Technologie Sieciowe
Lista 5

Mateusz Kościelniak

1. Omówienie skryptu.

Najpierw definiujemy adres połączenia oraz port na jakim będziemy odbierać żądania. Następnie zostaje odpalona pętla która warunkuje ciągłość połączenia, dalej mamy pętle z warunkiem otrzymania żądania „get_request”. Jeśli klient wysyła żądanie „GET” to w odpowiedzi zostaje załadowany plik „index.html” i wysłany do klienta, w przeciwnym wypadku klient dostaje kod błędu 403 - „RC_FORBIDDEN” .

```
use HTTP::Daemon;
use HTTP::Status;
#use IO::File;

my $d = HTTP::Daemon->new(
        LocalAddr => 'localhost',        #ustawiamy nazwe naszego hosta
        LocalPort => 4321,               #port na którym będzie odbierał nasz serwer
    )|| die;

print "Please contact me at: <URL:", $d->url, ">\n";


while (my $c = $d->accept) {             #zmienna $c przechowuje wskaznik do połączenia z klientem
    while (my $r = $c->get_request) {    #$r przechowuje wskaznik do żądania
        if ($r->method eq 'GET') {       #jeśli klient dostanie wyśle żądanie 'GET' w odpowiedzi w oknie przeglądarki
                                         #zostanie mu wyświetlona treść pliku index.html
            $file_s= "./index.html";     #index.html - jakis istniejacy plik
            $c->send_file_response($file_s);

        }
        else {
            $c->send_error(RC_FORBIDDEN)
        }

    }
    $c->close;
    undef($c);
}
```

2. Połączenie za pomocą przeglądarki.

Wszedłem na adres który wyświetlił mi się podczas uruchomienia skryptu „server3.pl’”, w ten sposób nawiązałem połączenie z serwerem. W oknie przeglądarki wyświetliła mi się zawartość pliku „index.html”.

3. Mój własny serwer

Serwer napisałem w języku programowania „JAVA”. Aby połączyć się z serwerem, należy w przeglądarce wpisać adres: „http://localhost:1235/myserver?home”. Serwer obsługuje żądania klienta i dodatkowo wyświetla nagłówek jego żądania. W katalogu „my server” zostały umieszczone trzy pliki „html”. Po otwarciu jednej ze stron możliwe jest przechodzenie pomiędzy stronami poprzez umieszczone na nich odnośniki.

- Accept encoding – określa rodzaj kodowania za pomocą którego zostanie przesłana zawartość
- Accept - zawiera akceptowane przez przeglądarke treści
- Connection - informuje o rodzaju nawiązanego połączenia
- Refer – zawiera adres strony z której nastąpiło przekierowanie
- Host – zawiera domenę do której wysyłane jest żądanie
- User agent – zawiera informacje o programie za pomocą którego dokonano zapytania
- Accept language – określa w jakim języku klient chce czytać strony

4. Użycie ‘sniffera’

Do przechwycenia komunikatu użyłem programu „wireshark”, używając filtra HTTP.

Komunikacja klienta i serwera:

```
 32 4.103380387   ip6-localhost        ip6-localhost        HTTP      580 GET /myserver?home HTTP/1.1
 36 4.106037103   ip6-localhost        ip6-localhost        HTTP     1177 HTTP/1.1 200 OK
 49 6.887540186   ip6-localhost        ip6-localhost        HTTP      579 GET /myserver?first HTTP/1.1
 52 6.889185976   ip6-localhost        ip6-localhost        HTTP     1179 HTTP/1.1 200 OK
```

Żądanie klienta :

```
▸ Frame 49: 579 bytes on wire (4632 bits), 579 bytes captured (4632 bits) on interface 0
▸ Linux cooked capture
▸ Internet Protocol Version 6, Src: ip6-localhost (::1), Dst: ip6-localhost (::1)
▸ Transmission Control Protocol, Src Port: 36390, Dst Port: 1235, Seq: 493, Ack: 1168, Len: 491
▾ Hypertext Transfer Protocol
  ▾ GET /myserver?first HTTP/1.1\r\n
    ▸ [Expert Info (Chat/Sequence): GET /myserver?first HTTP/1.1\r\n]
      Request Method: GET
    ▸ Request URI: /myserver?first
      Request Version: HTTP/1.1
  Host: localhost:1235\r\n
  Connection: keep-alive\r\n
  Upgrade-Insecure-Requests: 1\r\n
  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/74.0.3729.157 Safari/537.36\r\n
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3\r\n
  Referer: http://localhost:1235/myserver?home\r\n
  Accept-Encoding: gzip, deflate, br\r\n
  Accept-Language: pl-PL,pl;q=0.9,en-US;q=0.8,en;q=0.7\r\n
  \r\n
  [Full request URI: http://localhost:1235/myserver?first]
  [HTTP request 2/2]
  [Prev request in frame: 32]
  [Response in frame: 52]
```

Odpowiedź serwera: