Protokoły komunikacyjne

Protokół komunikacyjny to zbiór zasad i norm, których musza przestrzegać komunikujące się ze sobą obiekty.
Protokół komunikacyjny, wspólny dla współdziałających rozmówców, musi być określony w sposób jednoznaczny, wykluczający możliwość jakichkolwiek niejasności i nieporozumień.

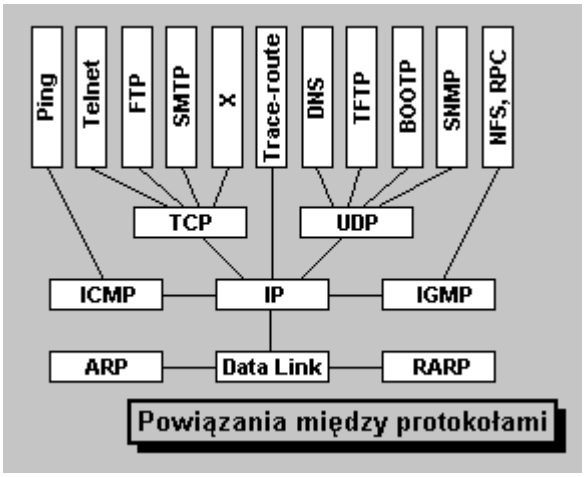
Wśród ogromnej liczby protokołów wykorzystywanych do komunikacji w sieciach, na szczególną uwagę zasługują protokoły z rodziny TCP/IP. Jest to rodzina protokołów, na których opiera się wiele sieci lokalnych oraz Internet. W skład tej rodziny wchodzą między innymi protokoły: IP, ARP, ICMP, UDP, TCP.

IP – Internet Protocol

Podstawowym protokołem w rodzinie TCP/IP jest protokół IP. Definiuje on identyfikację komputerów niezależną od sprzętu oraz steruje przepływem pakietów przez sieć. Podstawową informacją identyfikującą komputer na poziomie fizycznym jest adres sprzętowy w przypadku TCP/IP jest to 32-bitowy adres IP (lub 128-bitowy w IPv6). W odróżnieniu od adresu sprzętowego adres IP jest adresem niezależnym od sprzętu, nadawanym i przechowywanym przez oprogramowanie.

Powszechnie stosowaną wersją protokołu IP jest wersja 4. Jednak ze względu na ograniczenia dotyczące adresowania logicznego spowodowane niedostateczną, w stosunku do potrzeb, liczbą bitów przeznaczonych na adres IP protokół ten będzie zastąpiony nowszą wersją IPv6.

Protokół IP implementuje warstwę intersieci w modelu warstw oprogramowania sieciowego TCP/IP. Większość protokołów z rodziny TCP/IP korzysta w sposób pośredni lub bezpośredni z protokołu IP.



Protokół IP nie posiada mechanizmów sygnalizujących błędy oraz mechanizmów umożliwiających kontrolowanie przepływu pakietów. Z tego względu zgłaszaniem problemów z przesyłaniem datagramów oraz sterowaniem zajmuje się protokół ICMP.
Innym protokołem, który umożliwia bardziej efektywne rozsyłanie pakietów jest protokół IGMP. Protokół ten działa w oparciu o adresy rozsyłania grupowego.

IP został opracowany do działania w sytuacjach ekstremalnych, np. w trakcie wojny. W normalnych warunkach jego funkcja sprowadza się do wyboru optymalnej trasy i przesyłania nią pakietów. W przypadku wystąpienia awarii, na którymś z połączeń protokół będzie próbował dostarczyć pakiety trasami alternatywnymi (nie zawsze optymalnymi). Protokół IP jest podstawowym protokołem przesyłania pakietów w Internecie.

Jest on protokołem bezpołączeniowym. W celu przesłania pakietów nie jest nawiązywane połączenie z hostem docelowym. Pakiety mogą być przesyłane różnymi trasami do miejsca przeznaczenia, gdzie są następnie składane w całość.

Do przesyłania danych protokół IP używa specjalnego formatu pakietu. Pakiet ten składa się z nagłówka pakietu oraz danych do przesłania. Zgodnie z zasadą przesyłania strumieniowego dane protokołu IP są danymi pochodzącymi z wyższych warstw modelu ISO/OSI. Dane te są następnie enkapsulowane do postaci pakietu IP. Przy przejściu do warstwy łącza danych pakiet IP jest enkapsulowany do postaci ramki Ethernetowej.

Każdy adres można traktować jako parę (identyfikator sieci, identyfikator maszyny). W obszarze sieci lokalnej adresy IP wszystkich komputerów charakteryzują się jednakowym prefiksem identyfikatora sieci, którego długość zależy od klasy adresu. Klasa adresu IPv4 jest określana przez pięć najstarszych bitów:

- Klasa A - pierwszy bit równy `0`, siedem bitów przeznaczonych na identyfikację sieci, 24 bity przeznaczone na adres maszyny. Klasa ta jest przeznaczona dla dużych sieci, które mają ponad 2^{16} komputerów;
- Klasa B - pierwsze dwa bity to `10`, następne 14 bitów identyfikuje sieć, 16 bitów przeznaczonych jest do identyfikacji komputera. Klasa jest przeznaczona dla sieci, w których liczba komputerów leży w przedziale 2^8 - 2^{16};
- Klasa C - pierwsze trzy bity to `110`, następne 21 bitów przeznaczanych jest na identyfikator sieci, 8 bitów identyfikuje komputer. Klasa jest przeznaczona dla sieci obejmujących mniej niż 2^8 maszyn;
- Klasa D - pierwsze cztery bity to `1110`. Klasa jest przeznaczona jest do rozsyłania grupowego (pozostałe bity to adres rozsyłania grupowego);
- Klasa E - pierwsze 5 bitów to `11110`. Jest to klasa adresów zarezerwowana na przyszłość.

IPv4

+	Bity 0 - 3	4 - 7	8 - 15	16 - 18	19 - 31
0	Wersja	Długość nagłówka	Typ usługi	Całkowita długość	
32	Numer identyfikacyjny			Flagi	Kontrola przesunięcia
64	Czas życia pakietu (TTL)		Protokół warstwy wyższej	Suma kontrolna nagłówka	
96	Adres źródłowy IP				
128	Adres docelowy IP				
160	Opcje IP				Uzupełnienie
192	Dane				

Poszczególne pola pakietu mają następujące znaczenie:

Wersja (VERS) - pole 4-bitowe określające typ protokołu IP. Jeśli jest tam wpisana wartość 4 oznacza to wersję czwartą protokołu. Jeśli jest tam wartość 6 oznacza to IPv6. Rozróżnianie pomiędzy pakietami wersji 4 i 6 jest przeprowadzane już przy analizowaniu ramki warstwy drugiej poprzez badanie pola typu protokołu.
Długość nagłówka (HLEN) - pole 4 bitowe określające długość datagramu wyrażoną jako wielokrotność słów 32 bitowych.
Typ usługi (TOS ang. Type-of-Service) - 8-bitowe pole określające poziom ważności jaki został nadany przez protokół wyższej warstwy. Znaczenie poszczególnych bitów tego pola jest następujące: pierwsze 3 bity: wartość 0 - stopień normalny, wartość 7 - sterowanie siecią czwarty bit - O - prośba o krótkie czasy oczekiwania piąty bit - S - prośba o przesyłanie danych szybkimi łączami szósty bit P - prośba o dużą pewność przesyłania danych bity 6, 7 nieużywane
Całkowita długość - pole 16-bitowe. Długość całego pakietu wyrażona w bajtach. W celu uzyskania długości pola danych należy odjąć od długości całkowitej długość nagłówka. Wartość minimalna wynosi 576 oktetów zaś maksymalna 65535 oktetów, tzn. 64 kB
Identyfikacja - 16 bitowe pole używane do określania numeru sekwencyjnego bieżącego datagramu.
Znaczniki - 3 bitowe pole. Pierwszy najbardziej znaczący ma zawsze wartość 0. Kolejne znaczące bity sterują fragmentacją (0- oznacza, czy pakiet może zostać podzielony na fragmenty, 1 - nie może być podzielony). Trzeci bit oznacza: ostatni pakiet powstały w wyniku podzielenia (jeśli ma wartość 1) lub pakiet ze środka 0.
Przesunięcie fragmentu - 13-bitowe pole służące do składania fragmentów datagramu.
Czas życia (TTL, ang. Time To Live) - 8-bitowe pole określające liczbę routerów (przeskoków), przez które może być przesłany pakiet. Wartość tego pola jest zmniejszana przy przejście przez każdy router na ścieżce. Gdy wartość tego pola wynosi 0, wtedy pakiet taki jest odrzucany. Zasada ta pozwala na stosowanie mechanizmów zapobiegających zapętlaniu się tras routingu.
Protokół - 8-bitowe pole określające, który z protokołów warstwy wyższej odpowiada za przetworzenie pola Dane. Możliwe opcje tego pola zostały przedstawione na następnych slajdach.
Suma kontrolna nagłówka - 16-bitowe pole z sumą kontrolną nagłówka pozwalającą stwierdzić, czy nie nastąpiło, naruszenie integralności nagłówka. Ze względu na fakt, że każdy router dokonuje zmian w nagłówku musi ona być przeliczona na każdym z routerów.
Adres IP nadawcy - 32-bitowe pole z adresem IP nadawcy pakietu
Adres IP odbiorcy - 32-bitowe pole z adresem IP odbiorcy pakietu
Opcje - pole to nie występuje we wszystkich pakietach. Szczegółowe wartości tego pola zostaną omówione na następnym slajdzie.
Uzupełnienie (Wypełnienie) - pole to jest wypełnione zerami i jest potrzebne, żeby długość nagłówka była wielokrotnością 32 bitów (patrz-> Długość nagłówka) Dane - pole od długości do 64kB zawierające dane pochodzące z wyższych warstw.

IPv6

Protokół IP w wersji 6 posiada adresy 128-bitowe. Dzięki polu „Następny nagłówek” (ang. Next header) jest możliwość dołączania nagłówków rozszerzających. Nagłówek ten jest umieszczany w pakiecie za nagłówkiem podstawowym, a przed nagłówkiem warstwy transportowej. Nagłówki te powinny występować w określonej kolejności natomiast nie ma ograniczenia, co do ich liczby. Nagłówki te zastępują pola opcjonalne w IPv4.

<table>
<tr><th>Bity</th><th>0-3</th><th>4-7</th><th>8-11</th><th>12-15</th><th>16-19</th><th>20-23</th><th>24-27</th><th>28-31</th></tr>
<tr><td>0</td><td>Wersja</td><td colspan="2">Priorytet</td><td colspan="5">Etykieta przepływu</td></tr>
<tr><td>32</td><td colspan="4">Długość danych</td><td colspan="2">Następny nagłówek</td><td colspan="2">Limit przeskoków</td></tr>
<tr><td>64</td><td colspan="8" rowspan="4">Adres źródłowy (128 bitów)</td></tr>
<tr><td>96</td></tr>
<tr><td>128</td></tr>
<tr><td>160</td></tr>
<tr><td>192</td><td colspan="8" rowspan="4">Adres docelowy (128 bitów)</td></tr>
<tr><td>224</td></tr>
<tr><td>256</td></tr>
<tr><td>288</td></tr>
</table>

Budowa datagramu IPv6 została przedstawiona na rysunku. Znaczenie pól jest zgodne z ich opisem:

Wersja - 4b - wersja protokołu IP, w tym przypadku 6
Priorytet / Typ ruchu (ang.Traffic Class) - 8b - pole służące do określenia priorytetu przesyłanego pakietu. Jest to szczególnie istotne w przypadku pakietów transmitujących ruch multimedialny, gdzie ważnym aspektem jest zapewnienie wysokiego poziomu obsługi (ang. Quality of service). Pole to jest odpowiednikiem pola Type of Service w IPv4.
Etykieta przepływu (ang. Flow Label) - 20b - pole to zostało zarezerwowane na potrzeby zapewnienia wysokiego poziomu obsługi. Pole to składa się z kilku podpól: pierwsze cztery bity określają wrażliwość na zmiany czasów opóźnień, bity od 8 do 15 wyznaczają priorytet, reszta bitów identyfikuje potok danych.
Długość danych (ang. Payload length) - 16b - długość pola danych wyrażona wielokrotnością oktetów.
Następny nagłówek (ang. Next Header) - 8b - pole z informacją jaki będzie nagłówek rozszerzający. Ważną cechą tego pola, która umożliwia szybsze przesyłanie pakietów przez routery jest możliwość dołączania nagłówków rozszerzających. Szczegóły tego rozwiązania zostaną przedstawione na następnym slajdzie.
Limit przeskoków (ang. Hop Limit) - 8b - Liczba przeskoków, czyli liczba routerów, przez które pakiet może być przesłany zanim dotrze do celu. Po każdym przejściu przez router wartość tego pola jest zmniejszana o 1. Po osiągnięciu wartości 0 pakiet taki jest odrzucany. Odpowiednie pole w nagłówku IPv4 miało nazwę TTL. Maksymalna wartość tego pola podobnie jak pola TTL wynosi 255.
Adres źródłowy (ang. Source Address) - 128b - adres źródłowy
Adres docelowy (ang. Destination Addres) - 128b - adres docelowy

Przykładowa implementacja struktury nagłówka w systemie Linix:

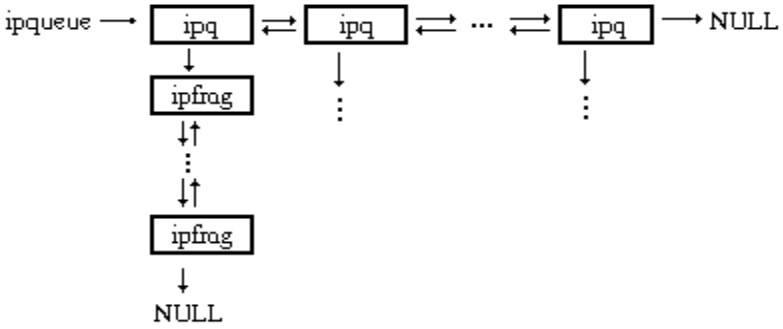
```
struct iphdr {
        __u8    ihl:4,          /* wersja i długość nagłówka  */
                version:4;
        __u8    tos;            /* typ usługi */
        __u16   tot_len;        /* całkowita długość */
        __u16   id;             /* nr identyfikacyjny */
        __u16   frag_off;       /* flagi i kontrola przesuniecia */
        __u8    ttl;            /* TTL */
        __u8    protocol;       /* protokół warstwy wyższej  */
        __u16   check;          /* suma kontrolna nagłówka */
        __u32   saddr;          /* adres źródłowy */
        __u32   daddr;          /* adres docelowy */
};
```

Oraz struktur służących do przechowywania fragmentów pakietów:

```
struct ipfrag
{
        int             offset;         /* offset fragmentu w pakiecie   */
        int             end;            /* ostatni bajt danych w pakiecie*/
        int             len;            /* dlugosc tego fragmentu        */
        struct sk_buff  *skb;           /* oryginalnie odebrany fragment */
        unsigned char   *ptr;           /* oryginalny fragment danych    */
        struct ipfrag   *next;          /* wskazniki polaczeniowe        */
        struct ipfrag   *prev;
};

struct ipq
{
        unsigned char   *mac;           /* wskaznik na naglowek sprzetowy*/
        struct iphdr    *iph;           /* wskaznik na naglowek IP       */
        int             len;            /* calkowita dlugosc
                                        rozdrobnionego pakietu          */
        short           ihlen;          /* dlugosc naglowka IP           */
        short           maclen;         /* dlugosc naglowka sprzetowego  */
        struct timer_list timer;        /* ile czasu pozostalo do
                                           przeterminowania fragmentow   */
        struct ipfrag   *fragments;     /* lista polaczonych fragmentow  */
        struct ipq      *next;          /* wskazniki polaczeniowe        */
        struct ipq      *prev;
        struct device   *dev;           /* Urzadzenie do wysylania
                                         informacji o bledzie           */
};
```

Organizacja przechowywania pakietów:



ICMP - Internet Control Message Protocol

ICMP jest protokołem służącym do przesyłania różnego rodzaju pakietów informujących o błędach i innych ważnych sytuacjach oraz do kontrolowania kondycji połączenia. Przykładem użycia może być polecenie ping.

Transportowanie pakietów ICMP jest procesem dwustopniowym. Najpierw komunikat ICMP dołącza się do pakietu IP, który jest przez Internet umieszczany i transportowany w postaci ramki. Protokół ICMP używa zawodnego sposobu komunikowania się, jak datagramy UDP. Oznacza to, że komunikaty ICMP mogą również zostać pogubione i powielone.

Datagram ICMP może zawierać jeszcze nagłówek i dane pakietu, który spowodował jego wysłanie (np. po błędzie). Pomaga to ustalić aplikację i protokół, które stały się przyczyną błędu.

Bit 0 7	Bit 8 15	Bit 16 23	Bit 24 31
Typ	Kod	Suma kontrolna	
Dane (opcjonalne)			

Najważniejsze dane przesyłane w komunikacie ICMP zawarte są w polach TYP i KOD.

Pole Typ:
0 - odpowiedź z echem (ang. Echo Reply)
3 - odbiorca nieosiągalny (ang. Destination Unreachable).
4 - zmniejszenie szybkości nadawania - tłumienie źródła (ang. source quench)
5 - zmiana trasowania - przekierowanie (ang. redirect).
8 - prośba o echo (ang. echo request)
9 - rozgłaszanie routera (ang. router advertisement)
10 - wywołanie routera (ang. router solicitation)
11 - przekroczenie TTL (ang. Time Exceeded)
12 - kłopot z parametrami datagramu
13 - prośba / żądanie o wysłanie znacznika czasu (ang. timestamp request)
14 - odpowiedź na prośbę / żądanie o wysłanie znacznika czasu (ang. timestamp reply)
15 - prośba o informację
16 - odpowiedź z informacją
17 - prośba o maskę adresu
18 - odpowiedź z maską adresu
30 - Traceroute
31 - błąd konwersji datagramu (ang. Datagram Conversion Error)
32 - przekierowanie hosta mobilnego (ang. Mobile Host Redirect)
33 - IPv6 Where-Are-You
34 - IPv6 Here-I-Am
35 - prośba o zarejestrowanie urządzenia mobilnego (ang. Mobile Registration Request)
36 - odpowiedź na prośbę o zarejestrowanie urządzenia mobilnego (ang. Mobile Registration Reply)
37 - żądanie nazw domeny (ang. Domain Name Request)
38 - zwrot nazwy domeny (ang. Domain Name Reply)
39 - SKIP Algorithm Discovery Protocol
40 - Photuris, Security Failures

Protokoły warstwy transportowej

W ramach stosu protokołów TCP/IP za transport danych odpowiadają protokoły TCP (ang. Transmission Control Protocol) oraz UDP (ang. User Datagram Protocol).
Głównym zadaniem protokołów tej warstwy jest dzielenie danych przychodzących z wyższych warstw modelu na segmenty. Segmenty te następnie przesyłane są do protokołów niższych warstw.
W warstwie transportowej istnieje mechanizm określania, do której aplikacji adresowane są przesyłane przy pomocy protokołu IP pakiety. Zarówno protokół TCP jak i UDP dysponują niezależnymi numerami, które określają numer portu.

UDP - User Datagram Protocol

Protokół UDP został zaprojektowany w celu dostarczania użytkownikowi mechanizmów do przesyłania datagramów pomiędzy programami użytkowymi. Nie nawiązuje on połączenia w trakcie wysyłania danych. Jest protokołem bezpołączeniowym.

Host wysyłający segment UDP nie uzyskuje informacji zwrotnej o tym, czy dane dotarły do adresata. W samym protokole UDP nie ma również mechanizmów pozwalających na kontrolę przepływu. W związku z tym w przypadku, gdy host, do którego mają dotrzeć segmenty nie jest w stanie ich obsłużyć, nie ma możliwości przesłać stosownej informacji. Jest więc protokołem zawodnym.
Do aplikacji wykorzystujących protokół UDP należą aplikacje komunikacji multimedialnej. Wszelkie programy wykorzystujące wideokonferencje, przesyłanie strumieniowe dźwięku wykorzystują szybkość tego protokołu. Innymi standardowymi protokołami sieciowymi korzystającymi z UDP są: DNS oraz TFTP.

Szybkość przesyłania segmentów UDP wynika z niewielkiej liczby pól (8 bajtów) w nagłówku oraz braku kontroli przepływu zaimplementowanej w tym protokole.

+	Bity 0 - 15	16 - 31
0	Port nadawcy	Port odbiorcy
32	Długość	Suma kontrolna
64	Dane	

Znaczenie poszczególnych pól:
port UDP nadawcy
port UDP odbiorcy
długość komunikatu UDP
suma kontrolna UDP - pole to jest opcjonalne i nie wykorzystywane w przypadku aplikacji pracujących w LAN (wtedy wartość tego pola wynosi 0). W przypadku, gdy suma kontrolna jest wyliczana jest to wykonywane podobnie jak w przypadku protokołu IP. Dane w tym przypadku są dzielone na 16 bitowe fragmenty, dla których wyliczane jest uzupełnienie do 1.

TCP - Transmission Control Protocol

Jest to protokół wiarygodny i połączeniowy. Zgodnie z założeniami protokół ten zapewnia wiarygodne przesyłanie danych pomiędzy dwoma hostami. Wykorzystywanych jest do tego kilka mechanizmów, takich jak: sumy kontrolne i numery sekwencyjne. Zagubione pakiety są ponownie retransmitowane.

Protokół TCP umożliwia kontrolę przeciążeń urządzeń znajdujących się pomiędzy komunikującymi się hostami. W przypadku, gdy następuje przeciążenie protokół TCP zmniejsza prędkość nadawania segmentów przez urządzenie nadawcze.

Segmenty protokołu TCP są enkapsulowane w pakiety IP i przesyłane przy użyciu tego protokołu. Ze względu na właściwości IP polegające na przesyłaniu w sposób skuteczny pakietów wszelkimi możliwymi drogami, segmenty mogą dotrzeć do adresata w różnej kolejności. Mechanizm porządkowania segmentów umożliwia nadawanie im numerów sekwencyjnych, które następnie ułatwiają ponowne złożenie danych.

Protokół TCP ma możliwość sterowania przepływem. Dzięki temu w sytuacji, gdy host docelowy lub łącze pozwala na szybszą transmisję następuje przesyłanie kilku segmentów w jednym pakiecie. W sytuacji odwrotnej, tzn. przy przeciążonym adresacie lub ograniczonej przepustowości łącza następuje zwolnienie transmisji poprzez przesyłanie mniejszej liczby segmentów lub pojedynczych segmentów.

Nagłówek protokołu zawiera 20 bajtów przeznaczonych na pola.

	Bity 0–3	4–7	8–15	16–31
0	Port nadawcy			Port odbiorcy
32	Numer sekwencyjny			
64	Numer potwierdzenia			
96	Długość nagłówka	Zarezerwowane	Flagi	Szerokość okna
128	Suma kontrolna			Wskaźnik priorytetu
160	Opcje (opcjonalnie)			
160/192+	Dane			

Znaczenie poszczególnych pól:
Numer portu źródłowego - 16 b - numer portu
Numer portu docelowego - 16 b - numer portu
Numer sekwencyjny pakietu - 32b - pole określające pozycję strumienia danych przesyłanych w strumieniu
Numer sekwencyjny potwierdzenia - 32 b - numer bajtu, który odbiorca spodziewa się otrzymać w następnej kolejności od nadawcy
Długość nagłówka - 4b - długość nagłówka wyrażona w 32-bitowych słowach
Pole zarezerwowane - 6b
Flagi - 6b - pole zawiera bity znacznikowe, które mają następujące znaczenie: URG - flaga oznaczająca, że dane zostały określone jako „pilne” przez warstwę wyższą ACK - oznacza, że wartość w polu numeru sekwencyjnego jest obowiązująca PSH - odbiorca powinien przesłać natychmiast dane do warstwy wyższej RST - połączenie w stanie zerowania SYN - nawiązanie połączenia i synchronizacja numerów początkowych FIN - zamknięcie połączenia, koniec strumienia danych u nadawcy
Szerokość okna - 16b - informacja, którą wysyła odbiorca o ilości danych, które może przyjąć od nadawcy.
Suma kontrolna - 16b - liczba umożliwiająca sprawdzenie, czy nie została naruszona integralność danych i nagłówka segmentu

Wskaźnik priorytetu - jeśli w polu flagi jest wpisana wartość URG, to pole to określa miejsce, w którym kończą się pilne dane.
Opcje (jeśli istnieją) - pole wykorzystywane do negocjacji wartości MSS pomiędzy klientem a serwerem, pole rzadko wykorzystywane
Wypełnienie - pole służące do wypełnienia długości nagłówka do pełnych 32 bitowych słów.

ARP i RARP

Jeżeli mamy ethernetową siec, w której się korzysta z TCP/IP, to mamy do czynienia z dwoma rodzajami adresów: 32-bitowy adres internetowy i 48-bitowy adres ethernetu. Mamy wiec do rozwiązania dwa problemy:

- Jeśli znamy adres internetowy drugiej stacji, to jak warstwa IP określi adres ethernetu? Taki problem nazywa się **problemem odwzorowania adresu** (*address resolution problem*).
- Kiedy mamy do czynienia z terminalami bezdyskowymi, to możemy określić adres ethernetu, ale nie znamy znowu adresu IP. Taki problem nazywa się **odwrotnym problemem odwzorowania adresu** (*reverse address resolution problem*).

Do rozwiązania pierwszego problemu służy protokół **ARP**, który pozwala stacji poprzez siec ethernet wysłać do wszystkich stacji specjalny pakiet, w którym prosi stacje o określonym numerze IP, by wysłała swój adres ethernetowy.

Protokol **RARP** dotyczy sieci z bezdyskowymi terminalami. Co najmniej jeden system w sieci jest serwerem RARP i zawiera 32-bitowy adres IP i odpowiadajacy mu 48-bitowy adres ethernet. Kazda stacja robocza moze w ten sposob otrzymac swoj adres IP.