka powinna być odpowiednia do typu kabla, z którego będzie korzystać (np. światłowód jednomodowy (SM) lub wielomodowy (MM)).

```
Switch(config)# interface gigabitEthernet 1/0/1
Switch(config-if)# no shutdown
```

Polecenie `no shutdown` włącza port, na którym zainstalowana jest wkładka SFP.

2. **Podłączenie kabla**: Po zainstalowaniu wkładki SFP, kabel światłowodowy należy podłączyć do wkładki, upewniając się, że złącze jest odpowiednio zabezpieczone.

Dane dostępne z wkładki SFP

Po zainstalowaniu wkładki SFP w switchu Cisco, istnieje możliwość monitorowania i pobierania różnych danych dotyczących wkładki oraz jej stanu. Można uzyskać takie informacje jak:

- **Typ wkładki**: Określenie, czy wkładka jest typu SFP, SFP+ lub QSFP+.
- **Numer seryjny**: Numer seryjny wkładki, który pozwala na identyfikację modułu.
- **Stan optyczny**: Informacje o mocy optycznej nadawanej i odbieranej przez wkładkę, co jest kluczowe dla oceny jakości połączenia.
- **Temperatura wkładki**: Temperatura pracy wkładki, co może pomóc w identyfikacji problemów związanych z przegrzewaniem.
- **Napięcie wkładki**: Informacje o napięciu zasilania wkładki, które pozwalają na diagnozowanie ewentualnych problemów z zasilaniem.

Aby sprawdzić te dane, można użyć polecenia:

```
    Switch# show interface transceiver
```

To polecenie wyświetli szczegóły dotyczące wkładki SFP, takie jak numer seryjny, typ, moc optyczną i inne parametry. Przykładowy wynik może wyglądać następująco:

```
    Switch# show interface transceiver
    Port  Transceiver Type       State Temperature Voltage  Rx Power  Tx
Power
    Gi1/0/1  SFP+     10GBase-SR OK    34°C        3.3V     -2.3dBm
-1.2dBm
```

Problemy, które mogą wystąpić przy konfiguracji

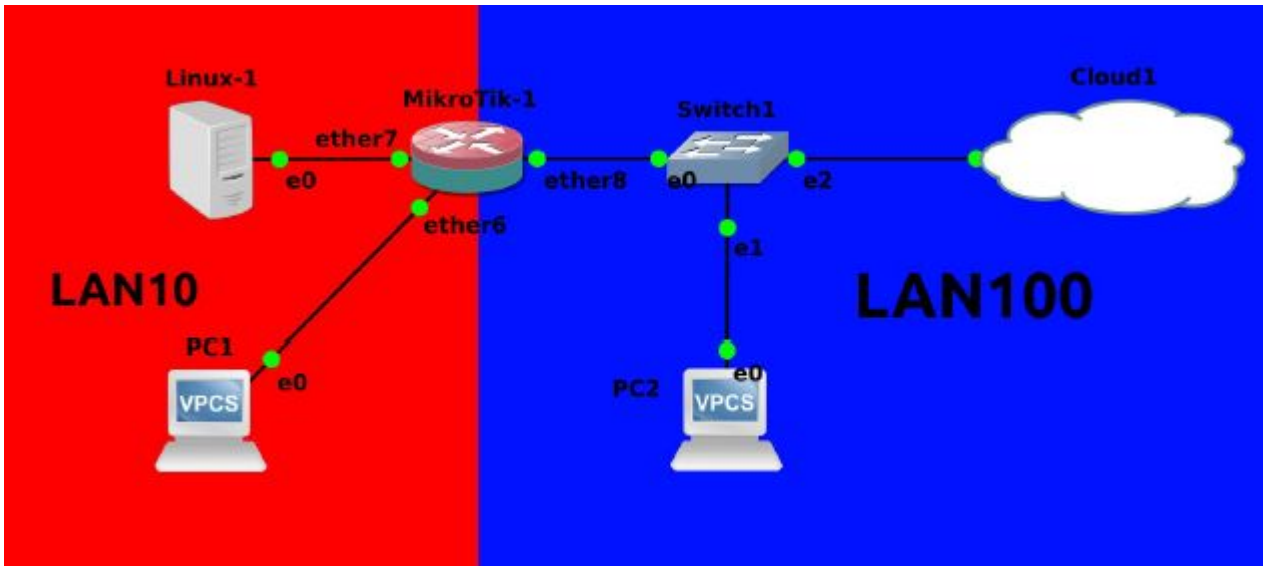
Mimo że instalacja wkładki SFP jest stosunkowo prosta, mogą wystąpić pewne problemy, które mogą utrudnić prawidłową konfigurację i działanie połączeń:

- **Niekompatybilność wkładki**: Nie wszystkie wkładki SFP są kompatybilne ze wszystkimi switchami Cisco. Należy upewnić się, że wkładka jest zatwierdzona przez Cisco i zgodna z modelem switcha.
- **Problemy z zasilaniem**: Niskie napięcie zasilania lub problemy z zasilaczem mogą powodować niestabilne działanie wkładki, co może prowadzić do problemów z połączeniem.
- **Błędy związane z światłowodem**: Niewłaściwe podłączenie kabla światłowodowego, jego uszkodzenie lub użycie niewłaściwego typu kabla (np. SM zamiast MM) może prowadzić do braku połączenia lub niskiej jakości transmisji.
- **Słaba moc optyczna**: Niska moc optyczna nadawana przez wkładkę może wskazywać na problem z optyką lub zbyt długą odległość między urządzeniami. Można to zweryfikować za pomocą polecenia `show interface transceiver`.
- **Problemy z temperaturowym limitem**: Wysoka temperatura pracy wkładki może prowadzić do jej uszkodzenia lub niestabilności. Należy monitorować temperaturę, zwłaszcza w środowiskach o wysokiej temperaturze.
- **Brak aktywacji portu**: Wkładka SFP nie będzie działać, jeśli port na switchu jest wyłączony. Należy upewnić się, że port jest aktywowany poprzez polecenie `no shutdown`.

W przypadku wystąpienia problemów, zaleca się sprawdzenie statusu wkładki za pomocą polecenia `show interface transceiver`, a także konsultację z dokumentacją Cisco w celu zapewnienia kompatybilności oraz prawidłowej konfiguracji.

Projekt: Prosta sieć dostępowa dla użytkowników domowych z przekierowaniem portów do lokalnego serwera z wykorzystaniem routerów MikroTik

Definicja projektu



Schemat środowiska wirtualnego

Z schematu powyżej możemy wywnioskować że komputer PC1 oraz serwer Linux-1 są po stronie lokalnej naszego routera. Natomiast komputer PC2 jest po stronie WAN razem z chmurą symbolizującą dostęp do internetu Cloud1. Komputer PC2 będzie wykorzystany do symulowania klienta który chce się podłączyć zdalnie do naszego serwera przez DNAT.
Parametry symulacji:

- LAN100 to sieć o adresacji `10.42.0.0./24` (udaje ona sieć WAN)
- Linux-1 ma usługe HTTP oraz DNS
- LAN10 to sieć o adresacji `192.168.1.0/24` (udaje ona sieć LAN)
- Port Ether8 na MikroTiku jest podłączony do sieci WAN
- Port Ether6 jest podłączony do komputera klienckiego PC1
- Port Ether7 jest podłączony do serwera Linux-1
- Switch1 oraz PC2 nie zostały wykorzystane w tej symulacji

Definicja Wymagań

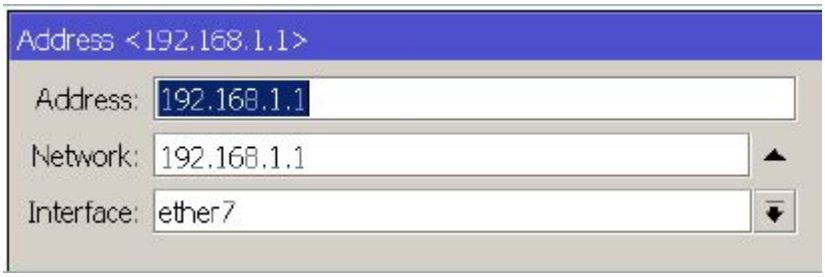
Funkcje routera:

- Mikrotik1 musi mieć skonfigurowane funkcję SNAT, DHCP client na porcie 8
- Mikrotik1 musi mieć skonfigurowane funkcję bridge, DHCP server na portach 6 i 7

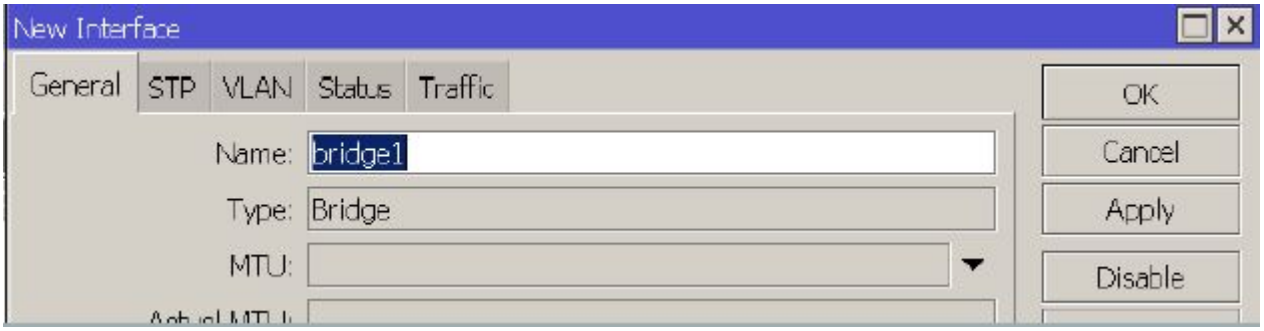
Testy:

- PC1 → ping `10.42.0.1` (ping bramy sieci LAN100)
- Linux-1 → HTTP `10.42.0.252` (wejście na serwer WWW w sieci LAN100)
- Linux-1 → DNS local.server.com (sprawdzenie funkcjonowania DNS)

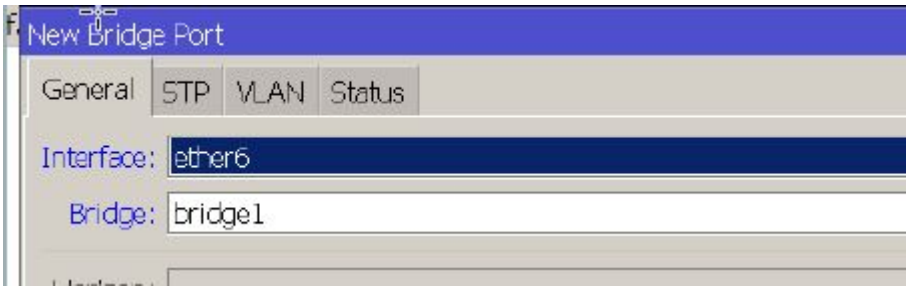
Implementacja



Konfiguracja adresu na interfejsie ether7

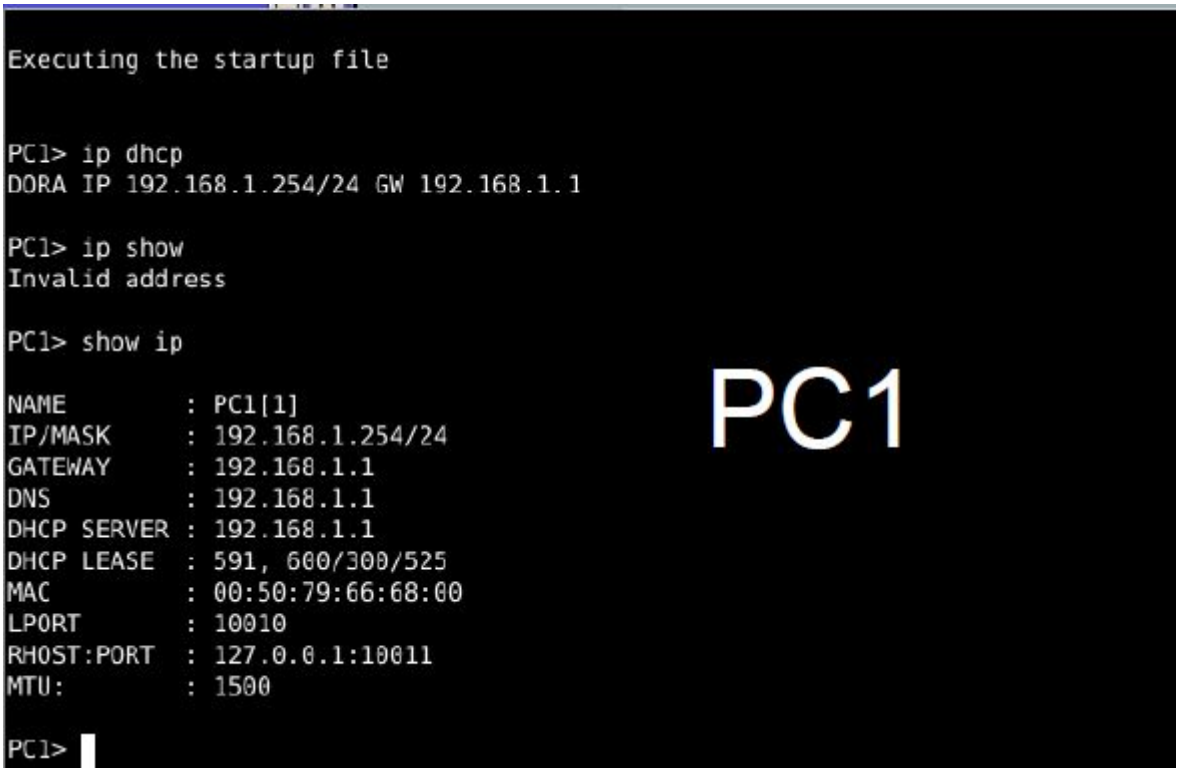


Dodanie bridge'a

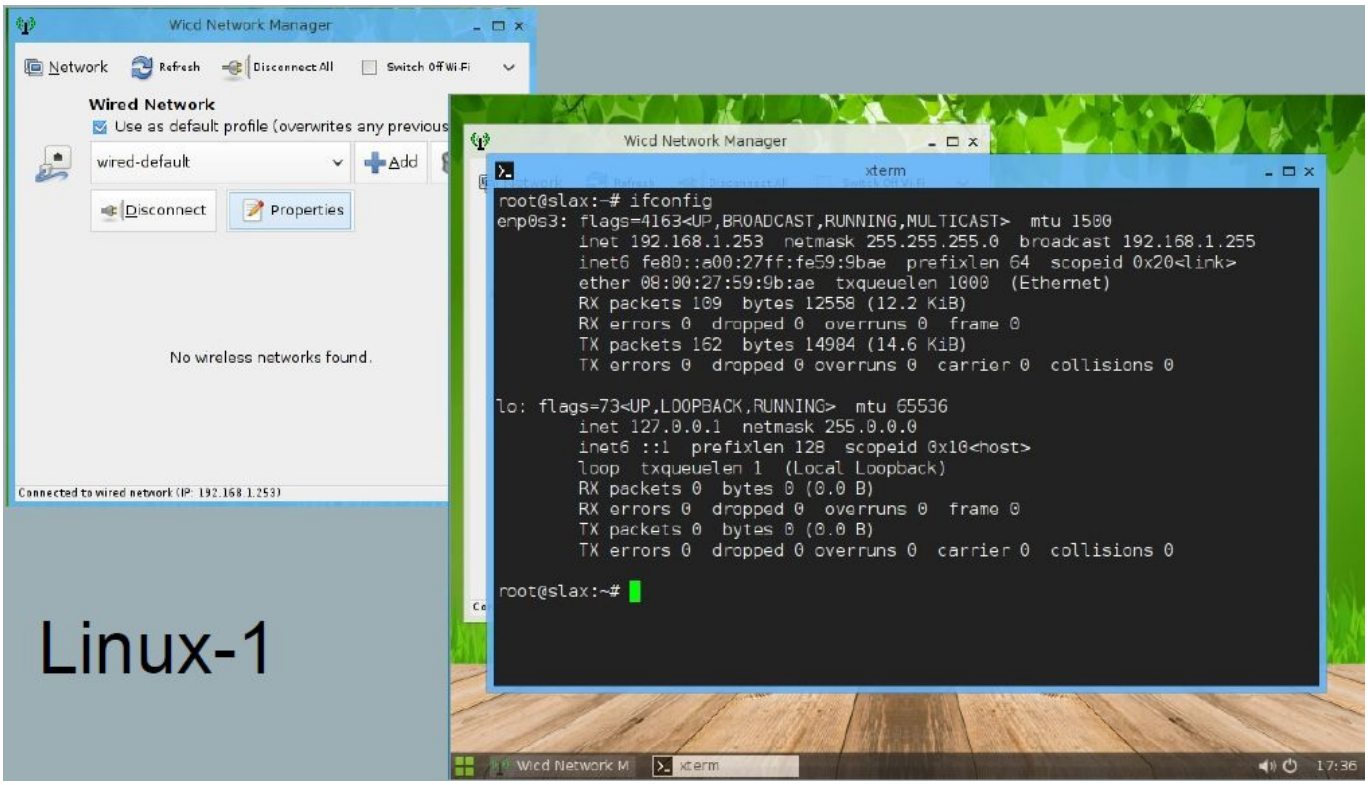


Dodanie portu do bridge'a

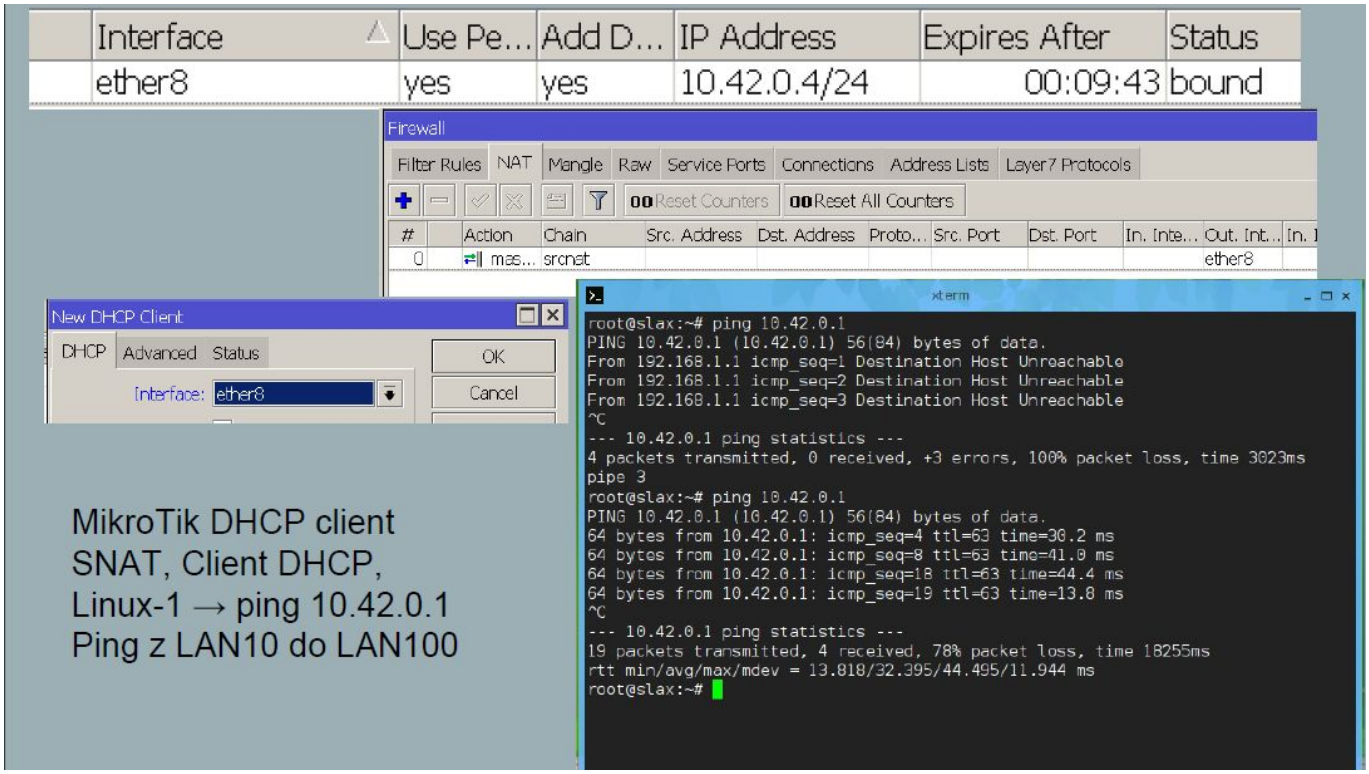
Konfiguracja DHCP server za pomocą kreatora



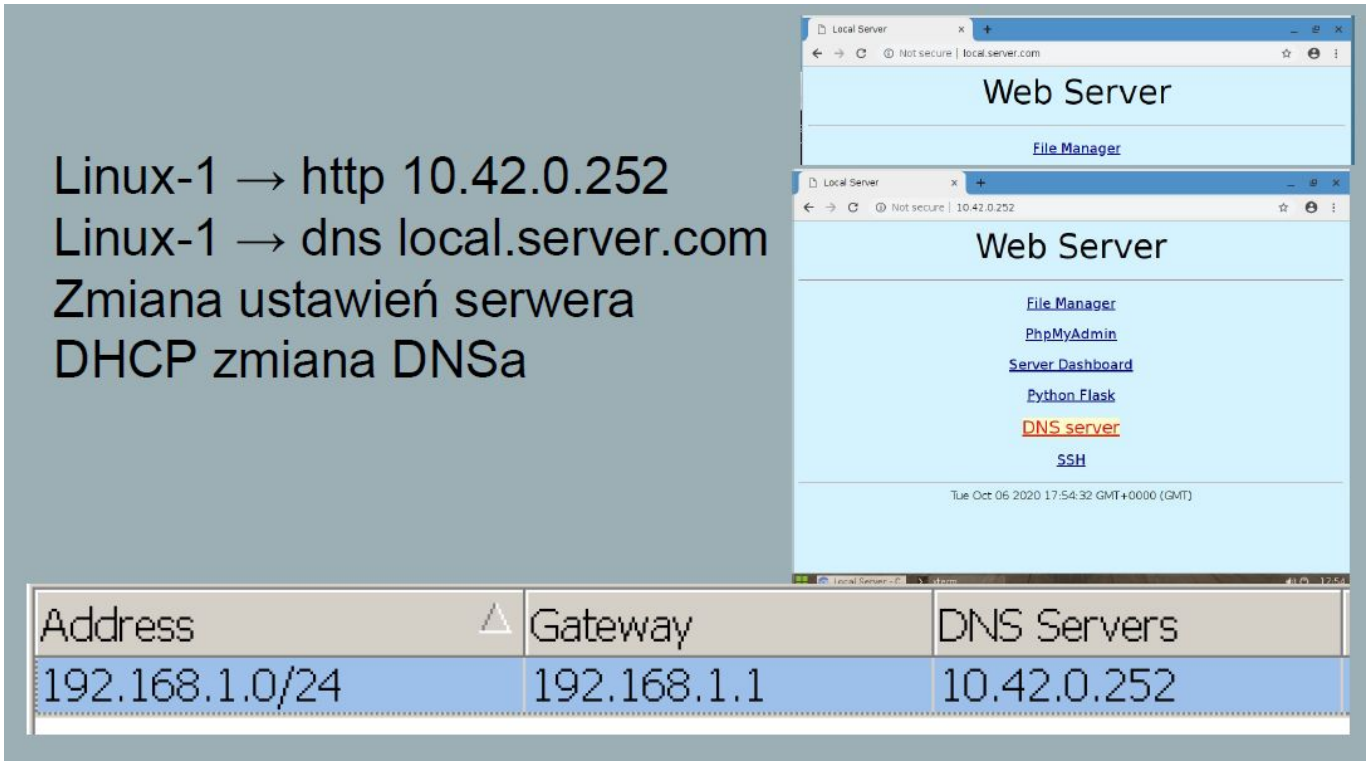
Sprawdzenie działania serwera DHCP



Sprawdzenie adresu IP serwera Linux-1



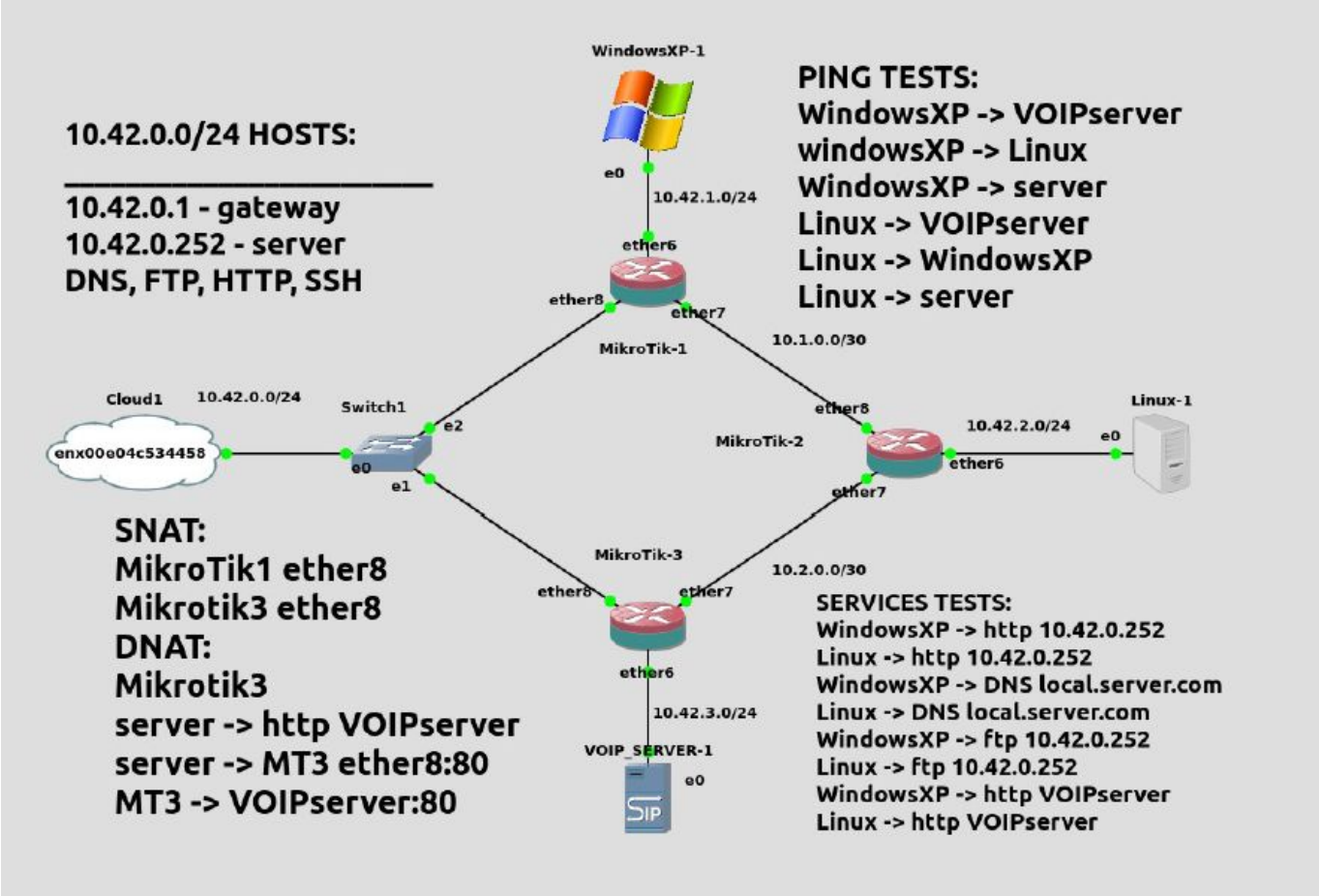
Sprawdzenie przydzielenia adresu WAN na mikrotiku, konfiguracja SNAT oraz sprawdzenie komunikacji na Linux-1



Sprawdzenie komunikacji WWW z serwerem zewnętrznym oraz sprawdzenie funkcjonalności DNS

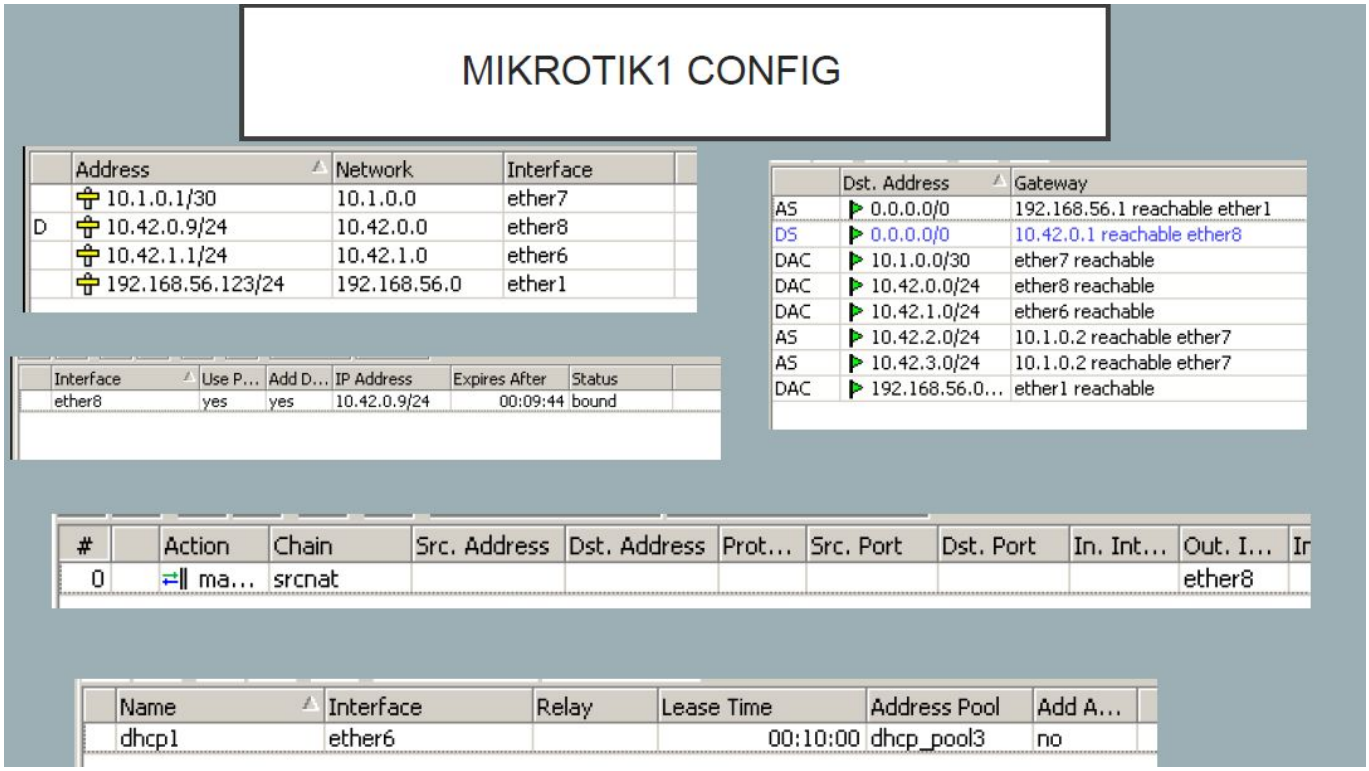
Projekt: Sieci WAN dla firmy z trzema

lokalizacjami gdzie dwie maja dostęp do internetu a trzecia jest podłączona poprzez łącze dedykowane

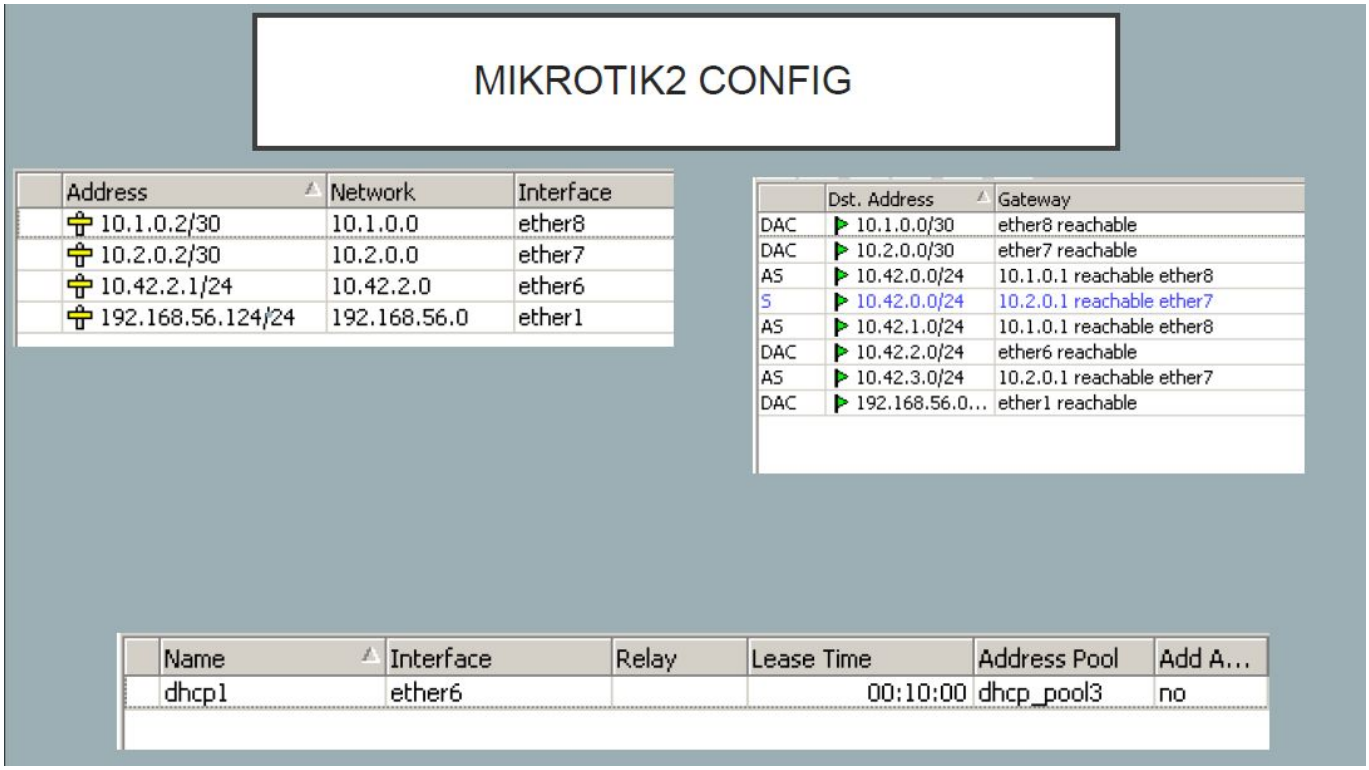


Schemat środowiska wirtualnego oraz wymagań projektu

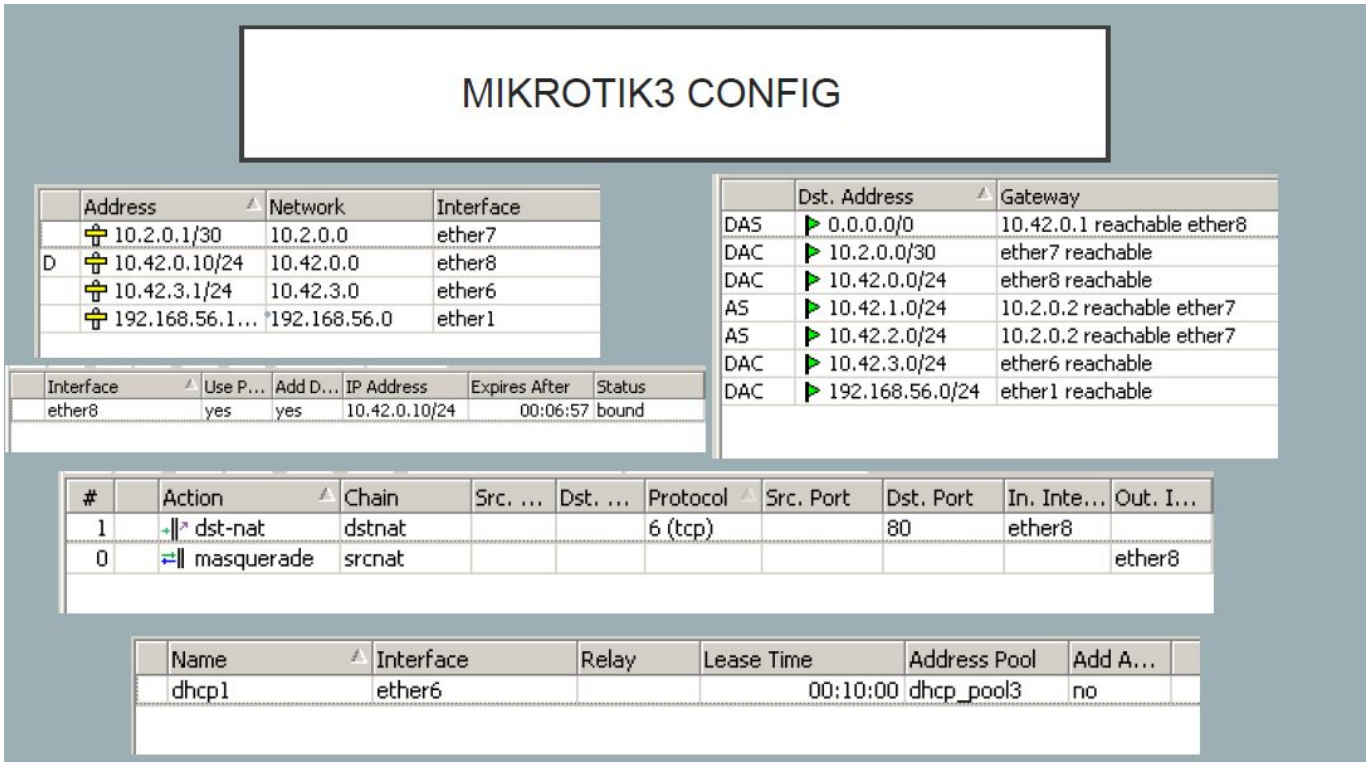
Sieć w tym przypadku składa się z trzech routerów z czego tylko dwa mają dostęp do sieci 10.42.0.0/24 która udaje WAN. Musimy skonfigurować tak sieć aby wszyscy z wszystkimi się widzieli pomiędzy trzema lokalizacjami oraz żeby mogli wszyscy wychodzić do internetu.



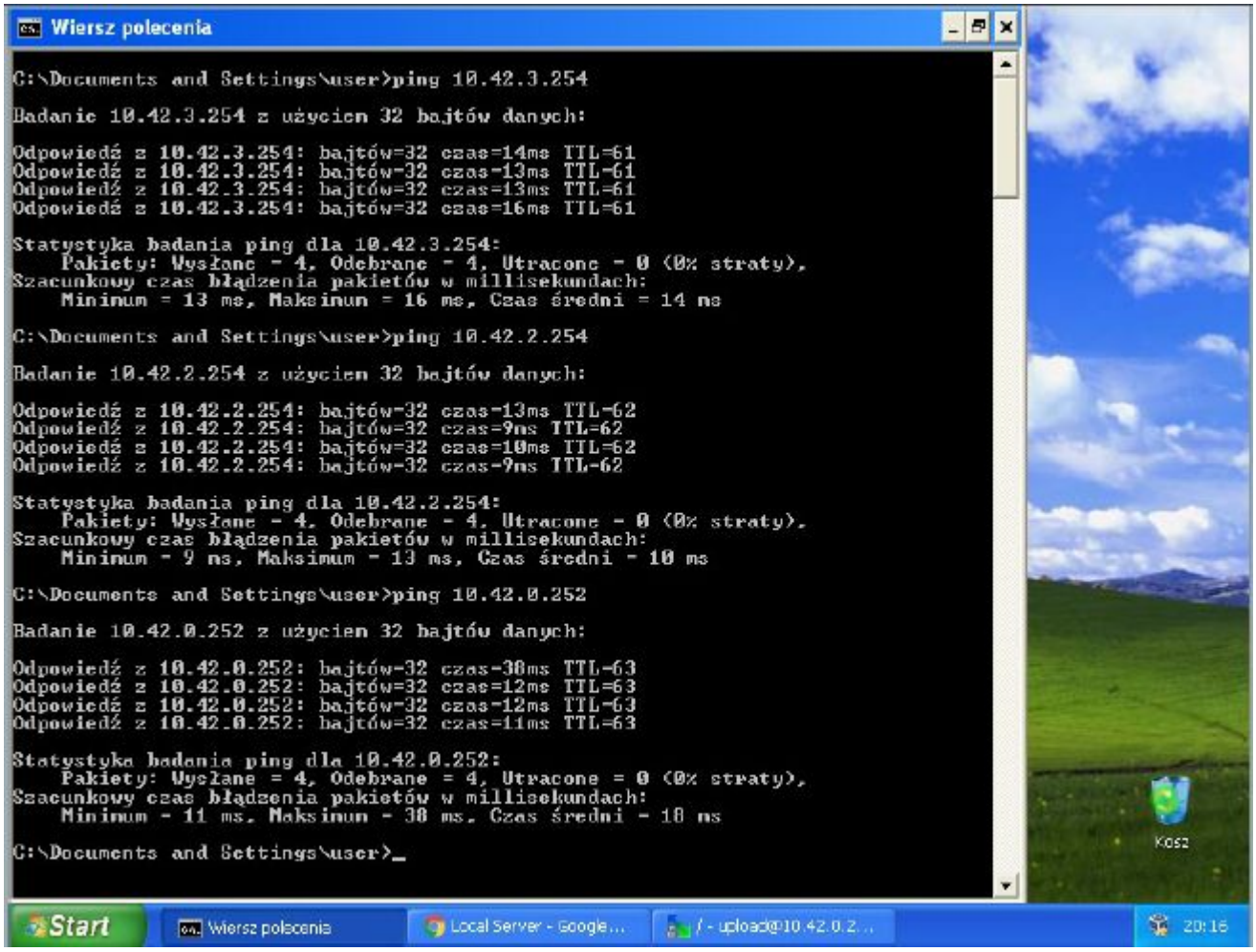
Konfiguracja Router 1



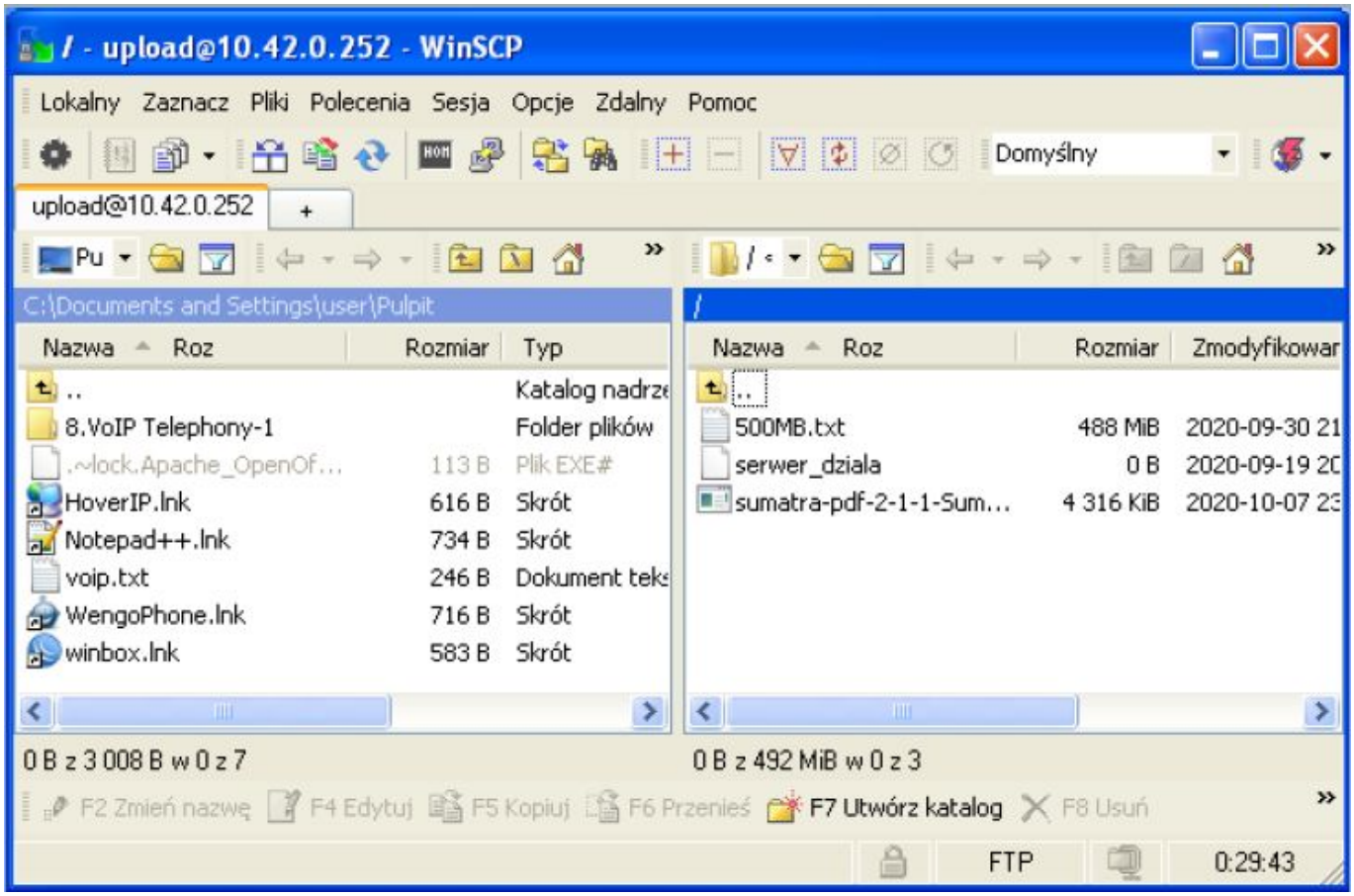
Konfiguracja Router 2



Konfiguracja Router 3



Testy ping na komputerze z windows



Testy FTP na komputerze z windows

Testy WWW i DNS na komputerze z windows

```
root@slax:~# ping 10.42.0.252
PING 10.42.0.252 (10.42.0.252) 56(84) bytes of data.
64 bytes from 10.42.0.252: icmp_seq=1 ttl=62 time=22.9 ms
64 bytes from 10.42.0.252: icmp_seq=2 ttl=62 time=13.5 ms
64 bytes from 10.42.0.252: icmp_seq=3 ttl=62 time=13.0 ms
^C
--- 10.42.0.252 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 13.024/16.507/22.971/4.575 ms
root@slax:~# ping 10.42.1.254
PING 10.42.1.254 (10.42.1.254) 56(84) bytes of data.
64 bytes from 10.42.1.254: icmp_seq=1 ttl=126 time=6.22 ms
64 bytes from 10.42.1.254: icmp_seq=2 ttl=126 time=9.76 ms
64 bytes from 10.42.1.254: icmp_seq=3 ttl=126 time=8.07 ms
^C
--- 10.42.1.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 6.222/8.019/9.765/1.446 ms
root@slax:~# ping 10.42.3.254
PING 10.42.3.254 (10.42.3.254) 56(84) bytes of data.
64 bytes from 10.42.3.254: icmp_seq=1 ttl=62 time=6.11 ms
64 bytes from 10.42.3.254: icmp_seq=2 ttl=62 time=6.18 ms
64 bytes from 10.42.3.254: icmp_seq=3 ttl=62 time=11.1 ms
^C
--- 10.42.3.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2009ms
rtt min/avg/max/mdev = 6.116/7.812/11.131/2.347 ms
root@slax:~#
```

Testy ping na komputerze z linux

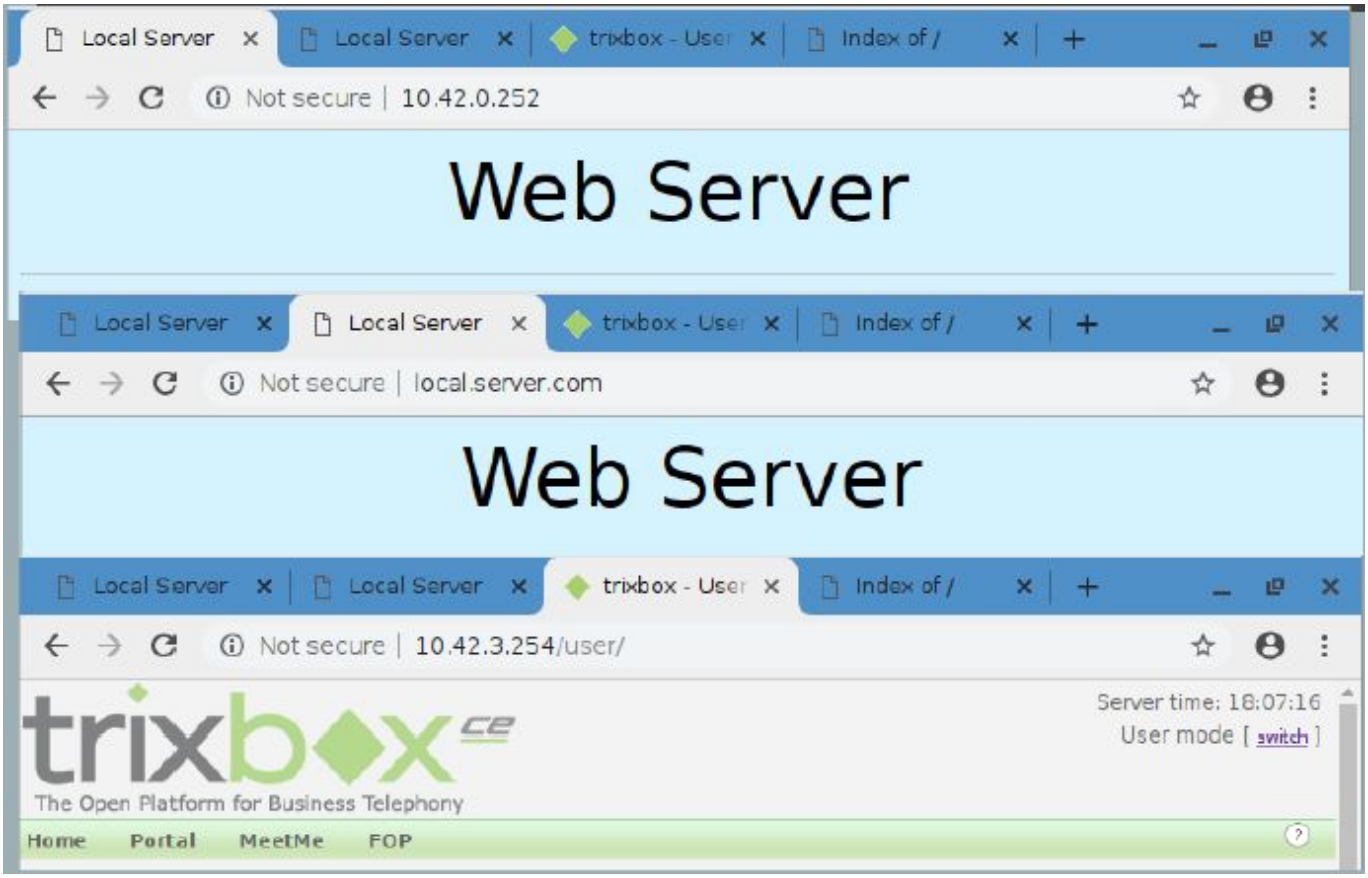
Index of / | Local Server | trixbox - User | Index of /

Not secure | ftp://10.42.0.252

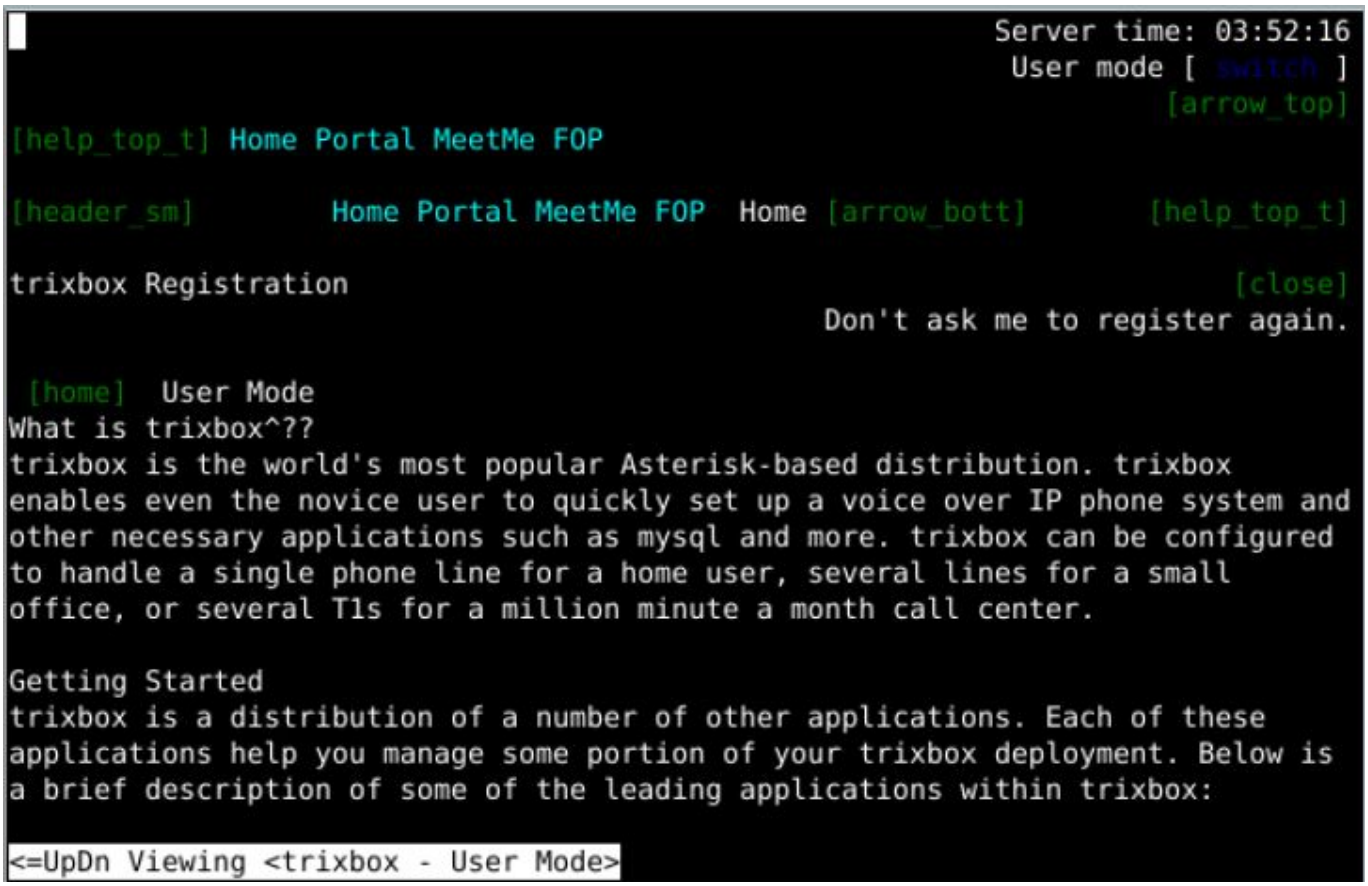
Index of /

Name	Size	Date Modified
500MB.txt	488 MB	9/30/20, 7:38:00 PM
serwer_dziala	0 B	9/19/20, 6:25:00 PM
sumatra-pdf-2-1-1-SumatraPDF-2.1.1-install.exe	4.2 MB	10/7/20, 9:24:00 PM

Testy FTP na komputerze z linux



Testy WWW i DNS na komputerze z linux



Testy WWW na serwerze VOIP (w trybie tekstowym)

Niszowe protokoły wykorzystywane w sieciach IP

W sieciach opartych na protokole IP dominują standardowe protokoły, takie jak TCP, UDP, ICMP czy BGP. Istnieją jednak również niszowe protokoły, które pełnią specjalistyczne funkcje w określonych zastosowaniach. Poniżej przedstawiono wybrane przykłady takich protokołów.

SCTP – Stream Control Transmission Protocol

SCTP (Stream Control Transmission Protocol) to protokół transportowy, który łączy cechy TCP i UDP. Oferuje niezawodność oraz mechanizmy wielościeżkowości (multi-homing), co czyni go bardziej odpornym na awarie łącza.

Cechy SCTP:

- **Obsługa wielu strumieni danych** – SCTP pozwala na jednoczesną transmisję wielu niezależnych strumieni w jednej sesji.
- **Mechanizm wielościeżkowości (multi-homing)** – możliwość korzystania z wielu adresów IP, co zwiększa odporność na awarie sieciowe.
- **Brak ataku reset (RST flood)** – w przeciwieństwie do TCP, SCTP stosuje inny mechanizm nawiązywania i kończenia połączenia.

Zastosowania SCTP:

- Sygnalizacja w sieciach telekomunikacyjnych (np. w protokole SIGTRAN do przesyłania sygnałów SS7 przez IP).
- Systemy krytyczne wymagające wysokiej niezawodności, np. infrastruktura lotnicza.

DCCP – Datagram Congestion Control Protocol

DCCP to niszowy protokół transportowy, który oferuje kontrolę przeciążenia przy jednoczesnym zachowaniu transmisji w trybie bezpołączeniowym (jak UDP).

Cechy DCCP:

- **Brak retransmisji** – w przeciwieństwie do TCP, DCCP nie retransmituje pakietów, co jest korzystne dla aplikacji czasu rzeczywistego.
- **Kontrola przeciążenia** – dostosowuje szybkość transmisji do warunków sieciowych, minimalizując ryzyko przeciążenia łącza.

Zastosowania DCCP:

- Transmisja multimediów strumieniowych (VoIP, wideo na żywo).
- Gry online, gdzie retransmisja pakietów TCP powodowałaby nadmierne opóźnienia.

LISP – Locator/ID Separation Protocol

LISP to protokół warstwy sieciowej, który rozwiązuje problem zmiany adresacji IP przy mobilności hostów. Opracowany przez Cisco, LISP oddziela tożsamość urządzenia od jego lokalizacji w sieci.

Cechy LISP:

- **Oddzielenie identyfikatora od lokalizacji** – hosty mogą zmieniać adresy IP bez konieczności aktualizacji routingu globalnego.
- **Redukcja rozgłoszeń BGP** – mniejsze obciążenie globalnej tablicy routingu.

Zastosowania LISP:

- Mobilność IP w systemach IoT i sieciach kampusowych.
- Ułatwienie wdrożeń IPv6 w środowiskach hybrydowych.

QUIC – Quick UDP Internet Connections

QUIC to nowoczesny protokół transportowy opracowany przez Google, który działa nad UDP, eliminując wiele problemów TCP.

Cechy QUIC:

- **Szybsze ustanawianie połączenia** – brak konieczności negocjacji 3-way handshake jak w TCP.
- **Wbudowane szyfrowanie** – wszystkie transmisje w QUIC są domyślnie szyfrowane.
- **Ochrona przed opóźnieniami retransmisji** – QUIC lepiej radzi sobie z utratą pakietów niż TCP.

Zastosowania QUIC:

- Transmisja danych w Google Chrome i YouTube.
- Szybsze ładowanie stron WWW (protokół HTTP/3).

EtherIP – tunelowanie Ethernetu przez IP

EtherIP (Ethernet over IP) to lekki protokół tunelowania, który umożliwia enkapsulację ramek Ethernet w pakietach IP.

Cechy EtherIP:

- **Minimalny narzut nagłówka** – bardziej efektywny niż GRE czy VXLAN.
- **Proste wdrożenie** – obsługiwany natywnie w systemach BSD i niektórych routerach.

Zastosowania EtherIP:

- Rozszerzanie sieci LAN przez internet.
- Połączenia między sieciami VLAN przez sieci IP.

ROHC – Robust Header Compression

ROHC (Robust Header Compression) to protokół optymalizujący transmisję w sieciach o ograniczonej przepustowości, takich jak sieci komórkowe.

Cechy ROHC:

- **Kompresja nagłówków IP, TCP/UDP** – redukcja narzutu w sieciach mobilnych.
- **Adaptacyjne dopasowanie kompresji** – mechanizm pozwalający na optymalizację w zmiennych warunkach sieciowych.

Zastosowania ROHC:

- Sieci LTE/5G w celu zmniejszenia obciążenia transmisji VoIP.
- Wojskowe i kosmiczne sieci komunikacyjne.

Ciekawostki

W tej sekcji przedstawiono interesujące technologie i rozwiązania sieciowe, które mają szerokie zastosowanie w komunikacji przewodowej i bezprzewodowej.

DOCSIS – standard szerokopasmowej transmisji przez sieci kablowe

DOCSIS (Data Over Cable Service Interface Specification) to standard transmisji danych przez sieci telewizji kablowej. Pozwala na świadczenie usług internetowych za pomocą infrastruktury koncentrycznej używanej pierwotnie do transmisji telewizji.

Zasada działania:

- DOCSIS wykorzystuje modulację QAM (np. 64-QAM, 256-QAM) do transmisji danych.
- Podział na kanały downstream (odbiór) i upstream (nadawanie), gdzie downstream działa w pasmach 50-1000 MHz, a upstream w 5-85 MHz.
- Współdzielenie medium transmisyjnego – dane z wielu użytkowników przesyłane są w jednym kanale, a dostęp do niego jest regulowany przez mechanizmy harmonogramowania.

Zastosowania:

- Internet szerokopasmowy dla użytkowników domowych i firmowych.
- Integracja usług VoIP i telewizji interaktywnej (IPTV).

WDS i NSTREAM – rozszerzenia sieci Wi-Fi

WDS (Wireless Distribution System) umożliwia łączenie punktów dostępowych w sieci Wi-Fi bez

potrzeby użycia kabli. Wykorzystuje MAC bridging, co pozwala na transparentne rozszerzanie sieci.

NSTREAM to technologia firmy MikroTik, zwiększająca wydajność połączeń bezprzewodowych poprzez agregację pakietów i eliminację opóźnień związanych z protokołem 802.11.

Zastosowania:

- Tworzenie mostów bezprzewodowych między budynkami.
- Zwiększanie przepustowości w sieciach punkt-punkt i punkt-wielopunkt.

Power Line Communication (PLC) – transmisja danych przez sieć elektryczną

Technologia PLC pozwala na przesyłanie danych przez istniejące linie elektryczne. Wykorzystuje pasmo do 300 MHz i modulację OFDM, umożliwiając transmisję z prędkością do kilku Gb/s.

Zastosowania:

- Internet domowy w miejscach bez infrastruktury Ethernet.
- Inteligentne sieci energetyczne (Smart Grid) – monitorowanie i zarządzanie zużyciem energii.

Internet satelitarny – LEO, MEO, GEO

Podział satelitów:

- **LEO (Low Earth Orbit)** – wysokość 300-2000 km, małe opóźnienia (<50 ms), np. Starlink.
- **MEO (Medium Earth Orbit)** – wysokość 2000-35000 km, stosowane w nawigacji GPS.
- **GEO (Geostationary Orbit)** – 35786 km, duże opóźnienia (600 ms), stosowane w telekomunikacji.

Zastosowania:

- Dostęp do internetu w odległych regionach.
- Komunikacja w lotnictwie i marynarce.

MPLS – wieloprotokołowe przełączanie etykiet

MPLS (Multiprotocol Label Switching) to technologia stosowana w sieciach szkieletowych, umożliwiająca szybsze przesyłanie pakietów poprzez nadawanie im etykiet zamiast tradycyjnego routingu IP.

Zasada działania:

- Pakiety otrzymują etykiety (*labels*), które określają trasę w sieci.
- Routery MPLS (Label Switching Routers – LSR) przekazują pakiety na podstawie etykiet, zamiast analizować nagłówki IP.

Zastosowania:

- Sieci VPN klasy operatorskiej (L3VPN, L2VPN).
- Inżynieria ruchu (Traffic Engineering) w dużych sieciach operatorskich.

Łącza światłowodowe pod oceanem i Long Fat Networks (LFN)

Podmorskie kable światłowodowe stanowią podstawę globalnej komunikacji. Stosuje się w nich amplifikatory optyczne co 50-100 km w celu kompensacji tłumienia sygnału.

Long Fat Networks (LFN) to sieci o wysokiej przepustowości i dużym opóźnieniu, np. transoceaniczne łącza światłowodowe. Ze względu na efekt Bandwidth-Delay Product (BDP) wymagają specjalnych mechanizmów TCP, takich jak TCP CUBIC czy BBR.

Zastosowania:

- Transmisja danych między kontynentami.
- Globalne sieci finansowe wymagające niskich opóźnień.

WiMAX – szerokopasmowa sieć bezprzewodowa

WiMAX (Worldwide Interoperability for Microwave Access) to technologia szerokopasmowej komunikacji bezprzewodowej oparta na standardzie IEEE 802.16. Oferuje dużą przepustowość i szeroki zasięg, stanowiąc alternatywę dla kablowych i światłowodowych sieci dostępowych.

Charakterystyka i architektura WiMAX

WiMAX działa w pasmach licencjonowanych (2,3 GHz, 2,5 GHz, 3,5 GHz) oraz nielicencjonowanych (5,8 GHz). Jego architektura składa się z następujących elementów:

- **Stacje bazowe (BS - Base Station)** – centralne punkty sieci, obsługujące użytkowników końcowych w zasięgu do 50 km.
- **Stacje abonentów (SS - Subscriber Station)** – urządzenia klienckie, takie jak modemy WiMAX, odbierające sygnał od stacji bazowych.
- **Rdzeń sieci (Backhaul)** – łącza łączące stacje bazowe z infrastrukturą szkieletową dostawcy usług internetowych.

Zasada działania i transmisja danych

WiMAX wykorzystuje techniki poprawiające efektywność transmisji:

- **Modulacja OFDM (Orthogonal Frequency Division Multiplexing)** – podział pasma na wiele nośnych, co zwiększa odporność na zakłócenia.
- **QoS (Quality of Service)** – różnicowanie ruchu dla VoIP, transmisji wideo i standardowego dostępu do internetu.

- **Technologie MIMO (Multiple Input Multiple Output)** – zwiększenie przepustowości poprzez wykorzystanie wielu anten nadawczych i odbiorczych.

Zastosowania WiMAX

WiMAX znalazł zastosowanie w różnych dziedzinach telekomunikacji:

- **Dostęp do internetu** – bezprzewodowy internet szerokopasmowy dla terenów wiejskich i trudno dostępnych.
- **Sieci korporacyjne** – połączenia między oddziałami firm bez potrzeby instalacji kabli światłowodowych.
- **Łączność w sytuacjach awaryjnych** – szybkie wdrażanie łączności w przypadku katastrof naturalnych.

WiMAX vs. LTE – dlaczego WiMAX przegrał?

Chociaż WiMAX był obiecującą technologią, został wyparty przez LTE z kilku powodów:

- **Większe wsparcie operatorów dla LTE** – globalni dostawcy infrastruktury skupili się na LTE, co ograniczyło rozwój WiMAX.
- **Lepsza kompatybilność LTE z sieciami GSM/UMTS** – LTE było naturalnym następcą 3G, podczas gdy WiMAX wymagał osobnej infrastruktury.
- **Większa efektywność LTE w zakresie mobilności** – LTE lepiej radzi sobie ze zmianami komórek przy dużych prędkościach ruchu użytkowników.

Obecnie WiMAX jest w dużej mierze zastąpiony przez LTE i 5G, ale wciąż znajduje niszowe zastosowania w niektórych krajach i sektorach przemysłowych.

99

P. Baran, *On Distributed Communications*, RAND Corporation, 1962. E. Krol, *The Whole Internet User's Guide and Catalog*, O'Reilly Media, 1993. V. Cerf, R. Kahn, *A Protocol for Packet Network Intercommunication*, IEEE Transactions on Communications, 1974. R. Metcalfe, *Ethernet: Distributed Packet Switching for Local Computer Networks*, Communications of the ACM, 1973. R. Tomlinson, *The First Email Message*, BBN Technologies, 1971. NASK, *Historia Internetu w Polsce*, Naukowa i Akademicka Sieć Komputerowa, 1991.

Lokalna sieć komputerowa, pl.wikipedia.org/wiki/Lokalna_sie%C4%87_komputerowa, dostęp: 25 lutego 2025.

Zintegrowana Platforma Edukacyjna, *Rodzaje sieci komputerowych ze względu na zasięg*, zpe.gov.pl/a/rodzaje-sieci-komputerowych-ze-wzgledu-na-zasieg/D26Fh8pbt, dostęp: 25 lutego 2025. Wikipedia, *Internet*, pl.wikipedia.org/wiki/Internet, dostęp: 25 lutego 2025

R. Metcalfe, D. Boggs, *Ethernet: Distributed Packet Switching for Local Computer Networks*, Communications of the ACM, Vol. 19, No. 7, 1976. IEEE, *IEEE 802.3 Standard for Ethernet*, 2018. A. S. Tanenbaum, D. J. Wetherall, *Computer Networks*, 5th Edition, Pearson, 2011. J. F. Kurose, K. W. Ross, *Computer Networking: A Top-Down Approach*, 7th Edition, Pearson, 2017. C. Hornig, *Standard for the Transmission of IP Datagrams over Ethernet Networks (RFC 894)*, Internet Engineering Task Force (IETF), 1984. Dostępne: https://tools.ietf.org/html/rfc894.

IEEE, *IEEE 802.3 Standard for Ethernet*, 2018.

IEEE, *IEEE 802.11 Standard for Wireless LAN*, 2016.

Bluetooth Special Interest Group, *Bluetooth Core Specification*, 2020.

ITU-T, *G.992.5: Asymmetric Digital Subscriber Line (ADSL) Transceivers*, 2003.

IEEE, *IEEE 802.3 Standard for Ethernet*, 2018. IEEE, *IEEE 802.11 Standard for Wireless LAN*, 2016. W. Stevens, *PPP: The Point-to-Point Protocol*, RFC 1661, 1994. ISO, *ISO/IEC 13239:2002 High-Level Data Link Control (HDLC)*, 2002.

https://vijayababuj.wordpress.com/wp-content/uploads/2015/08/network_devices.jpg.

Wireshark wiki, https://wiki.wireshark.org/TCP_3_way_handshaking.

Network address translation, https://en.wikipedia.org/wiki/Network_address_translation, dostęp: 25 lutego 2025.