Wprowadzenie do transmisji sieciowej – opracowanie

Opracowano na podstawie materiałów dostępnych w sieci. Wybór źródeł.

# Wprowadzenie do transmisji

Urządzenia w sieci komputerowej komunikują się ze sobą i wymieniają informacje. Wymiana informacji musi przebiegać w ściśle określony sposób umożliwiający przesyłanie danych. Każde z komunikujących się urządzeń musi przestrzegać pewnych ustalonych zasad i reguł postępowania. Zbiór zasad i norm, których muszą przestrzegać komunikujące się ze sobą urządzenia, nazywamy **protokołem komunikacyjnym.**Komunikacja pomiędzy urządzeniami może przebiegać w trybie **połączeniowym (***connection oriented)*lub **bezpołączeniowym (***connectionless oriented).*

Tryb **połączeniowy**polega na ustanowieniu logicznego połączenia pomiędzy dwoma komunikującymi się ze sobą urządzeniami. Aby rozpocząć komunikację, należy najpierw nawiązać połączenie. Z trybu połączeniowego korzysta się wtedy, gdy powstaje potrzeba przesyłania wielu komunikatów w obu kierunkach, np. podczas korzystania z usługi telnet. W trybie **bezpołączeniowym**komunikaty przekazywane są niezależnie, np. przekazywanie wiadomości za pomocą poczty elektronicznej.

Półdupleks (half-duplex)

To transmisja dwukierunkowa, naprzemienna. W danym momencie jest ustalony tylko jeden kierunek transmisji, a urządzenie może albo nadawać, albo odbierać informacje. Dla odwrócenia kierunku transmisji potrzebny jest system sygnalizacji wskazujący, że urządzenie ukończyło nadawanie i może odbierać informacje. Przykładem jest amatorska stacja krótkofalowa lub radio CB.

Dupleks (full-duplex)

To transmisja jednoczesna i dwukierunkowa. Wymaga zazwyczaj dwóch par przewodów dla sieci cyfrowych. Dla połączeń analogowych dla jednej pary przewodów szerokość pasma dzielona jest na dwie części. Przykładem jest rozmowa telefoniczna.

Aby usługi i aplikacje sieciowe mogły prawidłowo działać, muszą mieć zapewnione odpowiednie warunki. Najważniejszymi warunkami są:

• niezawodność przesyłania danych,

• przepustowość łączy,

• czas odpowiedzi.

Aby działać bez zakłóceń, niektóre aplikacje muszą być w stanie transmitować lub odbierać dane z określoną prędkością. Na przykład aplikacja odtwarzająca pliki multimedialne musi odbierać taką ilość danych, która jest niezbędna do wyświetlania filmu lub odtwarzania muzyki. Aplikacja taka wymaga określonej przepustowości. Jeżeli warunek ten nie będzie spełniony, to aplikacja nie będzie działała prawidłowo (lub w ogóle się nie uruchomi) albo będzie zmuszona pracować przy innych parametrach, np. rozdzielczości obrazu lub jakości dźwięku.

# ISO/OSI

Model sieci OSI (ang. *Open System Interconnection*) został opracowany i wdrożony przez ISO - Międzynarodową Organizację Normalizacyjną (ang. *International Organization for Standardization*) na początku lat osiemdziesiątych.  Standard ten jest traktowany jako model odniesienia (wzorzec) dla większości rodzin protokołów komunikacyjnych. Podstawowym założeniem modelu jest podział systemów sieciowych na 7 warstw (ang. *layers*) współpracujących ze sobą w ściśle określony sposób.

Model OSI dzieli procesy zachodzące podczas sesji komunikacyjnej na siedem warstw funkcjonalnych, które zorganizowane są według naturalnej sekwencji zdarzeń zachodzących podczas sesji komunikacyjnej. Warstwy od 1 do 3 umożliwiają dostęp do sieci, a warstwy od 4 do 7 obsługują logistycznie komunikację końcową.

Model ten ilustruje procesy zachodzące w czasie komunikacji od nadawcy do odbiorcy. W warstwach: aplikacji prezentacji i sesji przygotowywane są dane do transportu poprzez sieć. W warstwie sesji zostaje nawiązana sesja transmisyjna. W warstwie transportowej dane zostają podzielone na mniejsze porcje – SEGMENTY i w procesie encapsulacji każdy segment otrzymuje nagłówek z numerami portów. Dalej segmenty danych przechodząc do warstwy sieci otrzymują kolejny nagłówek z adresami IP hosta źródłowego i docelowego – powstaje pakiet, który jest routowany za pomocą routerów do sieci docelowej. Kolejnym etapem encapsulacji jest utworzenie ramki poprzez dodanie kolejnego nagłówka z Mac adresami: źródłowym karty sieciowej hosta i docelowym interfejsu do którego ramka zostanie przekserowana. Ramki zostają następnie skierowane do warstwy fizycznej i tam skierowana do medium transmisyjnego, w którym odbywa się transmisja ramek bit po bicie do odbiorcy.

Po stronie odbiorcy odbywa się zupełnie analogiczny proces odwrotny – dekapsulacji, czyli odczytywania adresów docelowych z nagłówków, przekierowywania ramek, pakietów i segmentów do hosta docelowego. Jednocześnie zdejmowane są kolejne nagłówki, odczytywane adresy docelowe: MAC, IP i PORTÓW aż dane trafią do aplikacji u odbiorcy. Adres portu docelowego jednoznacznie identyfikuje aplikację u odbiorcy a zatem i rodzaj danych.

Funkcje kolejnych warstw są następujące:

**(7) Warstwa aplikacji**– pełni rolę interfejsu pomiędzy aplikacjami użytkownika a usługami w sieci. Inicjuje ona proces komunikacji

**(6) Warstwa prezentacji**– odpowiedzialna jest za sposób kodowania danych. Może też być wykorzystana do szyfrowania i rozszyfrowywania wiadomości

**(5) Warstwa sesji**– inicjuje sesję transmisyjną pomiędzy nadawcą i odbiorcą

**(4) Warstwa transportowa**–dzieli dane na segmenty i rozpoczyna proces transmisji, a u odbiorcy ustala właściwą  kolejność odebranych segmentów i wskazuje za pomocą numeru portów w nagłówkach segmentów która aplikacja będzie prezentować odebrane dane.

**(3) Warstwa sieci** -routuje pakiety, tzn. jest odpowiedzialna za określenie trasy transmisji pomiedzy hostem nadawcy i hostem odbiorcy

**(2) Warstwa łącza danych** – jest odpowiedzialna za utworzenie ramek i przygotowuje je do transmisji przez sieć. Jest też odpowiedzialna za kontrolę transmisji i detekcję błędów za pomocą sumy kontrolnej. U odbiorcy warstwa jest odpowiedzialna za ukształtowanie ramek po odebraniu strumienia bitów.

**(1) Warstwa fizyczna** jest odpowiedzialna za przesyłanie strumieni bitów bit po bicie przez sieć, a u odbiorcy za odbiór strumieni bitów i przekazanie ich do warstwy łącza danych gdzie zostaną na nowo ukształtowane ramki.

# Model TCP/IP

Oprócz modelu OSI istnieją także inne modele sieci, z których najbardziej popularny jest **TCP/IP,**który  powstał na zamówienie Departamentu Obrony USA. Model ten jest powszechnie stosowany, między innymi w sieci Internet oraz ściśle związany ze stosem protokołów TCP/IP. Podobnie jak w modelu OSI możemy w nim wyróżnić warstwy, jednak funkcje są różne, pomimo iż niektóre z nich posiadają jednakowe nazwy. Model TCP/IP składa się z czterech warstw. **Warstwa aplikacji**obejmuje funkcje trzech najwyższych warstw modelu OSI (aplikacji, prezentacji i sesji). Użytkownicy uruchamiają programy, które uzyskują dostęp do usług za pośrednictwem protokołu na poziomie warstwy transportu i wysyłają lub odbierają dane w postaci pojedynczych komunikatów lub strumienia bajtów. Programy użytkowe przekazują do warstwy transportowej dane w wymaganym formacie, aby mogły one zostać dostarczone w odpowiednie miejsce. W warstwie tej działa wiele protokołów aplikacji, miedzy innymi http, ftp, telnet, ssh, smtp, pop3 itp.

Podstawowym zadaniem **warstwy transportowej**jest zapewnienie komunikacji między programami użytkownika. Warstwa ta może zarządzać przepływem informacji oraz zapewniać niezawodność przesyłania przez porządkowanie segmentów danych i retransmisję uszkodzonych lub zagubionych segmentów.

W komputerze może działać wiele aplikacji wymieniających dane w sieci przy wykorzystaniu portów określonych dla każdego połączenia i nie nastąpi wymieszanie się przesyłanych przez nie danych. Warstwa transportowa dzieli strumień danych na segmenty, a w nagłówku umieszcza numer portu identyfikujący aplikację wysyłającą lub odbierającą dane.

W warstwie tej działa **protokół połączeniowy TCP**oraz **bezpołączeniowy UDP.**

**Warstwa internetowa**przyjmuje segmenty z warstwy transportowej razem z informacjami identyfikującymi odbiorcę. Zadaniem jej jest wysyłanie pakietów i dostarczenie ich do miejsca przeznaczenia, niezależnie od trasy, po której będą przesyłane. Protokołem zarządzającym tą warstwą jest **protokół IP.**Warstwa dzieli dane na pakiety, dodaje nagłówek zawierający między innymi adres IP nadawcy i odbiorcy. Na podstawie adresu IP miejsca docelowego podejmowana jest decyzja, czy wysłać pakiet wprost do odbiorcy w sieci lokalnej, czy też do routera, który przekaże go do odpowiedniego interfejsu sieciowego. Routery pracujące w niej wyznaczają najlepsze trasy do miejsca przeznaczenia pakietów.

**Warstwa dostępu do sieci**odbiera pakiety IP i przesyła je przez daną sieć. Zapewnia interfejs z siecią fizyczną i zajmuje się przekazywaniem danych przez fizyczne połączenia między urządzeniami sieciowymi. Najczęściej są to karty sieciowe lub modemy. Formatuje dane do transmisji przez nośnik oraz adresuje dane do podsieci, opierając się na adresach fizycznych. Zapewnia sprawdzanie błędów przesyłu danych za pomocą sumy kontrolnej ramki.

W modelu TCP/IP sieci lokalne w warstwie dostępu do sieci budowane są w oparciu o standard Ethernet. W sieciach rozległych WAN w tej warstwie stosowane są różne technologie, np. połączenia modemowe, DSL, Frame Relay, ATM. Na styku pomiędzy warstwą internetową i warstwą dostępu do sieci działa protokół **ARP***(Address Resolution Protocol),*który pozwala na ustalenie adresu sprzętowego MAC hosta, gdy dany jest adres warstwy sieciowej IP. Z protokołu tego korzystamy podczas wysyłania danych.

Podczas komunikacji urządzeń w sieci dane muszą przejść wszystkie etapy enkapsulacji. W nagłówku pakietu urządzenie nadawcze umieszcza adres IP nadawcy oraz odbiorcy. Adres nadawcy jest przydzielony każdemu urządzeniu, adres odbiorcy jest wprowadzany przez użytkownika w postaci adresu **IP**lub nazwy domenowej komputera. Adresy te są więc znane i urządzenie może utworzyć pakiet. W nagłówku ramki potrzebny jest adres MAC nadawcy i odbiorcy. Każda karta sieciowa posiada unikatowy adres MAC, więc urządzenie „zna" swój adres, brakuje jeszcze adresu MAC urządzenia odbiorcy. Do ustalenia tego właśnie adresu wykorzystywany jest protokół ARP. Gdy komputer chce skorzystać z protokołu ARP, przygotowuje specjalny pakiet zapytania ARP, który jest wysyłany na adres rozgłoszeniowy, dzięki czemu dociera do wszystkich urządzeń w sieci lokalnej. Urządzenie o szukanym adresie sieciowym odpowiada, przesyłając pakiet z odpowiedzią zawierającą adres sprzętowy MAC. Komputer dysponuje już wszystkimi adresami i może przygotować ramkę. Aby uniknąć konieczności wysyłania kolejnego zapytania ARP, komputer zapisuje sobie w specjalnej tablicy informacje o adresach urządzeń, z którymi się komunikował.

Protokół odwrotny **RARP***(Reverse Address Resolution Protocol)*pozwala na ustalenie adresu IP na podstawie adresu fizycznego MAC.

## Protokoły warstwy sieciowej

**Protokół IP***(Internet Protocol)*jest odpowiedzialny za przesyłanie pakietów pomiędzy użytkownikami sieci. Jest protokołem bezpołączeniowym, co oznacza, że w trakcie transmisji nie sprawdza się poprawności pakietów przesyłanych przez sieć. Nie ma zatem gwarancji ich dostarczenia, ponieważ mogą one zostać po drodze zagubione lub uszkodzone. Podstawowymi funkcjami protokołu IP jest:

• określanie i tworzenie struktury pakietu,

• określanie schematu adresowania logicznego IP,

• kierowanie ruchem pakietów w sieci.

IP jest protokołem zawodnym. Jedynym kryterium pozwalającym sprawdzić poprawność przesyłania jest suma kontrolna nagłówka zawarta w polu Header Checksum. Jeżeli w trakcie transmisji został odkryty błąd, to pakiet jest niszczony przez stację, która wykryła niezgodność. W takim przypadku nie ma żadnych powtórek transmisji i kontroli przepływu danych. Nagłówek protokołu IP jest wykorzystywany do transportu danych między urządzeniem źródłowym i docelowym.

Ściśle związany z protokołem IP jest protokół ICMP (*Internet Control Message Protocol).*Protokół IP, jako bezpołączeniowy, nie posiada mechanizmów informowania o błędach w funkcjonowaniu sieci IP oraz diagnostyki sieci. Do tego celu przeznaczony jest protokół ICMP. Umożliwia on przesyłanie między komputerami lub routerami informacji o błędach występujących w funkcjonowaniu sieci IP. Najczęściej używanymi poleceniami korzystającymi w protokołu ICMP są ping i tracert. Ping jest to program używany w sieciach komputerowych działających w oparciu o protokół TCP/IP, służący do diagnozowania połączeń sieciowych. Pozwala na sprawdzenie, czy istnieje połączenie pomiędzy hostem testującym i testowanym oraz określenie jakości połączenia przez pomiar liczby zgubionych pakietów oraz czasu potrzebnego na ich transmisję. Ping wysyła pakiety żądania echa ICMP (*Echo Request)*i odbiera odpowiedzi na żądanie echa ICMP *(Echo Reply).*Jako argument dla polecenia ping można podać adres IP lub nazwę domenową komputera testowanego

Komputery powinny odpowiadać na żądanie echa, lecz większość współczesnych programów typu firewall blokuje ten proces, w konsekwencji czego możemy nie otrzymać odpowiedzi, mimo że istnieje połączenie pomiędzy hostami. Program tracert (w systemach Linux program nazywa się traceroute) jest przeznaczony do śledzenia trasy, po jakiej przesyłane są pakiety w sieci. Program ten wysyła pakiet żądania echa z polem TTL (*Time To Live)*ustawionym na kolejne wartości, od 1 do 30. Wartość TTL jest zmniejszana przy przechodzeniu przez kolejne routery na trasie. Jeżeli pole TTL osiągnie wartość 0, pakiet jest kasowany przez router.

Router dodatkowo wysyła za pomocąprotokołu ICMP informację zwrotną o błędzie. Komputer źródłowy uzyskuje, bezpośrednio po wysłaniu żądania o wartości 1, adres IP pierwszego routera na trasie. W następnym pakiecie pole TTL ma wartość 2, co powoduje, że pierwszy router zmniejszy tę wartość do 1, a drugi router zmniejszy TTL do 0 i skasuje pakiet wysyłając komunikat o błędzie. W ten sposób program tracert może prześledzić trasę w sieci zawierającej nie więcej niż 30 routerów. Brak odpowiedzi na zadany pakiet sygnalizowany jest znakiem gwiazdki „\*" i może wynikać z konfiguracji firewalla lub przeciążenia sieci.

## Protokoły warstwy transportowej

W warstwie transportowej w stosie protokołów TCP/IP może działać protokół połączeniowy TCP lub protokół bezpołączeniowy UDP.

Protokół **TCP (***Transmission Control Protocol)*działa w warstwie transportowej w trybie połączeniowym. Korzystanie z trybu połączeniowego umożliwia zagwarantowanie dostarczenia danych do odbiorcy. Połączenia TCP są połączeniami wirtualnymi, rozpoznawanymi po adresach i portach urządzeń docelowych i źródłowych. Połączenia takie charakteryzują się możliwościami sterowania przepływem, potwierdzaniem odbioru, zachowywaniem kolejności danych, kontrolą błędów i przeprowadzaniem retransmisji. Segmenty TCP składają się z nagłówka i danych.

Najważniejszymi polami nagłówka TCP są: port źródłowy, port docelowy, numer sekwencyjny, numer potwierdzenia, okno.

Ponieważ na komputerze posiadającym jeden adres IP może jednocześnie działać wiele aplikacji, to do ich identyfikacji wykorzystuje się porty. Porty reprezentowane są przez liczby naturalne z zakresu od 0 do 65535. Numery portów od 0 do 1023 są ogólnie znanei zarezerwowane dla usług, np. WWW korzysta z portu 80, a telnet z portu 23. Dzięki portom możemy określić, dla jakiej aplikacji przeznaczony jest segment danych (port docelowy), lub z którego portu wysłano dane (port źródłowy). Komunikacja między aplikacjami może się odbywać za pomocą gniazd *(socket).*Gniazdo to kombinacja adresu IP i numeru portu. Gniazdo jednoznacznie określa proces w sieci lub zakończenie logicznego łącza komunikacyjnego między dwiema aplikacjami. Jeśli aplikacje uruchomione są na dwóch różnych komputerach, to para odpowiadających im gniazd definiuje połączenie. Gniazdo możemy traktować jako kanał komunikacyjny - jeden program wpisuje do niego dane, a drugi je odbiera. Serwer otwiera gniazdo i oczekuje na połączenie. Klient łączący się z otwartym gniazdem musi znać sieciowy adres komputera oraz numer portu. Każdy przesyłany segment danych oznaczany jest kolejnym numerem sekwencyjnym. Przed rozpoczęciem transmisji nadawca i odbiorca wymieniają między sobą numery sekwencyjne. Odbiorca wiadomości na podstawie numeru sekwencyjnego ustala kolejność segmentów oraz sprawdza, czy wszystkie segmenty dotarły do miejsca przeznaczenia. Potwierdzenie odebrania segmentu polega na wysłaniu przez odbiorcę numeru kolejnego segmentu, który powinien być przesłany. Na przykład jeżeli ostatni poprawnie odebrany segment miał numer 123, to odbiorca wyśle numer potwierdzenia 124 (numer następnego segmentu, który ma być przesłany). Potwierdzenie wysyłane jest po odebraniu pewnej liczby danych określonych w polu okno. Jeżeli w sieci występuje dużo błędów, to wielkość okna jest zmniejszana, aby częściej otrzymywać potwierdzenia i przez to zmniejszyć liczbę segmentów danych wymagających retransmisji. Jeżeli liczba błędów się zmniejsza, to rozmiar okna jest powiększany, aby zapewnić większą przepustowość sieci.

**Protokół UDP***(User Datagram Protocol)*działa w warstwie transportowej w trybie bezpołączeniowym. Protokół ten nie gwarantuje dostarczenia danych do odbiorcy. Jeżeli pakiet nie dotrze do odbiorcy, lub dotrze uszkodzony, UDP nie podejmie żadnych działań zmierzających do retransmisji danych, a zapewnienie niezawodności pozostawi warstwie wyższej. Nagłówek protokołu UDP jest prostszy niż TCP. Protokół wykorzystywany jest do szybkiego przesyłania danych w niezawodnych sieciach. Dzięki temu, że istnieją dwa alternatywne względem siebie protokoły w warstwie transportowej, TCP i UDP, możliwy jest dobór przez aplikacje odpowiedniego dla siebie rozwiązania.