

# Laboratorium 10

## Programowanie sieciowe 1

Maryna Lukachyk(308294)

### Pytania:

Co identyfikuje adres IP a co port?

Adres IP wyznacza węzeł w sieci, numer portu określa proces w węźle.

Czym są i do czego służą gniazda?

- podstawowy mechanizm służący do wymiany danych między maszynami (przy użyciu sieci)
- są używane w czwartej warstwie sieciowego modelu OSI/ISO
- służą do komunikacji między procesami, w tym przez sieć
- kombinacja IP i nr portu

Czym różni się deskryptor gniazda od deskryptora pliku?

Unix udostępnia dwa typy deskryptorów - deskryptory plików, komunikacji międzyprocesowej oraz deskryptory gniazd.

Deskryptory gniazd, tak samo jak inne deskryptory udostępniają operacje IO. Operacje zapisu i odczytu są bardzo uogólnione w systemach uniksowych, deskryptor opisuje obiekt, a operacje IO opisują stosowane do niego metody, dlatego w przypadku deskryptorów gniazd, gdy zapisujemy do niego, oznacza to, że wysyłamy nim dane do innego urządzenia.

Jeśli program dostanie deskryptor odpowiadający plikowi – do pliku. Jeśli program otrzyma deskryptor odpowiadający gniazdu, wyśle dane przez internet do odległego komputera.

Który argument funkcji służy do określenia typu gniazda?

Drugi parametr w wywołaniu funkcji gniazda określa typ gniazda.

```
int socket(int domain, int type, int protocol)
```

Jakie wartości przyjmuje ten argument dla gniazd połączeniowych a jakie dla bezpołączeniowych?

int type określa tryb komunikacji:

SOCK\_STREAM dla gniazd TCP // połączeniowy

SOCK\_DGRAM dla UDP //beipołączeniowy

Jaki jest zakres liczbowy portów dostępnych do wykorzystania dla użytkownika niebędącego administratorem?

1024 – 49 151 – porty użytkownika (registered)

### Reprezentacja sieciowa

Co oznaczają pojęcia:

1)Big-Endian?

to forma zapisu danych, w której najbardziej znaczący bajt (zwany też górnym bajtem, z ang. high-order byte) umieszczony jest jako pierwszy.

2)Little-Endian?

to forma zapisu danych, w której najmniej znaczący bajt (zwany też dolnym bajtem, z ang. low-order byte) umieszczony jest jako pierwszy.

3)Network Byte Order?

Aby umożliwić maszynom o różnych konwencjach kolejności bajtów komunikowanie się ze sobą, protokoły internetowe określają kanoniczną konwencję kolejności bajtów dla danych przesyłanych przez sieć.

Do czego służą funkcje: htonl, htons, ntohl, ntohs? Co oznaczają ich nazwy (od czego są to skróty)?

Function	Description
htons()	Host to Network Short
htonl()	Host to Network Long
ntohl()	Network to Host Long
ntohs()	Network to Host Short

Listed below are some more detail about these functions –

- **unsigned short htons(unsigned short hostshort)** – This function converts 16-bit (2-byte) quantities from host byte order to network byte order.
- **unsigned long htonl(unsigned long hostlong)** – This function converts 32-bit (4-byte) quantities from host byte order to network byte order.
- **unsigned short ntohs(unsigned short netshort)** – This function converts 16-bit (2-byte) quantities from network byte order to host byte order.
- **unsigned long ntohl(unsigned long netlong)** – This function converts 32-bit quantities from network byte order to host byte order.

Liczbę w postaci szesnastkowej: cafe zapisano na dwóch bajtach w postaci: feca. Jaka to reprezentacja?

## Little-Endian

Obsługa adresów IPv4 oraz IPv6

IPv4: 192.168.1.1

IPv6: 2001:0db8:0000:0000:0000:0000:1428:57ab (postać 16-stkowa) (8 grup po 2 bajty)

IPv4 and IPv6 share the local port space. When you get an IPv4 connection or packet to a IPv6 socket, its source address will be mapped to v6 and it will be mapped to v6.

#### Address format

```
struct sockaddr_in6 {
    sa_family_t    sin6_family;   /* AF_INET6 */
    in_port_t      sin6_port;     /* port number */
    uint32_t       sin6_flowinfo; /* IPv6 flow information */
    struct in6_addr sin6_addr;     /* IPv6 address */
    uint32_t       sin6_scope_id; /* Scope ID (new in 2.4) */
};

struct in6_addr {
    unsigned char  s6_addr[16];   /* IPv6 address */
};
```

Jakie są typowe numery portów dla http, ftp, ssh itp.?

53	DNS
20	FTP – przesyłanie oraz pobieranie plików i folderów
21	FTP – przesyłanie poleceń
67	DHCP – serwer
68	DHCP – klient
79	Finger
70	Gopher
80	HTTP, dodatkowe serwery, np. proxy, są najczęściej umieszczane na porcie 8080
443	HTTPS (HTTP na SSL)
143	IMAP
220	IMAP3

Do czego służą porty?

Do identyfikowania procesów działających na odległych systemach.

Czym się różni komunikacja połączeniowa od bezpołączeniowej?

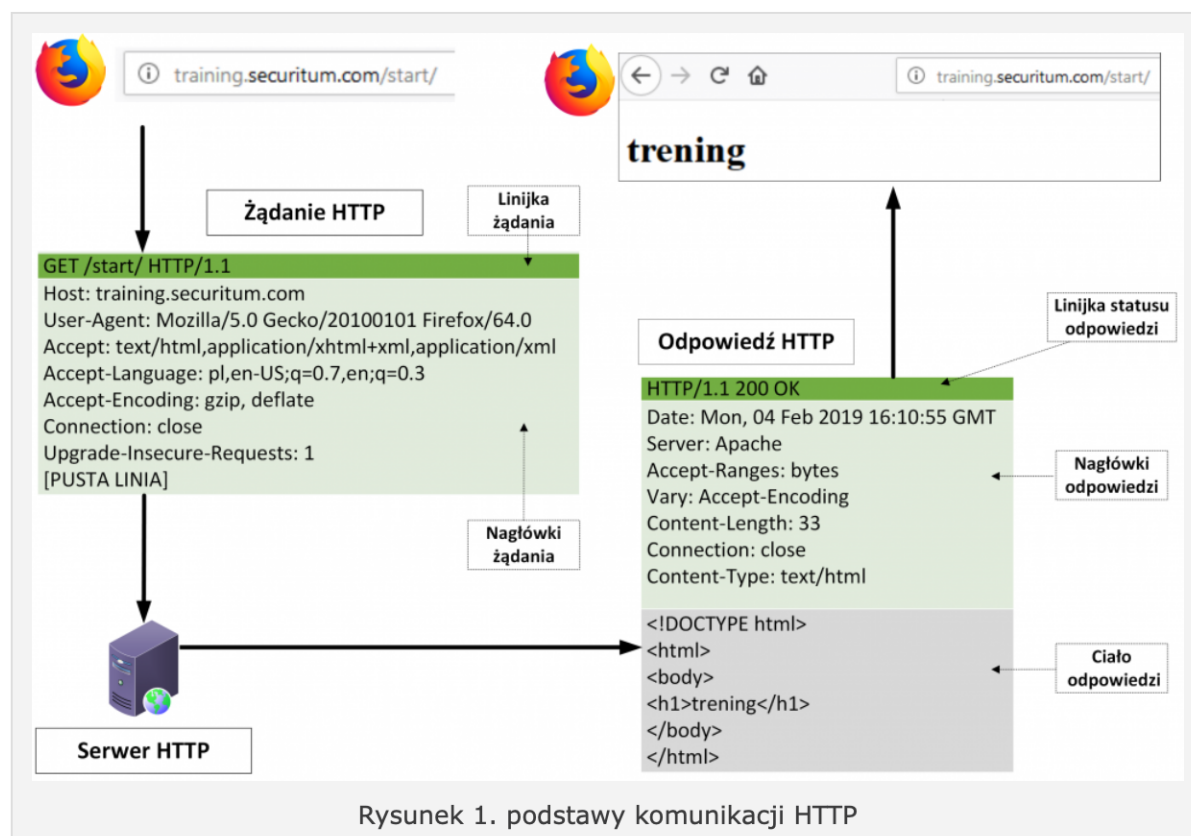
W celu korzystania z komunikacji połączeniowej programy muszą zestawić połączenie, zanim za jego pośrednictwem będą mogły przesyłać dane.

Protokół TCP musi wysłać żądanie otworzenia połączenia TCP ze wskazanym programem. Dopiero po nawiązaniu połączenia program może zacząć przysyłać dane. Po zakończeniu wymiany informacji połączenie należy zamknąć.

Za pomocą interfejsu bezpołączeniowego program może wysłać komunikat do innego programu w dowolnej chwili, bez żadnych czynności wstępnych. Przy korzystaniu z tego interfejsu program, wysyłając każdy komunikat, musi wskazywać adresata. Na przykład w zestawie protokołów TCP/IP przesyłanie bezpołączeniowe jest realizowane przez protokół UDP (ang. User Datagram Protocol). Program użytkowy korzystający z UDP może wysłać ciąg komunikatów, z których każdy będzie skierowany do innego odbiorcy.

## Jak wygląda komunikacja z użyciem protokołu HTTP?

Protokół HTTP (ang. Hypertext Transfer Protocol) to zasady wymiany informacji i współpracy programów. Programami są serwery i klienci. Programy te wysyłają żądania (klienci) lub odpowiedzi (serwery). Przykładem klienta HTTP może być przeglądarka internetowa. Klienci mogą interpretować uzyskane odpowiedzi, na przykład przeglądarka internetowa potrafi wyświetlić stronę internetową, która została przesłana przez serwer.



## Zadanie1

Server → Client

```
MacBook-Pro-Marina:untitled4 marinalukacik$ gcc -o client client.c
MacBook-Pro-Marina:untitled4 marinalukacik$ ./client MacBook-Pro-Marina.local
Received: Hello, world!
MacBook-Pro-Marina:untitled4 marinalukacik$
```

```
MacBook-Pro-Marina:untitled4 marinalukacik$ gcc -o server server.c
MacBook-Pro-Marina:untitled4 marinalukacik$ ./server
server: got connection from 127.0.0.1
MacBook-Pro-Marina:untitled4 marinalukacik$
```

```
From server: Confirmed
how are you
From server: Confirmed
From server: Confirmed
nhgre bergkjbk
From server: Confirmed
```

```
Received: how
Received: areyou
Received: areyouok
Received: hihfrjnfkebkrbjkbevkjbr
Received: how
```

## Zadanie2

### Browser

```
d4 marinalukacik — lukamary@charon: ~ — ssh lukamary@charon.kis.agh.edu.pl — 147x48

URL: wikipedia.pl
Received: HTTP/1.1 200 OK
Date: Wed, 20 May 2020 23:12:28 GMT
Server: Apache
X-Powered-By: PHP/7.0.23
Content-Language: pl
Content-Length: 2745
Connection: close
Content-Type: text/html; charset=utf-8

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl" dir="ltr">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Serwer narzędziowy - Wikimedia Polska - tools.wikimedia.pl</title>
    <link rel="stylesheet" type="text/css" href="/modern.css" />
  </head>
  <body>
    <!-- heading -->
    <div id="mw_header">
      <h1 id="firstHeading">Serwer narzędziowy - Wikimedia Polska</h1>
    </div>
    <div id="mw_main">
      <div id="mw_contentwrapper">
        <!-- content -->
        <div id="mw_content">
          
          <p>Ten serwer wspiera rozwój i działanie narzędzi do analizowania i ulepszania zawartości projektów Wikimedia.</p>
          <p>Serwer sponсорuje Stowarzyszenie Wikimedia Polska.</p>
          <p>Na serwerze między innymi: <a href="http://dvd.wikimedia.pl">Polska Wikipedia na DVD</a>.</p>
          <p>Strony użytkowników: </p>
          <table>
            <tr>
              <td style="vertical-align:top"><ul>
                <li><a href="/~a_bach">a_bach</a></li>
                <li><a href="/~adziura">adziura</a></li>
                <li><a href="/~akbot">akbot</a></li>
                <li><a href="/~akira">akira</a></li>
                <li><a href="/~annamk">annamk</a></li>
                <li><a href="/~beau">beau</a></li>
                <li><a href="/~comp1089">comp1089</a></li>
                <li><a href="/~csteipp">csteipp</a></li>
              </ul>
            </td>
          </table>
        </div>
      </div>
    </div>
  </body>
</html>
```