

# Programowanie sieciowe

---

Wykład dla studentów Informatyki Stosowanej  
i Fizyki Komputerowej  
UJ  
2014/2015

Michał Cieśla  
pok. D-2-47, email: [michal.ciesla@uj.edu.pl](mailto:michal.ciesla@uj.edu.pl)  
konsultacje: środy 10-12  
<http://users.uj.edu.pl/~ciesla/>



# Plan wykładu

---

1. Podstawowe informacje o sieciach TCP/IP.
2. Popularne protokoły i usługi sieciowe.
3. Programowanie sieciowe w oparciu o interfejs gniazd.
4. Elementy programowania rozproszonego.
5. Współczesne technologie w programowaniu rozproszonym.



# Literatura

---

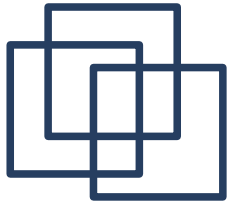
1. D.E. Comer, D. L. Stevens „Sieci komputerowe TCP/IP tom 3”, WNT 1997 (Internetworking with TCP/IP vol. III).
2. W. R. Stevens „UNIX programowanie usług sieciowych”, WNT 1999. (UNIX Network Programming).
3. R. Scrimger, P. LaSalle, M.Parihar, M. Gupta, C. Leitzke „Biblia TCP/IP”, Helion 2002 (TCP/IP Bible).
4. <http://www.ietf.org/rfc.html>
5. <http://www.google.pl>



# Wprowadzenie

---

1. Adres IP.
2. Model TCP/IP a ISO/OSI
  - protokół UDP
  - protokół TCP
2. Połączenie i rozłączenie w TCP
3. Programowanie z wykorzystaniem gniazd.



# Podstawy konfiguracji sieci

---

1. Adres IP (149.156.74.65)
2. Maska podsieci (255.255.255.0)
3. Brama (149.156.74.1)
4. Adresy serwerów DNS (149.156.74.21)



# Adres IP

---

Adres IP identyfikuje komputer w sieci. Jest on połączeniem adresu sieci z adresem hosta (komputera) wewnątrz tej sieci.

Przykład:

**149.156.74.65:**

**10010101.10011100.01001010.01000001**

**sieć: 149.156.0.0    host: 74.65**

adres rozgłoszeniowy:    **149.156.255.255**

zakres adresów hostów:    **0.1 - 255.254 (2<sup>16</sup> - 2 adresy)**

**192.168.1.133:**

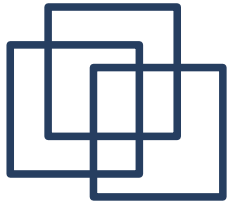
**11000000.10101000.00000001.10000101**

**sieć: 192.168.1.0    host: 133**

adres rozgłoszeniowy:    **192.168.1.255**

zakres adresów hostów:    **1 - 254 (2<sup>8</sup> - 2 adresy)**

---

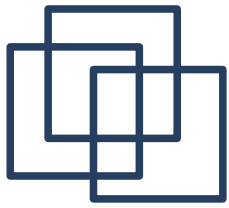


# Klasy adresów IP

Klasa	Adres	Zakres pierwszego oktetu	Liczba sieci	Liczba hostów
A	0NNNNNNN.HHHHHHHH.HHHHHHHH.HHHHHHHH	1 – 126	126	16777214
B	10NNNNNNN.NNNNNNNN.HHHHHHHH.HHHHHHHH	128 – 191	16384	65534
C	11NNNNNNN.NNNNNNNN.NNNNNNNN.HHHHHHHH	192 – 223	2097152	254
D	Adresowanie grupowe	224 – 239	-	-
E	Adresy eksperymentalne	240 – 254	-	-

## Adresy prywatne:

- 10.0.0.0 – 10.255.255.255
- 172.16.0.0 – 172.31.255.255
- 192.168.0.0 – 192.168.255.255



# Maska podsieci

---

Maska umożliwia uzyskanie adresu (pod)sieci na podstawie adresu IP hosta.

Dzięki temu możemy podzielić sieć na wiele podsieci:

adres sieci: **11000000.10101000.00000001.00010000 (192.168.1.16)**

maska: **11111111.11111111.11111111.11111000**

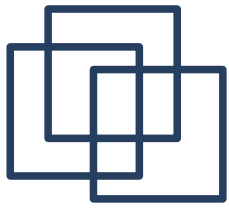
**(255.255.255.248)**

zakres adresów hostów w podsieci:

	<b>.00010001</b> (	<b>.17)</b>
	<b>.00010010</b> (	<b>.18)</b>
	<b>.00010011</b> (	<b>.19)</b>
	<b>.00010100</b> (	<b>.20)</b>
	<b>.00010101</b> (	<b>.21)</b>
	<b>.00010110</b> (	<b>.22)</b>
adres rozgłoszeniowy:	<b>.00010111</b> (	<b>.23)</b>

często używana notacja: **192.168.1.16/29** (bo 29 jedynek).





# Maska podsieci

---

Przykładowy podział sieci 192.168.1.0 na podsieci korzystając z maski:

192.168.1.0/29	adresy hostów: 1 - 6	adres rozgłoszeniowy: 7
192.168.1.8/29	adresy hostów: 9 - 14	adres rozgłoszeniowy: 15
192.168.1.16/28	adresy hostów: 17 - 30	adres rozgłoszeniowy: 31
192.168.1.32/27	adresy hostów: 33 - 62	adres rozgłoszeniowy: 63
192.168.1.64/26	adresy hostów: 65 - 126	adres rozgłoszeniowy: 127
192.168.1.128/25	adresy hostów: 129 - 254	adres rozgłoszeniowy: 255

w sumie  $256 - 2 \cdot 6 = 244$  różne adresy hostów.



# Przykłady

---

Przykłady konfiguracji dla hosta o adresie IP **149.156.74.65**

adres IP:           **149.156.74.65**  
maska:             **255.255.0.0**  
adres podsieci:       **149.156.0.0**  
zakres:             **.0.1 - .255.254**

adres IP:           **149.156.74.65**  
maska:             **255.255.255.0**  
adres podsieci:       **149.156.74.0**  
zakres:             **.1 - .254**

adres IP:           **149.156.74.65**  
maska:             **255.255.255.128**  
adres podsieci:       **149.156.74.0**  
zakres:             **.1 - .126**

adres IP:           **149.156.74.65**  
maska:             **255.255.255.252**  
adres podsieci:       **149.156.74.64**  
zakres:             **.65 - .66**



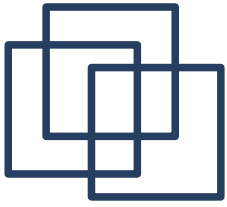
# Brama domyślna

---

Brama domyślna pełni funkcję routera - kieruje ruchem pomiędzy siecią lokalną (LAN) i siecią rozległą (WAN). Brama sieciowa posiada zwykle dwa adresy sieciowe: jeden dla komputerów z sieci lokalnej (LAN), drugi dla komputerów z poza sieci lokalnej (WAN).

Wysyłając informacje nadawca sprawdza, czy odbiorca znajduje się w tej samej sieci lokalnej. Jeśli tak, pakiet jest wysyłany bezpośrednio do odbiorcy. Jeśli nie informacja ma zostać dostarczona poza sieć lokalną jest ona wysyłana na adres bramy domyślnej.

Jeśli odbiorca znajduje się w tej samej sieci do której należy adres WAN routera, brama przesyła do niego informacje. Jeśli nie brama przesyła informacje do kolejnego routera w sieci WAN.



# Serwery DNS

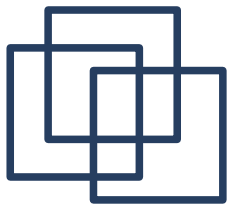
---

Serwery DNS (*Domain Name Server*) tłumaczą nazwy hostów na adres IP, np. **www.if.uj.edu.pl = 149.156.89.131**, umożliwiając używanie nazw w miejsce adresów IP.

1. Aby przetłumaczyć adres **www.if.uj.edu.pl** należy zapytać serwera nazw dla domeny **if.uj.edu.pl**,
2. Jeśli adres serwera nazw dla domeny **if.uj.edu.pl** jest nieznany należy odnaleźć serwer nazw dla domeny **uj.edu.pl**. itd.
3. Adresy wszystkich serwerów nazw w domenie **pl** są przechowywane przez serwery NASK'u (Naukowa Akademicka Sieć Komputerowa).

Serwery DNS zapamiętują na jakiś czas (typowo 24 godz.) otrzymane odpowiedzi co przyspiesza proces odszukania adresu IP.

---



# Narzędzie nslookup

```
C:\WINDOWS\system32\cmd.exe - nslookup

C:\>nslookup
Serwer domyślny:  th.if.uj.edu.pl
Address:  149.156.74.9

> interia.pl
Serwer:  th.if.uj.edu.pl
Address:  149.156.74.9

Nieautorytatywna odpowiedź«:
Nazwa:  interia.pl
Address:  217.74.65.119

> set q=mx
> interia.pl
Serwer:  th.if.uj.edu.pl
Address:  149.156.74.9

Nieautorytatywna odpowiedź«:
interia.pl      MX preference = 5, mail exchanger = mx.poczta.interia.pl

interia.pl      nameserver = dns.poczta.fm
interia.pl      nameserver = ns.interia.pl
mx.poczta.interia.pl  internet address = 80.48.65.10
dns.poczta.fm   internet address = 195.150.20.131
ns.interia.pl   internet address = 217.74.65.131
> _
```



# Usługa DHCP

---

Usługa DHCP (*Dynamic Host Configuration Protocol*) [RFC 2131] umożliwia automatyczną konfigurację (dzierżawę) parametrów sieciowych hostów.

Komunikacja pomiędzy hostem a serwerem DHCP przebiega następująco:

[host] DHCPDISCOVER (komunikat wysyłany do wszystkich)

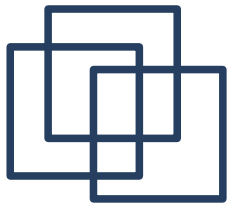
[server] DHCPOFFER (zawiera proponowane parametry konfiguracji, wysyłany na adres karty sieciowej hosta)

[host] DHCPREQUEST (prośba o konfigurację lub przedłużenie jej ważności)

[server] DHCPACK (przesłanie ostatecznych parametrów)

...

[host] DHCPRELEASE (zwolnienie konfiguracji)



# Testowanie konfiguracji

Start / uruchom / cmd

```
ipconfig [/all]
```

```
C:\WINDOWS\system32\cmd.exe

C:\>ipconfig

Konfiguracja IP systemu Windows

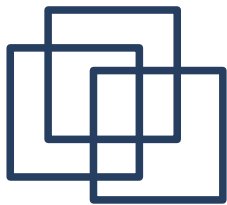
Karta Ethernet Połączenie sieci bezprzewodowej:

    Stan nośnika . . . . . : Nośnik odłączony

Karta Ethernet Połączenie lokalne:

    Sufiks DNS konkretnego połączenia :
    Adres IP autokonfiguracji . . . . : 149.156.74.158
    Maska podsieci. . . . . : 255.255.255.0
    Brama domyślna. . . . . : 149.156.74.1

C:\>
```



# Testowanie konfiguracji

Start / uruchom / cmd

ping

```
C:\WINDOWS\system32\cmd.exe

C:\>ping 149.156.76.1

Badanie 149.156.76.1 z użyciem 32 bajtów danych:

Odpowiedź z 149.156.76.1: bajtów=32 czas=1ms TTL=254
Odpowiedź z 149.156.76.1: bajtów=32 czas=1ms TTL=254
Odpowiedź z 149.156.76.1: bajtów=32 czas=1ms TTL=254
Odpowiedź z 149.156.76.1: bajtów=32 czas=1ms TTL=254

Statystyka badania ping dla 149.156.76.1:
    Pakiety: Wysłane = 4, Odebrane = 4, Utracone = 0 (0% straty),
Szacunkowy czas błędzenia pakietów w milisekundach:
    Minimum = 1 ms, Maksimum = 1 ms, Czas średni = 1 ms

C:\>ping www.google.pl

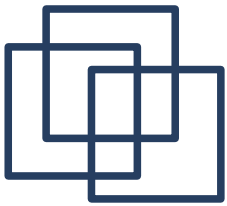
Badanie www.google.akadns.net [66.102.9.99] z użyciem 32 bajtów danych:

Odpowiedź z 66.102.9.99: bajtów=32 czas=62ms TTL=239
Odpowiedź z 66.102.9.99: bajtów=32 czas=62ms TTL=239
Odpowiedź z 66.102.9.99: bajtów=32 czas=63ms TTL=239
Odpowiedź z 66.102.9.99: bajtów=32 czas=64ms TTL=239

Statystyka badania ping dla 66.102.9.99:
    Pakiety: Wysłane = 4, Odebrane = 4, Utracone = 0 (0% straty),
Szacunkowy czas błędzenia pakietów w milisekundach:
    Minimum = 62 ms, Maksimum = 64 ms, Czas średni = 62 ms

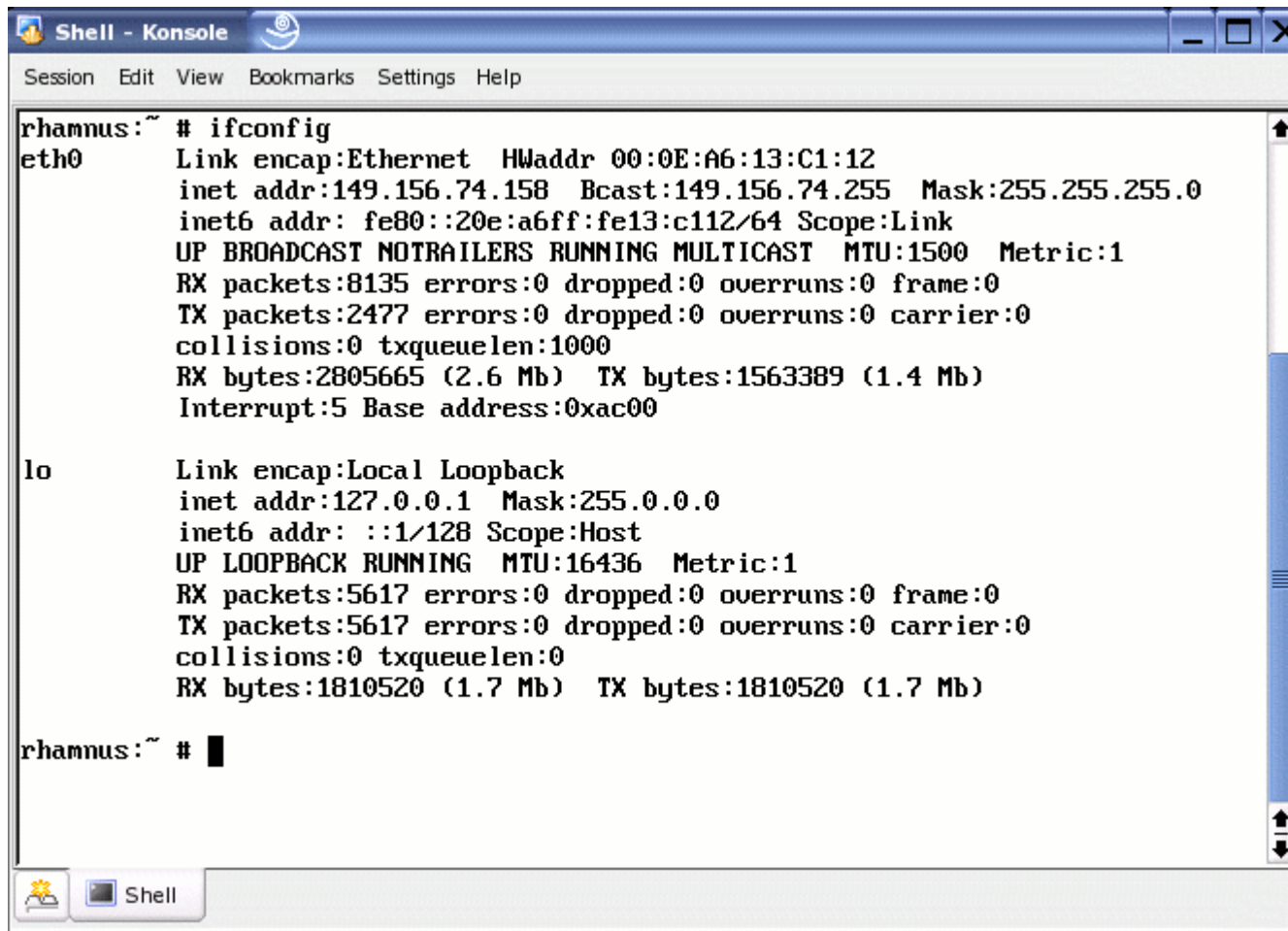
C:\>_
```





# Konfiguracja sieci Linux

`ifconfig ( /sbin/ifconfig )`



```
Shell - Konsole
Session Edit View Bookmarks Settings Help

rhammus:~ # ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0E:A6:13:C1:12
          inet addr:149.156.74.158  Bcast:149.156.74.255  Mask:255.255.255.0
          inet6 addr: fe80::20e:a6ff:fe13:c112/64  Scope:Link
          UP BROADCAST NOTRAILERS RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:8135 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2477 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2805665 (2.6 Mb)  TX bytes:1563389 (1.4 Mb)
          Interrupt:5 Base address:0xac00

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:5617 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5617 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1810520 (1.7 Mb)  TX bytes:1810520 (1.7 Mb)

rhammus:~ # █
```



# Konfiguracja sieci Linux

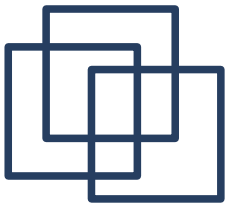
---

Przykład:

```
ifconfig eth0 hw ether 000e0e0e0e0e netmask 255.255.255.0  
192.168.0.2
```

skonfiguruje interfejs sieciowy **eth0** do pracy w standardzie **ethernet** z adresem sprzętowym **00:0E:0E:0E:0E:0E**.

Do interfejsu zostanie przypisany adres IP: **192.168.0.2** i maska sieciowa **255.255.255.0**.



# Konfiguracja sieci Linux

`route ( /sbin/route )`

```
Shell - Konsole
Session Edit View Bookmarks Settings Help

rhamnus:~ # route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
149.156.74.0     0.0.0.0         255.255.255.0   U        0      0      0 eth0
169.254.0.0     0.0.0.0         255.255.0.0    U        0      0      0 eth0
127.0.0.0       0.0.0.0         255.0.0.0      U        0      0      0 lo
0.0.0.0         149.156.74.1   0.0.0.0        UG       0      0      0 eth0
rhamnus:~ #
```



# Konfiguracja sieci Linux

---

Przykład:

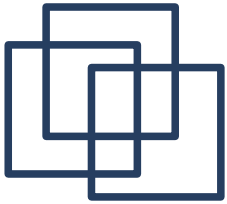
```
route add default gw 192.168.0.1
```

dodaje (**add**) domyślną (**default**) bramę (**gw**) o adresie IP **192.168.0.1**.

Aby zmieniać konfigurację sieciową za pomocą poleceń **ifconfig** i **route** zwykle należy mieć prawa root'a.

Informacje o serwerach DNS są zapisane w pliku **/etc/resolv.conf**:

```
domain if.uj.edu.pl  
nameserver 8.8.8.8
```



# netstat

```
m02 ~ # netstat
```

```
Active Internet connections (w/o servers)
```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	268	m02.gdziestam.pl:ssh	ja.uj.edu.pl:1325	ESTABLISHED
tcp	0	0	m02.gdziestam.pl:ssh	inny.jakis.pl:45808	ESTABLISHED

```
...
```

```
m02 ~ # netstat -l
```

```
Active Internet connections (only servers)
```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	*:imaps	*:*	LISTEN
tcp	0	0	*:pop3s	*:*	LISTEN
tcp	0	0	localhost:mysql	*:*	LISTEN
tcp	0	0	*:pop3	*:*	LISTEN
tcp	0	0	*:imap	*:*	LISTEN
tcp	0	0	*:http	*:*	LISTEN
tcp	0	0	*:ssh	*:*	LISTEN
tcp	0	0	*:smtp	*:*	LISTEN
udp	0	0	*:1033	*:*	
udp	0	0	*:1036	*:*	
udp	0	0	*:1037	*:*	
udp	0	0	*:2845	*:*	
udp	0	0	*:snmp	*:*	

```
...
```



# Model TCP/IP a ISO/OSI

Model OSI

warstwa aplikacji
warstwa prezentacji
warstwa sesji
warstwa transportowa
warstwa sieciowa
warstwa łącza danych
warstwa fizyczna

Model TCP/IP

warstwa aplikacji (zastosowań)	
TCP	UDP
IPv4, IPv6	
sterownik programowy i sprzęt	



# Protokół UDP

---

**UDP (User Datagram Protocol).** [RFC 768]

Protokół obsługi bezpołączeniowej procesów użytkownika. Nie daje gwarancji, że pakiety (datagramy) dotrą do wyznaczonego celu. Może korzystać zarówno z IPv4 oraz IPv6.



# Protokół TCP

---

## **TCP (Transmission Control Protocol).** [RFC 793]

- wykorzystuje połączenie między klientem i serwerem,
- zapewnia niezawodność przesyłu danych - potwierdzenia, szacowanie czasu powrotu (round-trip time),
- ustala kolejność danych - numeracja segmentów,
- steruje przepływem - nadawca nie może spowodować przepełnienia bufora odbiorcy,
- zapewnia połączenie, które jest w pełni dwukierunkowe (full-duplex).
- jest stosowany w większości internetowych programów użytkowych. Może korzystać zarówno z IPv4 oraz IPv6.

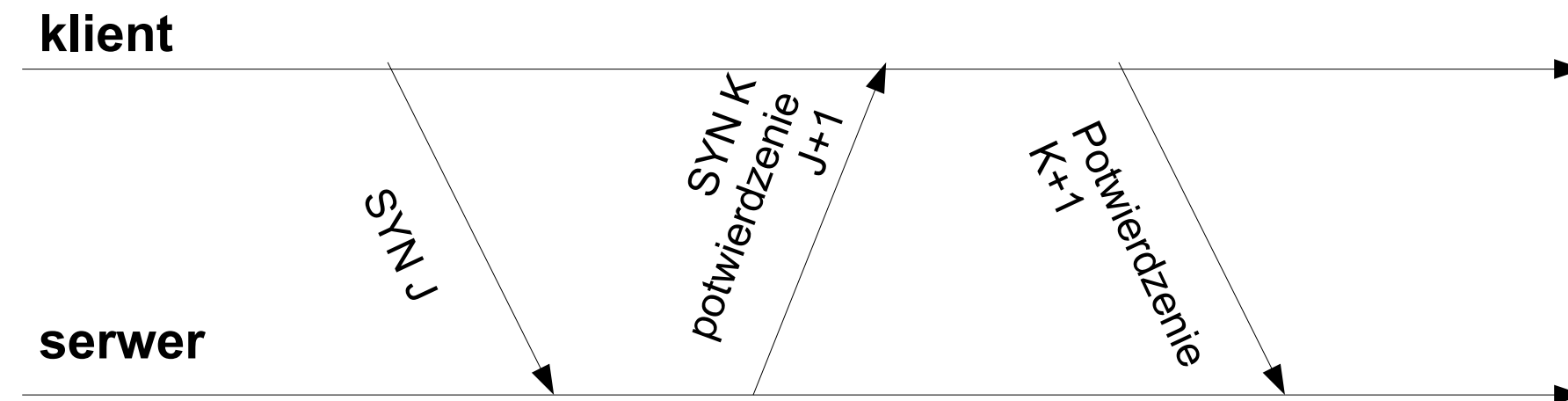




# Ustanowienie połączenia TCP

Uzgadnianie trójfazowe (three-way handshake).

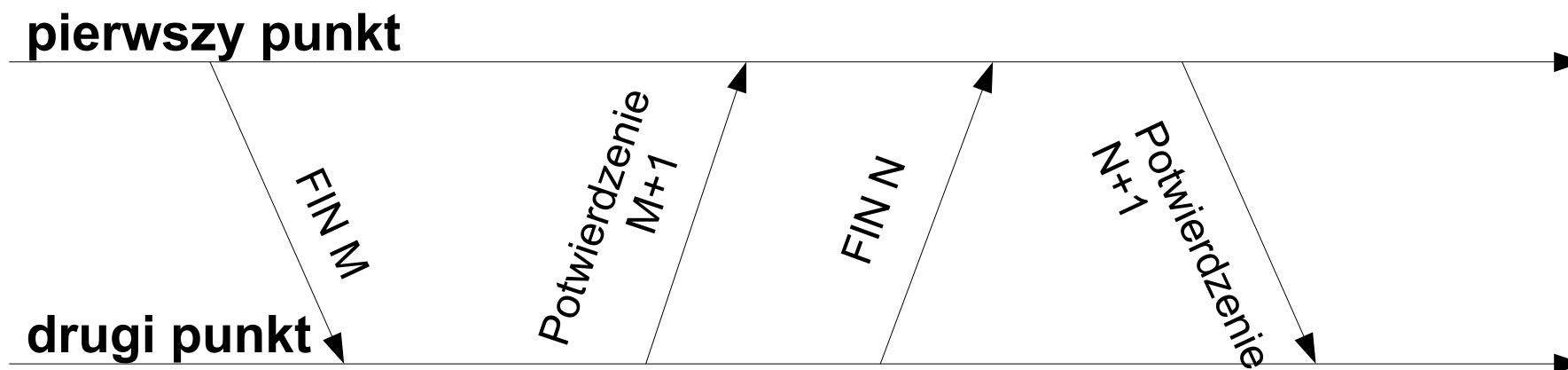
1. Serwer przygotowuje się na przyjęcie połączenia - otwarcie bierne.
2. Klient rezerwuje gniazdo za pomocą `listen` a następnie rozpoczyna otwarcie aktywne - TCP wysyła segment danych SYN (synchronize).
3. Serwer wysyła własny segment SYN zawierający potwierdzenie ACK (acknowledgment).
4. Klient potwierdza przyjęcie segmentu SYN od serwera.





# Zakończenie połączenia TCP

1. Jeden punkt końcowy inicjuje zakończenie połączenia - zamknięcie aktywne. TCP wysyła segment FIN.
2. Drugi punkt wykonuje zamknięcie bierne - TCP potwierdza odbiór FIN.
3. Po pewnym czasie drugi punkt zamyka swoje połączenie - TCP wysyła segment FIN.
4. Pierwszy punkt potwierdza odbiór segmenty FIN.





# Numery portów

---

Numerami portów zarządza IANA (Internet Assigned Numbers Authority). Wyróżniamy trzy grupy portów.

porty ogólne (well-known), numery z zakresu 0 - 1023, np.: HTTP 80, FTP 21, SMTP 25.

porty zarejestrowane (registered), numery 1024 - 49151. np.: X-  
Windows 6000-6063, MySQL 3036.

porty dynamiczne 49152 - 65535, nie posiadają przyporządkowania do usług.



# Podsumowanie

---

Konfiguracja komputera w sieci TCP/IP zawiera cztery parametry: adres IP, maskę podsieci, adres bramy domyślnej i adresy serwerów DNS.

W modelu TCP/IP dostępne są dwa protokoły warstwy transportowej: TCP i UDP. Ruch kierowany jest za pomocą protokołu IP (warstwa sieciowa).