

# Wykład #5

## Broadcasting & Multicasting

# Rozgłaszanie - Broadcasting

# Rozgłaszanie - Broadcasting

- Różne rodzaje adresowania
- Zastosowania rozgłaszania
- Adresacja
- Protokoły: UDP, IPv4, ~~IPv6, TCP, SCTP~~
- Zasady rozgłaszania dla IPv4
- Przykład

# Różne rodzaje adresowania

Rodzaj adresowania	IPv4?	IPv6?	TCP/SCTP?	UDP?	Liczba interfejsów określanych	Liczba używanych interfejsów
Jednostkowe - unicast	+	+	+	+	jeden	jeden
Swobodne - anycast	-	+	+/-	+	zbiór	jeden ze zbioru
Grupowe - multicast	opcja	+	-	+	zbiór	wszystkie ze zbioru
Rozgłaszanie - broadcast	+	-	-	+	wszystkie	wszystkie

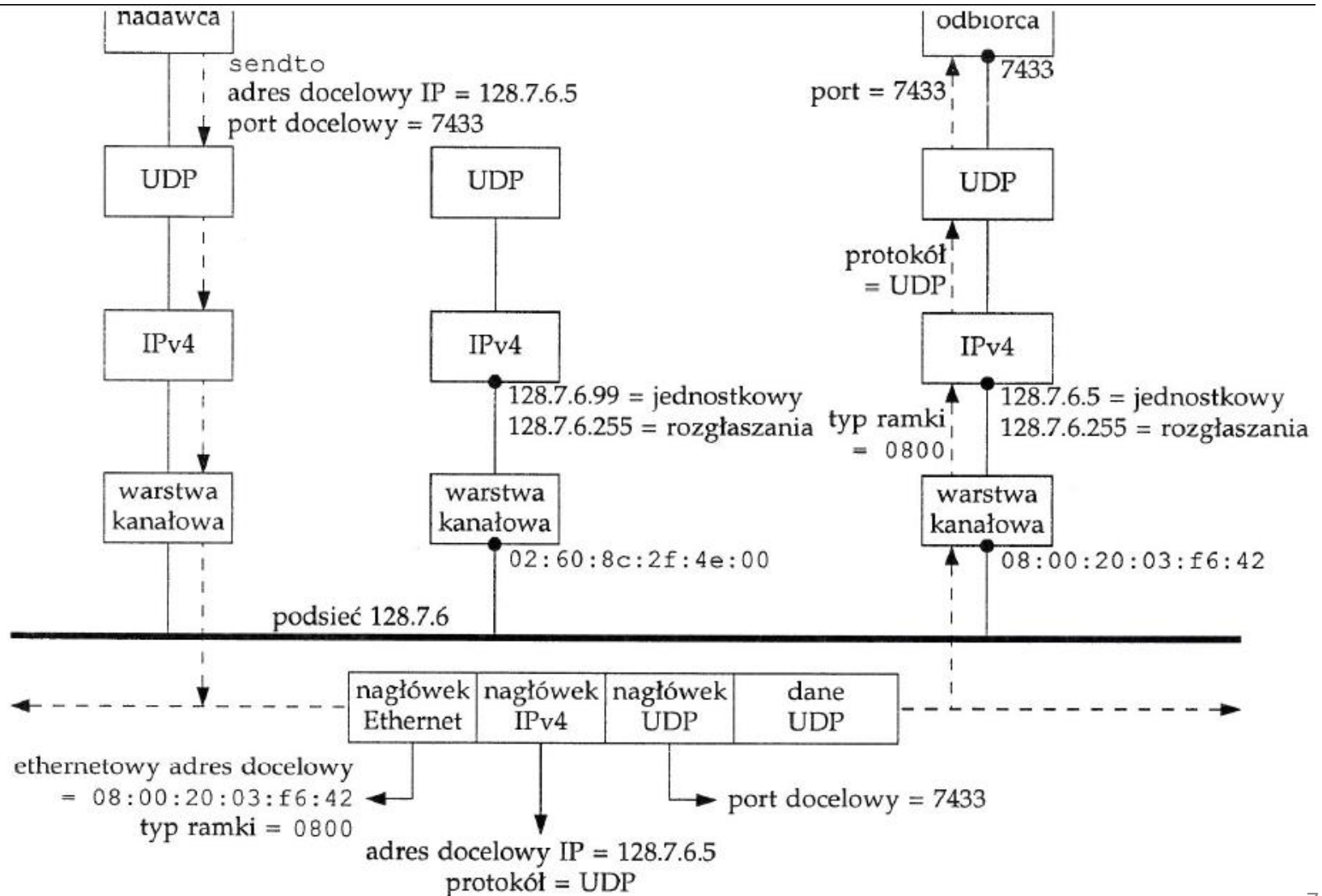
# Broadcasting - zastosowania

- Wyszukiwanie zasobów – wyszukiwanie połączenia/usługi bez znajomości adresu
- Zmniejszanie obciążenia w sieci?
- ARP (Address Resolution Protocol)
- BOOTP (Bootstrap Protocol)
- NTP (Network Time Protocol)
- Demony wyznaczania tras (routed/RIPv1)

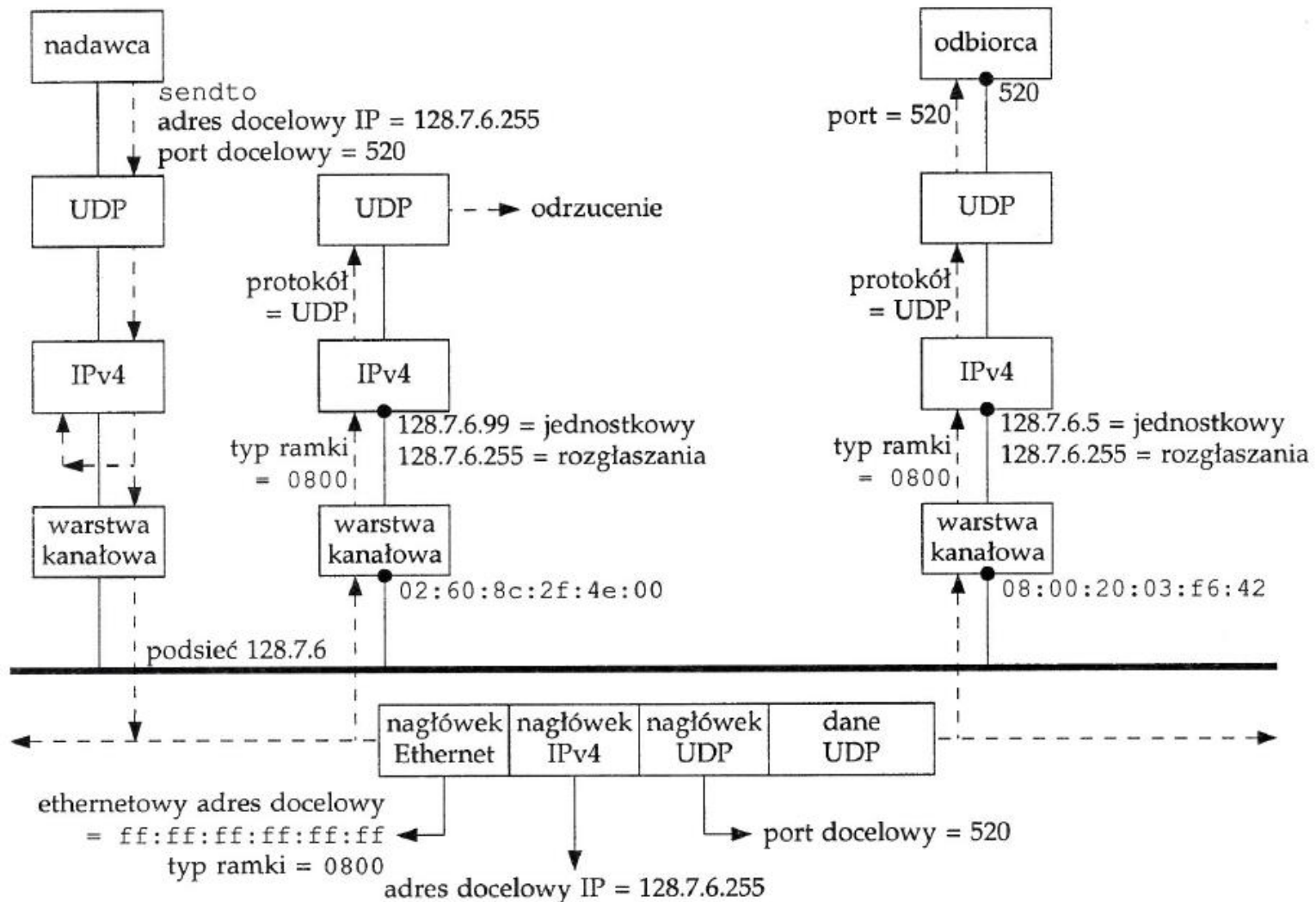
# Adresy rozgłaszania

- Adres rozgłaszania skierowany do podsieci:
  - Np. dla sieci o masce 24: 149.156.114.255
- Adres ograniczonego rozgłaszania:
  - 255.255.255.255
- Rutery nie powinny przekazywać datagramów rozgłoszeniowych

# UDP unicast



# Zasady rozgłaszania – UDP broadcasting





# Zasady rozgłaszania – UDP broadcasting

- **Datagram może być wysłany na adres rozgłoszeniowy jedynie gdy dla gniazda jest ustawiona opcja SO\_BROADCAST**
- Pakiety wysyłane na adres rozgłoszeniowy są także odbierane przez stację wysyłającą – rozgłaszany datagram ma być wysłany do wszystkich stacji w podsieci
- Nie wolno wysyłać pakietów ICMP o niedostępności usługi dla rozgłaszania

# Rozgłaszanie - przykład

- Klient – serwer usługi **daytime** – dwa przypadki:

1. Serwer rozsyła pakiety z określonym odstępem czasu – klient tylko nasłuchuje



2. Klient wysyła zapytanie na adres rozgłoszeniowy i czeka na odpowiedź (klient nie zna adresu serwera) – **service discovery**



# Rozgłaszanie grupowe

## Multicasting

# Rozsyłanie grupowe (MULTICAST)

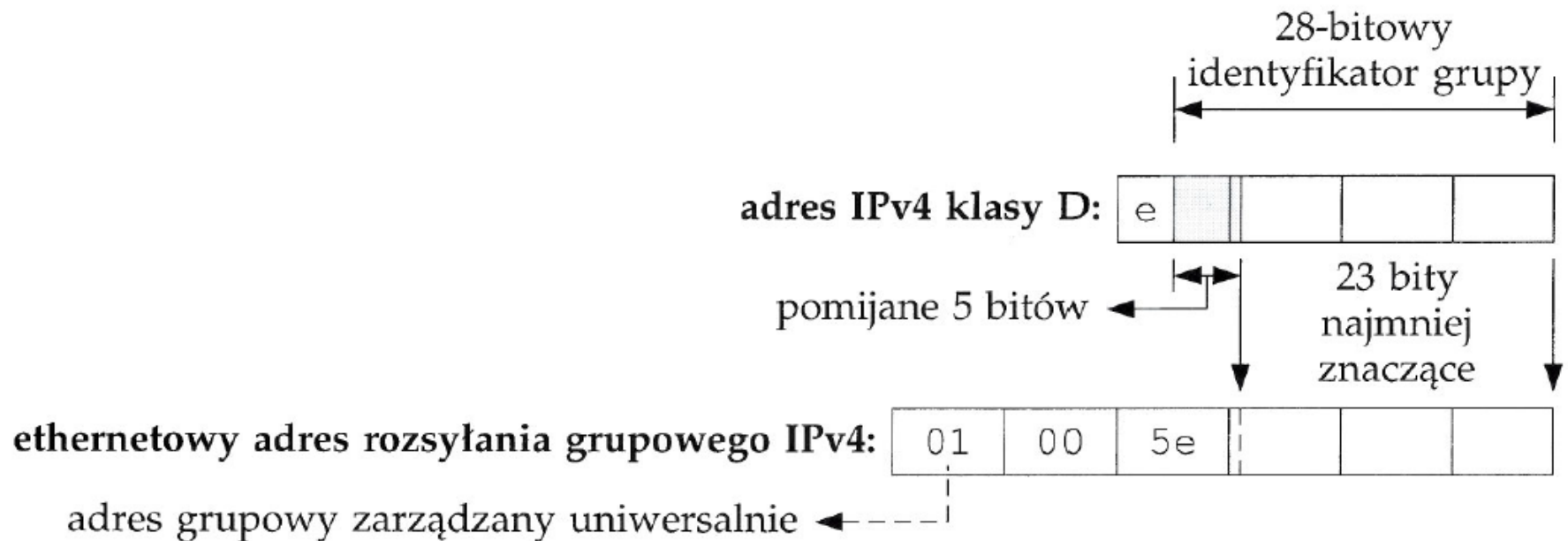
- Adresacja w warstwie sieciowej i łącza danych
- Rozsyłanie grupowe w sieciach lokalnych
- Rozsyłanie grupowe w sieciach rozległych
- Procedury wysyłania i odbierania pakietów dla adresacji multicast
- Opcje gniazd dla rozsyłania grupowego

# Adresacja rozsyłania grupowego – IPv4

- Adresy klasy D od 224.0.0.0 do 239.255.255.255
- Adresy 224.0.0.**0**/24 – zakres lokalny dla łącza – do wykrywania topologii sieci i zarządzania, nierutowalne
  - Np. 224.0.0.1 – grupa obejmująca wszystkie stacje
  - Np. 224.0.0.2 – grupa obejmująca wszystkie routery rozsyłania grupowego
- Odwzorowanie w ramki Ethernet na adres grupowy zarządzany globalnie

# Adresacja rozsyłania grupowego – IPv4

## – odwzorowanie na adresy Ethernetu



# Adresacja rozsyłania grupowego – IPv6

- Najbardziej znaczący bajt adresu ma wartość **FF**
- Dwa najbardziej znaczące bajty adresu Ethernet mają wartość **33:33** – adres grupowy zarządzany lokalnie
- Bity sygnalizatorów: **4 bity** (najmłodszy bit określa adres ogólnie znany lub tymczasowy)
- Bity zakresu: **4 bity**:
  - 1: lokalny dla węzła
  - 2: lokalny dla łącza
  - 5: lokalny dla siedziby
  - 8: lokalny dla organizacji
  - 14: globalny
- **32** najmniej znaczące bity kopiowane do ramki Ethernet
- **FF0X:0:0:0:0:0:114** – adres przeznaczony do eksperymentów

# Adresacja rozsyłania grupowego – format adresu IPv6

Bits	8	4	4	112
Field	Prefix FF	flags -RPT	scope	group ID

Scope – zasięg adresu:

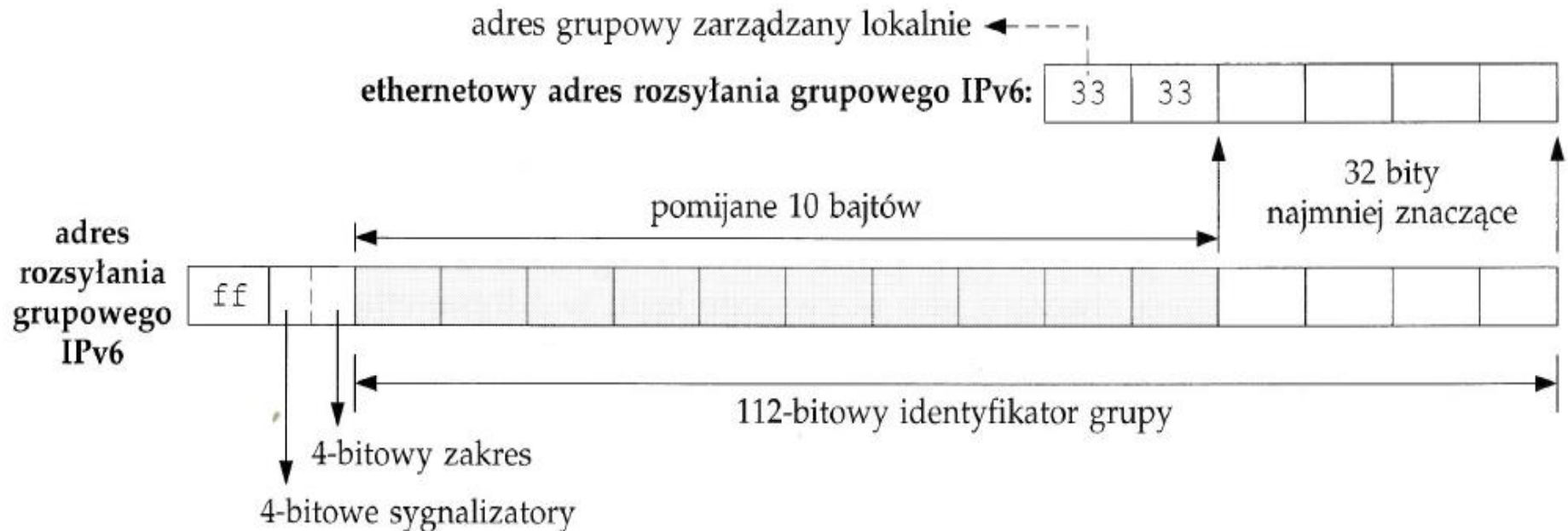
- 0x0 – zarezerwowane
- 0x1 – node local
- 0x2 – link local
- 0x5 – site local
- 0x8 – organisation local
- 0xE – global



# Flagi w adresie IPv6 typu multicast

Bit	Flag	0	1
0 (MSB)	Reserved	(Reserved)	(Reserved)
1	R (Rendezvous)	Rendezvous point not embedded	Rendezvous point embedded
2	P (Prefix)	Without prefix information	Address based on network prefix
3 (LSB)	T (Transient)	Well-known multicast address	Dynamically assigned multicast address

# Adresacja rozsyłania grupowego IPv6 – odwzorowanie na adresy Ethernet



# Adresacja rozsyłania grupowego – IPv6

## – przykłady zakresu

- FF01:0:0:0:0:0:0:101 means all NTP servers on the same node as the sender.
- FF02:0:0:0:0:0:0:101 means all NTP servers on the same link as the sender.
- FF05:0:0:0:0:0:0:101 means all NTP servers at the same site as the sender.
- FF0E:0:0:0:0:0:0:101 means all NTP servers in the internet.

# Adresacja rozsyłania grupowego – predefiniowane adresy IPv6, które nigdy nie powinny być używane dla grup

## Reserved Multicast Addresses:

1. FF00:0:0:0:0:0:0:0
2. FF01:0:0:0:0:0:0:0
3. FF02:0:0:0:0:0:0:0
4. FF03:0:0:0:0:0:0:0
5. FF04:0:0:0:0:0:0:0
- .....
12. FF0B:0:0:0:0:0:0:0
13. FF0C:0:0:0:0:0:0:0
14. FF0D:0:0:0:0:0:0:0
15. FF0E:0:0:0:0:0:0:0
16. FF0F:0:0:0:0:0:0:0

# Adresacja rozsyłania grupowego – predefiniowane adresy IPv6

- All Nodes Addresses:  
FF01:0:0:0:0:0:0:1  
FF02:0:0:0:0:0:0:1

The above multicast addresses identify the group of all IPv6 nodes, within scope 1 (node-local) or 2 (link-local).

- All Routers Addresses:  
FF01:0:0:0:0:0:0:2  
FF02:0:0:0:0:0:0:2  
FF05:0:0:0:0:0:0:2

The above multicast addresses identify the group of all IPv6 routers, within scope 1 (node-local), 2 (link-local), or 5 (site-local).

# Adresacja rozsyłania grupowego – predefiniowane adresy IPv6

- **Solicited-Node Address: FF02:0:0:0:0:1:FFXX:XXXX**
- The above multicast address is computed as a function of a node's unicast and anycast addresses. The solicited-node multicast address is formed by taking the low-order 24 bits of the address (unicast or anycast) and appending those bits to the prefix **FF02:0:0:0:0:1:FF00::/104** resulting in a multicast address in the range:
  - FF02:0:0:0:0:1:FF00:0000
  - To
  - FF02:0:0:0:0:1:FFFF:FFFF

Bits	8	4	4	88	24
Field	FF	0	2	0:0:0:0:1:FF	XX:XXXX

# Solicited-Node multicast address

- For example, the solicited node multicast address corresponding to the IPv6 address 4037::01:800:200E:8C6C is FF02::1:FF0E:8C6C. IPv6 addresses that differ only in the high-order bits, e.g. due to multiple high-order prefixes associated with different aggregations, will map to the same solicited-node address thereby reducing the number of multicast addresses a node must join.
- A **Solicited-Node multicast address** is an IPv6 multicast address valid within the local-link (e.g. an Ethernet segment or a Frame Relay cloud). Every IPv6 host will have at least one such address per interface. Solicited-Node multicast addresses are used in Neighbor Discovery Protocol for obtaining the layer 2 link-layer addresses of other nodes.
- A host is required to join a **Solicited-Node multicast group** for each of its configured unicast or anycast addresses.

# Adresacja rozsyłania grupowego IPv6 zbudowany w oparciu o prefiks adresu rozsyłania jednostkowego

<b>Bits</b>	8	4	4	8	8	64	32
<b>Field</b>	FF	flags	scope	reserved	plen	Network prefix	group ID

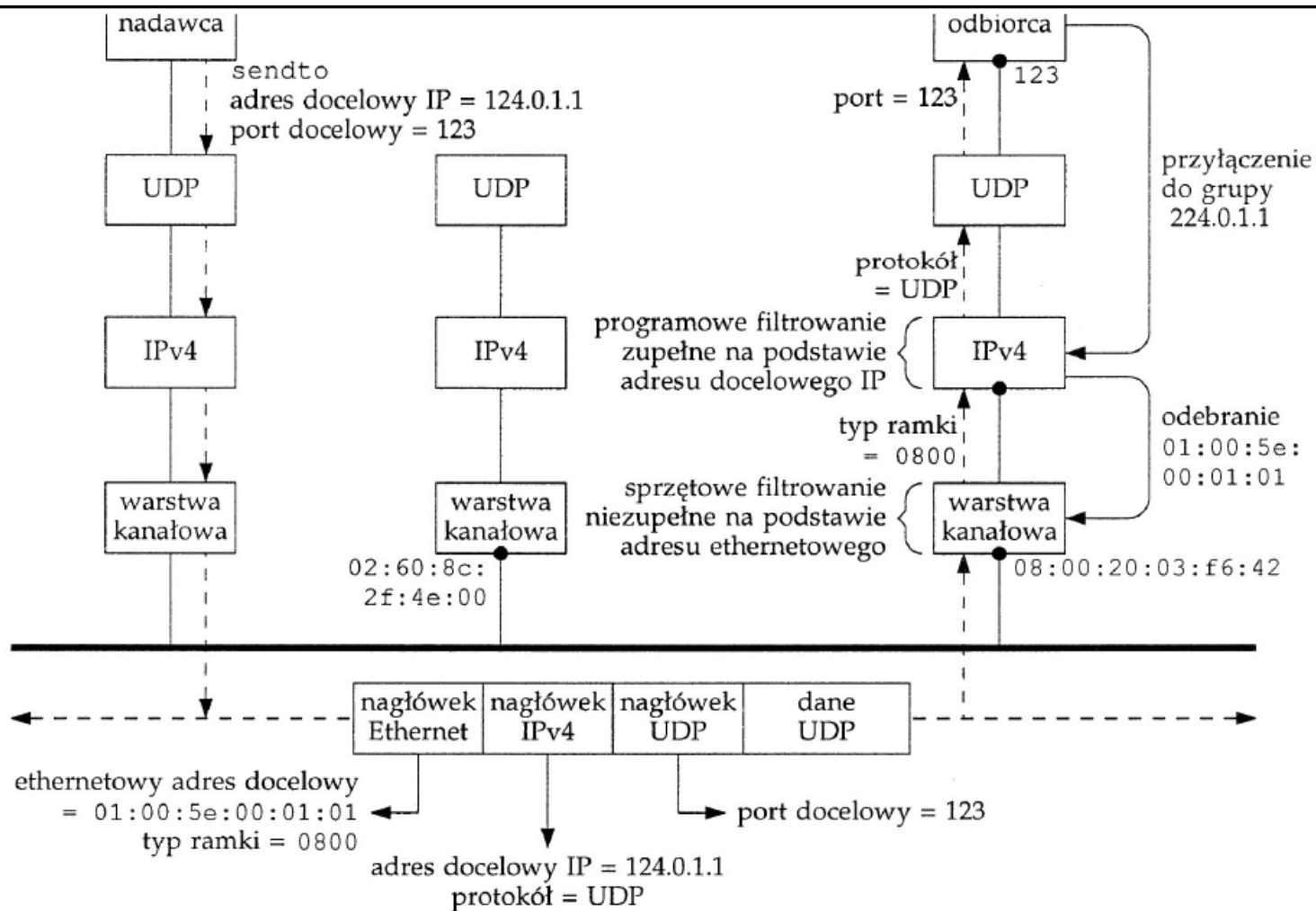
Flags : **P=1, T=1**



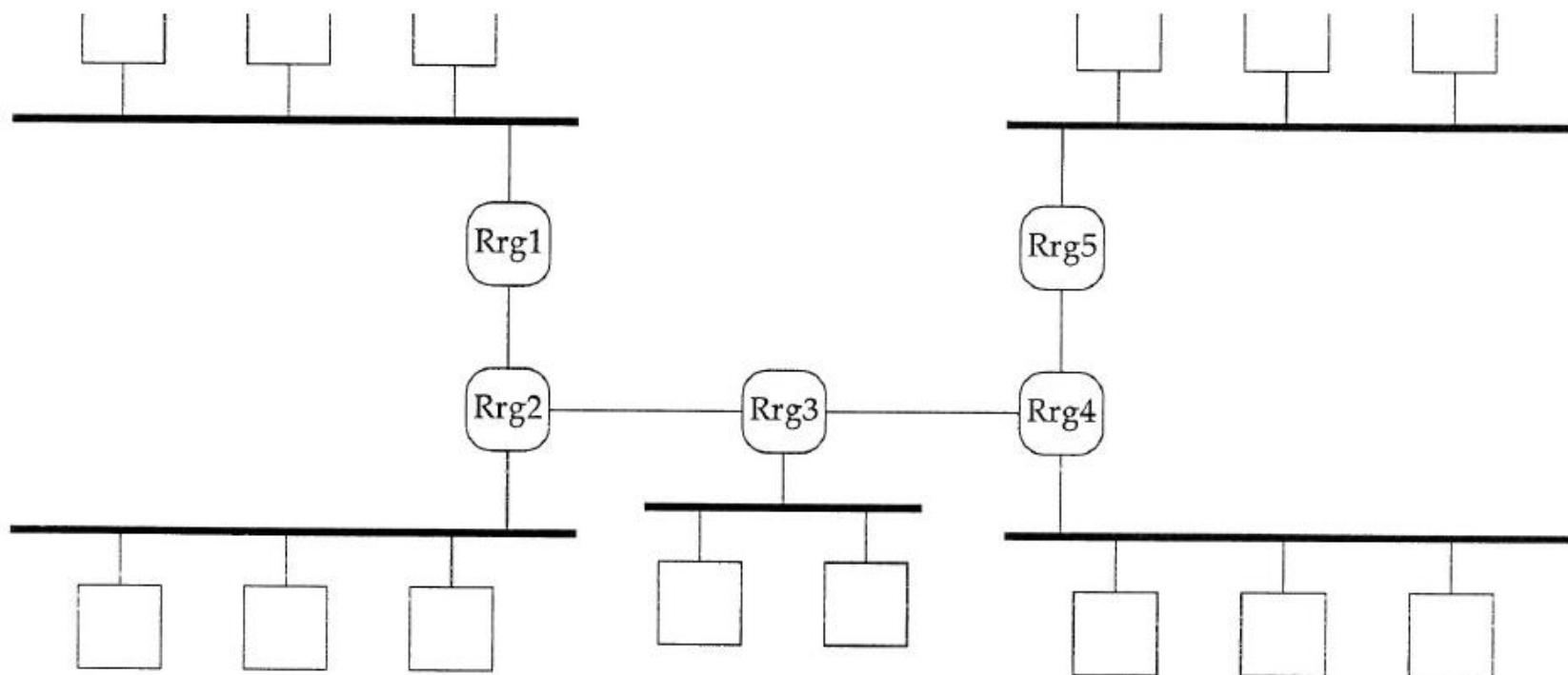
# Zakresy adresów rozsyłania grupowego IPv4 i IPv6

Zakres	Zakres IPv6	Dla protokołu IPv4	
		Zakres TTL	Zakres wyznaczany administracyjnie
Lokalny dla węzła	1	0	
Lokalny dla łącza	2	1	224.0.0.0 do 224.0.0.255
Lokalny dla siedziby	5	<32	239.255.0.0 do 239.255.255.255
Lokalny dla organizacji	8		239.192.0.0 do 239.195.255.255
globalny	14	<255	224.0.1.0 do 238.255.255.255

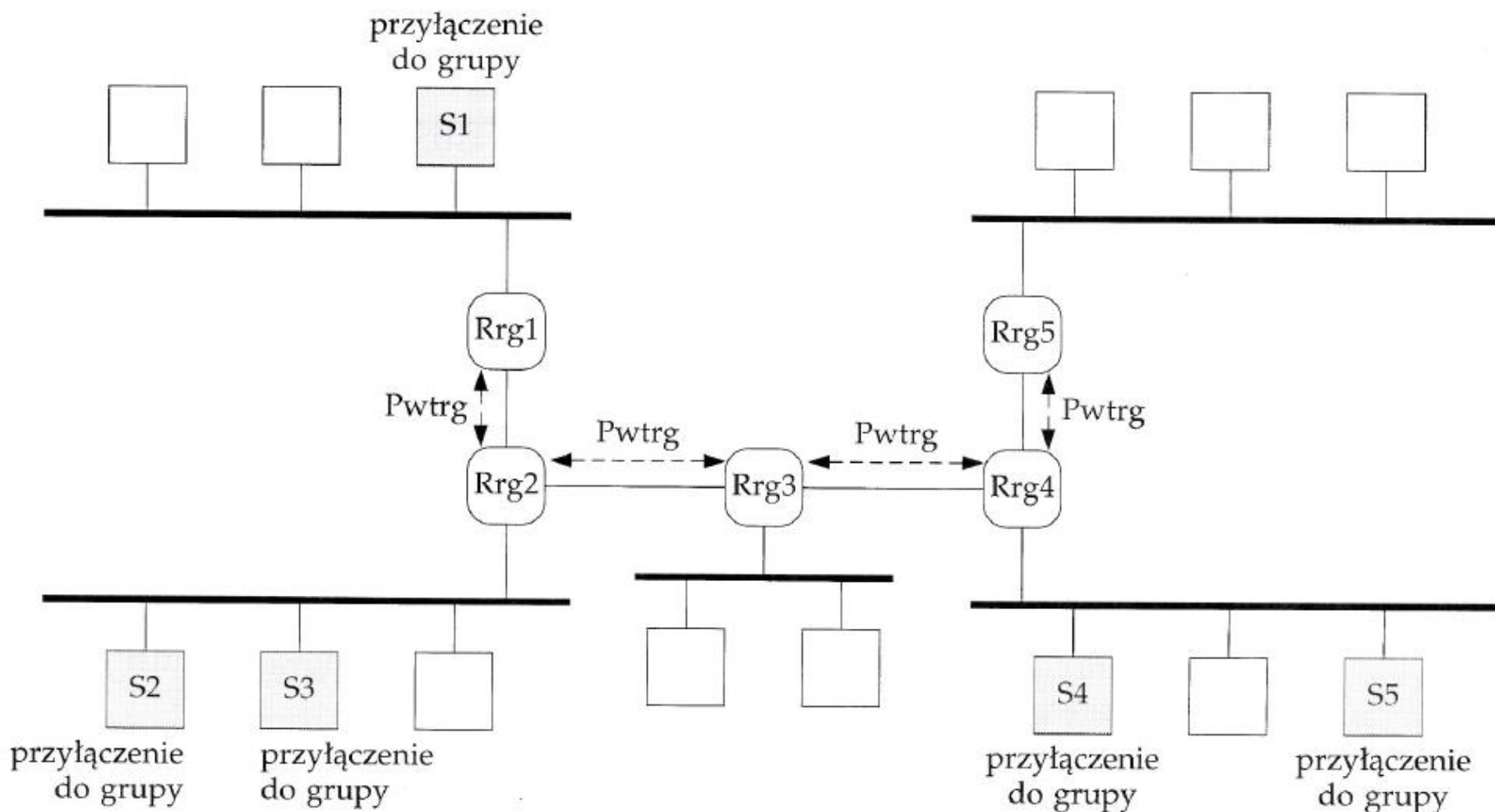
# Przykład datagramu UDP rozsyłanego na adres grupowy



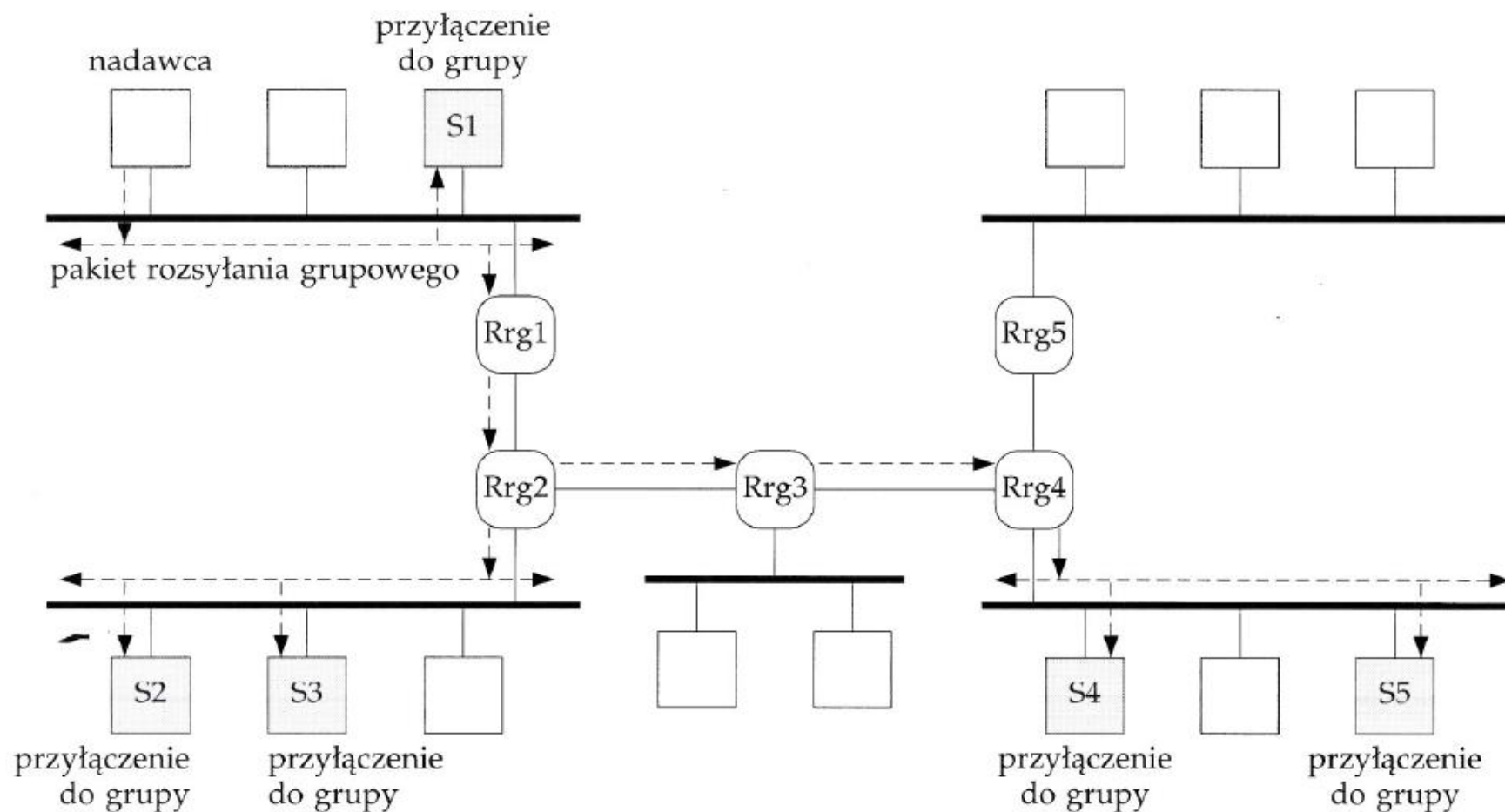
# Multicast w sieci globalnej



# Multicast w sieci globalnej



# Multicast w sieci globalnej



# Source specific multicast

- **Problem skalowalności multicastu** – jeden z ważnych powodów małego wykorzystania tego typu transmisji w sieci globalnej
- Brak mechanizmu/algorytmu przydziału adresów multicastowych
- Połączenie adresu nadawcy z adresem multicastowym rozwiązuje częściowo problem
- Rutery w sieci wiedzą gdzie jest nadawca, co znacznie upraszcza trasowanie

# Multicast – zasady komunikacji

- Stacja nadająca – wysyła pakiet na adres rozsyłania grupowego
  - Tworzy gniazdo UDP
  - Wysyła pakiet na adres typu multicast
- Stacja odbierająca – przyłącza się do sesji multicastowej:
  - Tworzy gniazdo UDP
  - Przyłącza się grupy multicastowej
  - Dowiązuje się gniazda na określonym porcie
  - Odbiera pakiety

# Multicast – wysyłanie

- Standardowa procedura:
  - Tworzymy gniazdo
  - Ustawiamy opcje (nieobowiązkowe):
    - `setsockopt()`
  - Wysyłamy datagram UDP na adres multicastowy

Polecenie	Typ danych	Znaczenie
IP_MULTICAST_IF	struct ip_addr	Domyślny interfejs wyjściowy
IP_MULTICAST_TTL	u_char	Określenie stałej TTL dla wyjścia
IP_MULTICAST_LOOP	u_char	Wysyłanie so samego siebie
IPV6_MULTICAST_IF	u_int	Domyślny interfejs wyjściowy
IPV6_MULTICAST_HOPS	int	Określenie stałej TTL dla wyjścia
IPV6_MULTICAST_LOOP	u_int	Wysyłanie so samego siebie



# Multicast – przyłączanie się do grupy multicastowej i odbieranie

- Tworzymy gniazdo
  - `socket(AF_INET, DGRAM,...)`
- Ustawiamy opcje (**obowiązkowe**)
  - `setsockopt(..., MCAST_JOIN_GROUP,...)`
- Nasłuchujemy i odbieramy datagramy
  - `bind(...)`
  - `recvfrom(...)`

# Multicast – opcje gniazd do odbioru datagramów

Polecenie	Typ danych	Opis
IP_ADD_MEMBERSHIP IP_DROP_MEMBERSHIP IP_BLOCK_SOURCE IP_UNBLOCK_SOURCE IP_ADD_SOURCE_MEMBERSHIP IP_DROP_SOURCE_MEMBERSHIP	struct ip_mreq struct ip_mreq struct ip_mreq_source struct ip_mreq_source struct ip_mreq_source struct ip_mreq_source	Dołączenie do grupy Odłączenie grupy Blokada źródła Odblokowanie źródła Dołączenie/odłączenie do/od grupy „źródłowej”
IPV6_JOIN_GROUP IPV6_LEAVE_GROUP	struct ipv6_mreq_source struct ipv6_mreq_source	Dołączenie do grupy Odłączenie grupy
<b>MCAST_JOIN_GROUP</b> <b>MCAST_LEAVE_GROUP</b> MCAST_BLOCK_SOURCE MCAST_UNBLOCK_SOURCE MCAST_JOIN_SOURCE_MEMBERSHIP MCAST_LEAVE_SOURCE_MEMBERSHIP	struct group_req struct group_req struct group_source_req struct group_source_req struct group_source_req struct group_source_req	Dołączenie do grupy Odłączenie grupy Blokada źródła Odblokowanie źródła Dołączenie/odłączenie do/od grupy „źródłowej”

# IP\_ADD\_MEMBERSHIP, IPV6\_JOIN\_GROUP/ IPV6\_ADD\_MEMBERSHIP, MCAST\_JOIN\_GROUP

- Subskrybcja do grupy multicastowej na wskazanym interfejsie sieciowym

```
struct ip_mreq {
    struct in_addr  imr_multiaddr; /* IPv4 class D multicast addr */
    struct in_addr  imr_interface; /* IPv4 addr of local interface */
};
struct ipv6_mreq {
    struct in6_addr  ipv6mr_multiaddr; /* IPv6 multicast addr */
    unsigned int     ipv6mr_interface; /* interface index, or 0 */
};
struct group_req {
    unsigned int     gr_interface; /* interface index, or 0 */
    struct sockaddr_storage gr_group; /* IPv4 or IPv6 multicast addr
*/
}
```

# IP\_ADD\_MEMBERSHIP, IPV6\_JOIN\_GROUP/ IPV6\_ADD\_MEMBERSHIP, MCAST\_JOIN\_GROUP

- Jeśli jako adres interfejsu podamy IN\_ADDR\_ANY (dla IPv4) lub indeks 0 (dla IPv6) to interfejs jest wybrany przez jądro systemu
- Węzeł należy do grupy multicastowej na danym interfejsie, gdy przynajmniej jeden proces należy do tej grupy na danym interfejsie
- Na jednym interfejsie można być zapisanym do wielu grup
- Dla IPv6, adresy różniące się tylko zasięgiem (scope bits) tworzą odrębne grupy multicastowe
- **IP\_MAX\_MEMBERSHIPS** – stała określająca maksymalną liczbę grup do których można się zapisać na pojedynczym interfejsie

# **IP\_DROP\_MEMBERSHIP, IPV6\_LEAVE\_GROUP/IPV6\_DROP\_MEMBERSHI, MCAST\_LEAVE\_GROUP**

- Wypisywanie się z grupy multicastowej
- Takie same struktury jak dla opcji „JOIN”
- Jeśli adresem będzie INADDR\_ANY lub wartość indeksu będzie równa 0, wtedy zostanie rozwiązana pierwsza pasująca subskrypcja
- Jeśli gniazdo jest zamykane to automatycznie następuje wypisanie z grup multicastowych

# IP\_BLOCK\_SOURCE, MCAST\_BLOCK\_SOURCE IP\_UNBLOCK\_SOURCE, MCAST\_UNBLOCK\_SOURCE

- Blokowanie/odblokowanie ruchu od podanego źródła

```
struct ip_mreq_source {
    struct in_addr imr_multiaddr; /* IPv4 class D multicast addr */
    struct in_addr imr_sourceaddr; /* IPv4 source addr */
    struct in_addr imr_interface; /* IPv4 addr of local interface */
};

struct group_source_req {
    unsigned int gsr_interface; /* interface index, or 0 */
    struct sockaddr_storage gsr_group; /* IPv4 or IPv6 multicast addr */
    struct sockaddr_storage gsr_source; /* IPv4 or IPv6 source addr */
}
```

# Source membership

- Opcje gniazd:
  - IP\_ADD\_SOURCE\_MEMBERSHIP (tylko IPv4),  
**MCAST\_JOIN\_SOURCE\_GROUP**
  - IP\_DROP\_SOURCE\_MEMBERSHIP (tylko IPv4),  
**MCAST\_LEAVE\_SOURCE\_GROUP**
- Struktury jak przy blokowaniu i odblokowaniu źródła

# Opcje gniazd dla rozgłaszania grupowego (1/2)

- **IP\_MULTICAST\_TTL, IPV6\_MULTICAST\_HOPS**
  - Set the IPv4 TTL or the IPv6 hop limit for outgoing multicast datagrams. If this is not specified, both will **default to 1**, which restricts the datagram to the local subnet.
- **IP\_MULTICAST\_LOOP, IPV6\_MULTICAST\_LOOP**
  - Enable or disable local loopback of multicast datagrams. By default, loopback is enabled: A copy of each multicast datagram sent by a process on the host will also be looped back and processed as a received datagram by that host, if the host belongs to that multicast group on the outgoing interface.



# Opcje gniazd dla rozgłaszania grupowego (2/2)

- **IP\_MULTICAST\_IF, IPV6\_MULTICAST\_IF**
  - Specify the interface for outgoing multicast datagrams sent on this socket. This interface is specified as either an `in_addr` structure for IPv4 or an interface index for IPv6. If the value specified is `INADDR_ANY` for IPv4 or is an interface index of 0 for IPv6, this removes any interface previously assigned by this socket option and the system will choose the interface each time a datagram is sent.

# Multicast - przykład serwera i klienta rozgłaszania grupowego

- Serwer i klient w jednym
- Wyszukiwanie sąsiadów
- Program tworzy dwa gniazda:
  - Do wysyłania
  - Do odbierania
- Nadaje i odbiera niezależnie w dwóch procesach



`multicast_example.c`