

HTTP(S)

HYPERTEXT TRANSFER PROTOCOL

HyperText Transfer Protocol

- Przesyłanie zróżnicowanych rodzajów danych/zasobów (ang. *resource*)
 - strony HTML
 - pliki graficzne, dane multimedialne
 - aplikacje
 - inne
- Zasoby identyfikowane przez
 - URL **Uniform Resource Locator**
<http://www.cs.agh.edu.pl/dydaktyka/index.html>

HTTP

- Protokół **klient-serwer**
 - serwer: serwer WWW, httpd
 - klient:
 - najczęściej przeglądarka HTML
 - specjalizowane aplikacje wykorzystujące HTTP do transferu danych (w tym RESTful services lub SOAP web services)
- Protokół **bezstanowy i bezpołączeniowy**
 - działa w oparciu o model żądanie/odpowiedź
 - po dostarczeniu danych połączenie najczęściej jest zamykane

HTTP

Domyślnie używa *dobrze znanego* portu TCP 80

- można używać inne numery portów, np. 8080
 - <http://www.server.com:8080/>
- możliwe wykorzystanie innego niż TCP, ale **niezawodnego** protokołu transportowego

Pakiet HTTP

- Nie ma ścisłego podziału na pola
- Składowe oddzielone są końcem linii
- Postać:
 - typ operacji, jedna linia
 - zero lub więcej linii z parametrami postaci:
nazwa: wartość
 - pusta linia
 - opcjonalne dane
 - zasób
 - treść formularza

Żądania i odpowiedzi HTTP

- Żądania HTTP
 - GET
 - POST
 - HEAD
 - inne - PUT, DELETE
- Odpowiedzi
 - kod odpowiedzi + tekst
- Po uzyskaniu odpowiedzi połączenie TCP między klientem a serwerem najczęściej jest zamykane

Żądania HTTP - GET

- Ciąg znaków identyfikujący zasób na serwerze
 - najprościej: statyczny zasób serwera
GET /dydaktyka/wyniki.html HTTP/1.0
 - parametry do skryptu lub bazy danych
GET /application/script.cgi?key1=value1
&key2=value2 HTTP/1.0
GET /data/record?field1=value1&field2=value2
HTTP/1.0
- Stosowane do przesyłania małych ilości informacji

Żądania HTTP - POST

- Żądanie nie jest zawarte w URL lecz w samym ciele informacji

POST /dane/script.cgi HTTP/1.0

Content-Type: application/x-www-form-urlencoded

Content-Length: 76

home=Cosby&favorite+flavor=flies

Dane z wypełnionego formularza będą przetworzone przez skrypt

- Często używane przy pobieraniu informacji dla stron generowanych dynamicznie lub do wysyłania formularzy

Żądania HTTP - HEAD

- Analogicznie jak GET; zwraca jedynie **nagłówek** strony, nie sam zasób
- Przydatne do zorientowania się w zawartości strony przed jej pobraniem (lub zamiast pobrania)

Inne żądania HTTP

- PUT
 - zapisuje dołączony zasób pod podanym URL
- DELETE
 - usuwa zasób podany w URL
 - nie ma gwarancji wykonania akcji

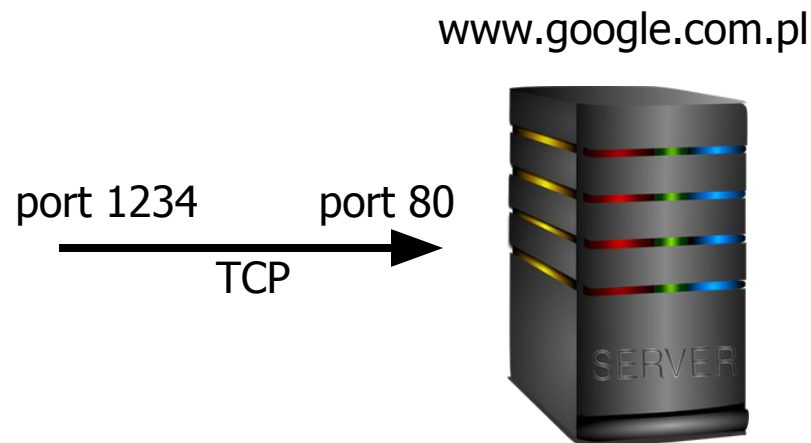
Odpowiedzi HTTP

- **1xx** - informacja
- **2xx** - powodzenie, żądanie zrozumiane i zaakceptowane
 - np. *200 OK*
- **3xx** – musi zostać podjęta dalsza akcja
 - np. *301 Moved Permanently, 304 Not Modified*
- **4xx** - błąd po stronie klienta
 - najczęściej *404 Not Found*,
także *403 Forbidden, 401 Unauthorized*
- **5xx** - błąd po stronie serwera
 - np. *500 Internal Server Error, 501 Not Implemented*

Negocjowalne opcje

- Żądania od klienta do serwera (**Accept: text/html**)
 - podanych może być kilka akceptowalnych wartości
 - **relative quality factor** (parametr $q=wartość$) określa preferencje klienta
 - możliwa odpowiedź serwera: 406 Not Acceptable
- Przykłady:
 - standard kodowania znaków (Accept-Charset)
Accept-Charset: iso-8859-1,* ,utf-8
 - standard kompresji (Accept-Encoding)
Accept-Encoding: compress, gzip
 - język naturalny (Accept-Language)
Accept-Language: pl, en-gb;q=0.8, en;q=0.7

Przykład połączenia HTTP



GET / HTTP/1.1

Host: www.google.com.pl

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) Firefox/38.0

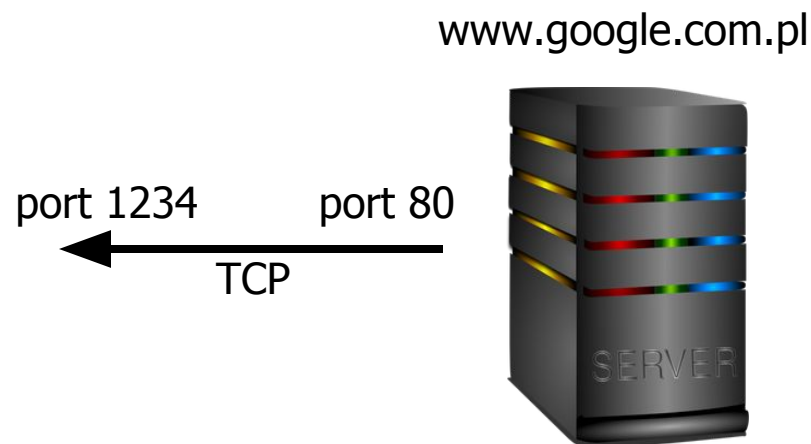
Accept-Language: pl, en-US;q=0.7 # możliwość negocjacji wersji językowej

Accept-Encoding: gzip, deflate

Connection: Keep-Alive

Cookie:

Przykład połączenia HTTP



HTTP/1.0 302 Moved Temporarily

Content-Length: 151

Date: Tue, 07 Jan 2013 18:13:42 GMT

Content-Type: text/html

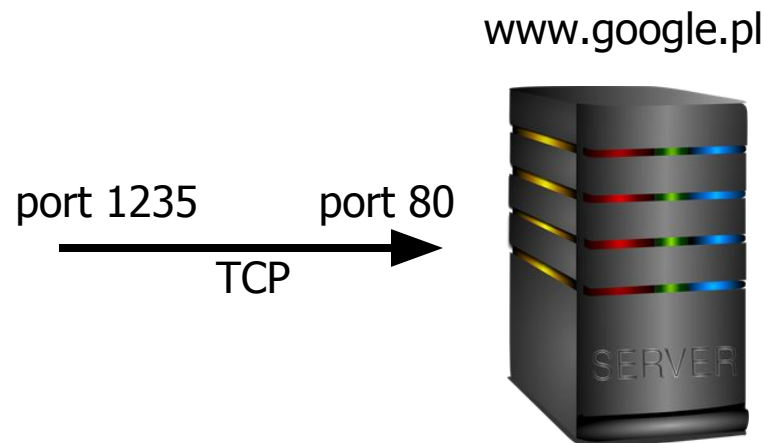
Location: http://www.google.pl

Set-Cookie: ...

poniżej dane (151 B)

```
<HTML><HEAD><TITLE>302 Moved</TITLE></HEAD><BODY><H1>The document has moved <A
  HREF=„http://www.google.pl/”>here</A></BODY></HTML>
```

Przykład połączenia HTTP



GET / HTTP/1.1

Host: **www.google.pl**

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) Firefox/38.0

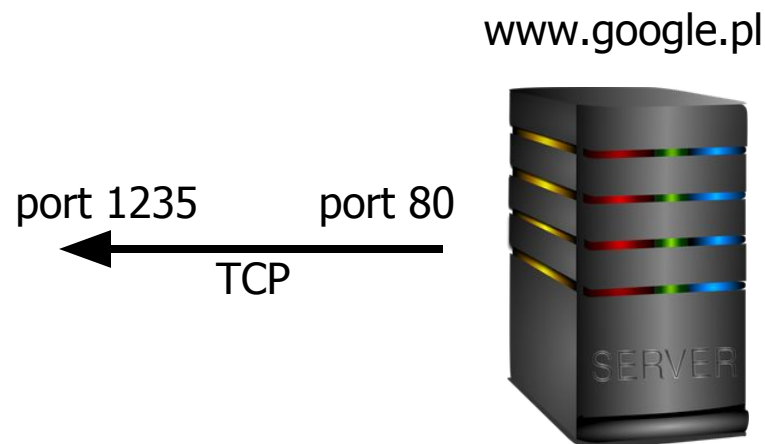
Accept-Language: pl, en-US;q=0.7 # możliwość negocjacji wersji językowej

Accept-Encoding: gzip, deflate

Connection: Keep-Alive

Cookie:

Przykład połączenia HTTP



HTTP/1.0 200 OK

Content-Length: 1420

Date: Tue, 07 Jan 2013 18:13:43 GMT

Content-Type: text/html

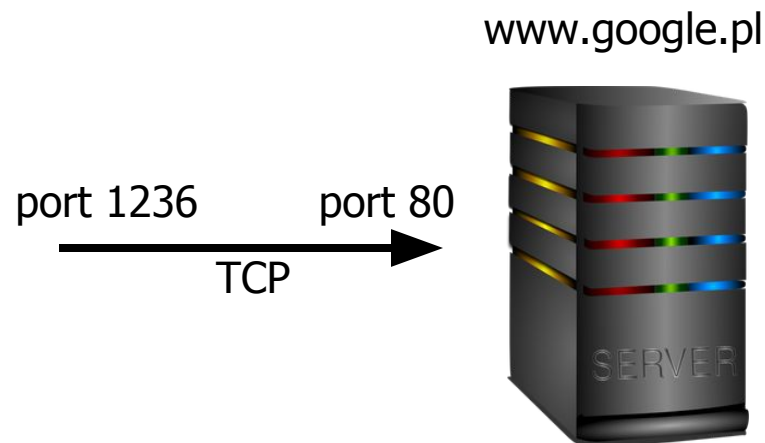
Dane (1228 B)

----- # kontynuacja w następnym pakiecie

Dane (28 B)

----- # kontynuacja w następnym pakiecie

Dane (164 B)



Host: www.google.pl

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) Firefox/38.0

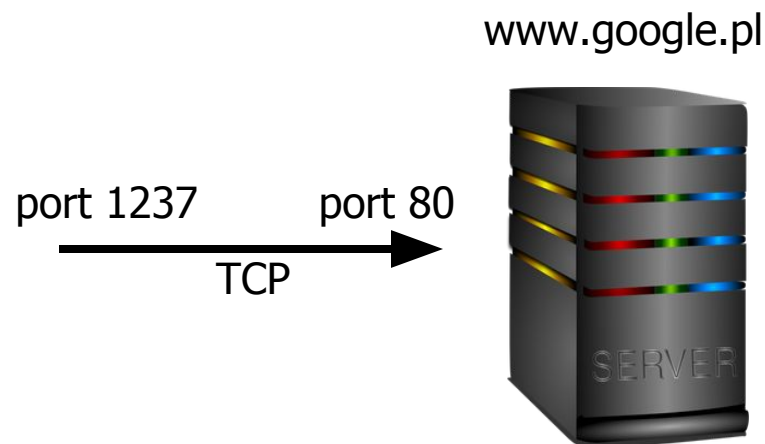
Accept-Language: pl, en-US;q=0.7 *# możliwość negocjacji wersji językowej*

Accept-Encoding: gzip, deflate

Connection: Keep-Alive

Cookie:

Przykład połączenia HTTP



GET /images/hp1.gif HTTP/1.1

If-Modified-Since: Wed, 28 Nov 2011 19:25:06 GMT # jest w buforze

Accept-Language: pl

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) Firefox/38.0

Host: www.google.pl

Connection: Keep-Alive

Cookie:

Przykład połączenia HTTP



port 1236 port 80
← TCP

www.google.pl



HTTP/1.0 200 OK.

Date: Wed, 07 Jan 2013 18:13:44 GMT

Last-Modified: Wed, 28 Nov 2011 19:25:06

Content-Type: image/gif

typ zasobu

Content-Length: 4277

jego długość

Expires: Sun, 17 Jan 2018 19:14:07 GMT

Dane (1198 B)

----- *# kontynuacje w następnych pakietach, stosownie do MSS*

Dane (1460 B)

Przykład połączenia HTTP



port 1237 port 80
← TCP

www.google.pl



HTTP/1.0 304 Not Modified

Content-Length: 0

Date: Thu, 07 Jan 2013 18:13:45 GMT

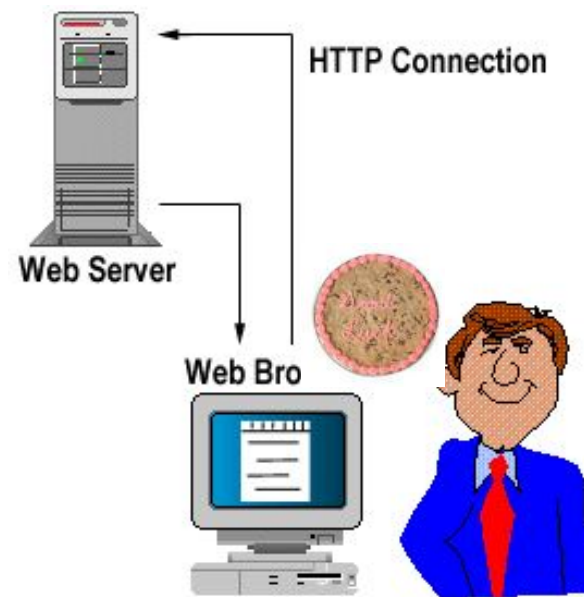
Content-Type: image/gif

Age: 893

nie ma potrzeby wysłać

Ciasteczko - cookie

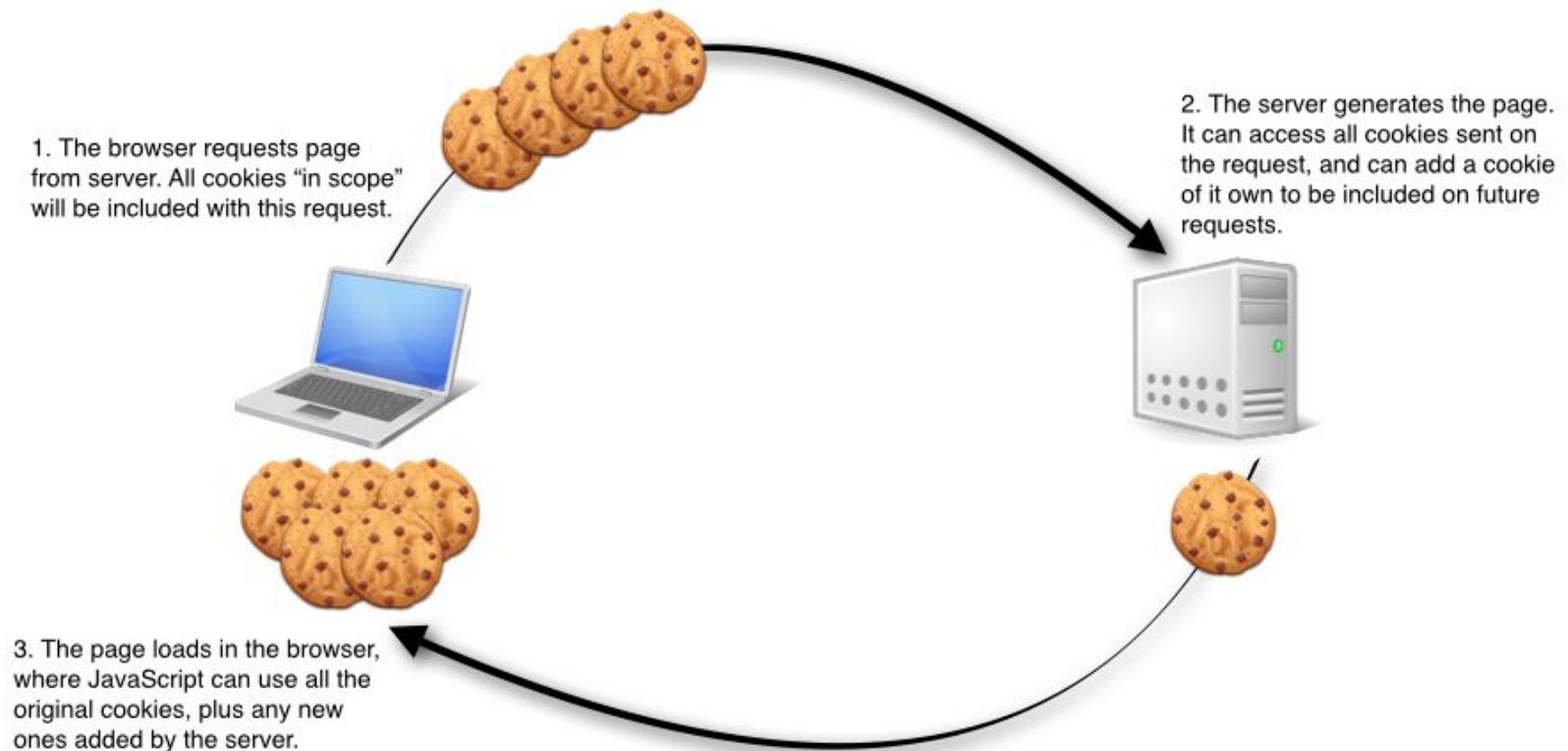
- **Bezstanowość** HTTP - każde odwołanie do serwera HTTP jest takie samo
- Ciasteczka to sposób na:
 - utrzymanie stanu sesji
 - zapewnienia personalizacji stron WWW
 - prowadzenie statystyk przez administratorów serwera
 - ...



Ciasteczka, cd.

- Ciąg znaków przechowywany w pamięci przeglądarki. Jeśli czas ważności jest dłuższy - może być zapisany w pliku
- Przekazywane w nagłówku HTTP
- Wady:
 - związane z pojedynczym komputerem (a nie z użytkownikiem)
 - mogą być łatwo usunięte z komputera użytkownika

Ciasteczko - cookie

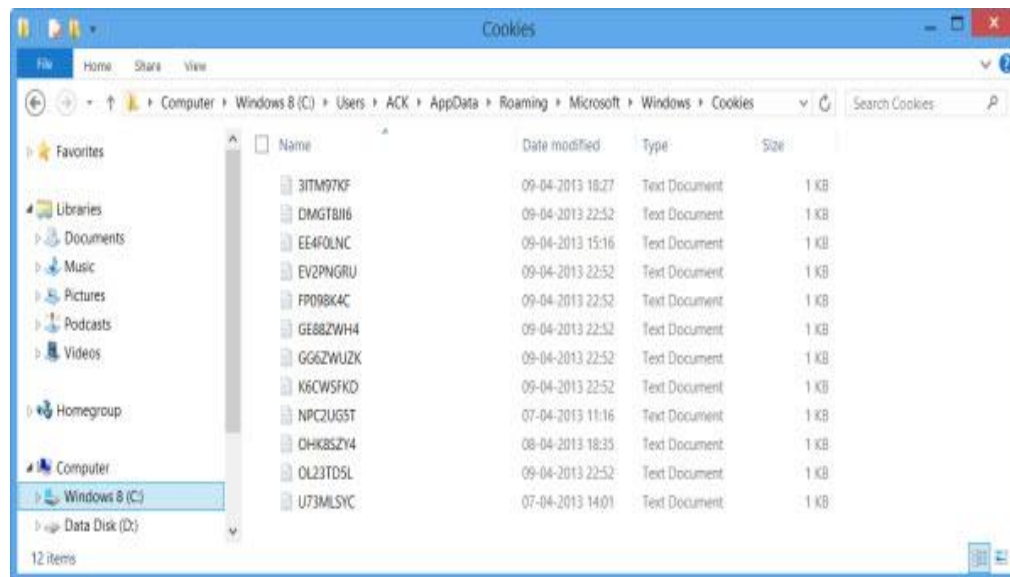


Ciasteczka, cd.

- Przeglądarki mają ograniczenia na ilość i wielkość.
- Według specyfikacji Cookie, ogólne przeglądarki muszą zapewnić obsługę dla minimum:
 - 300 ciasteczek
 - 4 kB na ciasteczko
 - 20 ciasteczek na domenę
- W rzeczywistości:
 - Chrome: 180 / 4096
 - Firefox: 150 / 4097
 - IE: 50 / 5117
 - Opera: 60 / 4096
 - Safari: 600 / 4093

Ciasteczka, cd.

- Ciąg znaków przechowywany w pamięci przeglądarki.
- Jeśli czas ważności jest dłuższy - może być zapisany w pliku
- Przekazywane w nagłówku HTTP



- Wady:
 - związane z pojedynczym komputerem (a nie z użytkownikiem)
 - mogą być łatwo usunięte z komputera użytkownika

Ciasteczka, cd.

- Zawartość: sekwencje par nazwa-wartość
- Przekazywanie od serwera do klienta:

Set-Cookie: *nazwa=wartość; expires=data;*
path=ścieżka; domain=nazwa-domeny;
secure=true/false

secure=true/false - wskazuje, czy ciasteczko ma być używane jedynie w stronach zabezpieczonych kryptograficznie - domyślnie wyłączone

- Przekazywanie od klienta do serwera:

Cookie: *nazwa1=wartość1; nazwa2=wartość2 ...*

- Przeglądarka na podstawie czasu i parametrów pobieranego URL podejmuje decyzję, czy i które ciasteczko przesłać

Ciasteczka - przykład

1. Pierwsze odwołanie do serwera
www.pajacyk.pl

GET /cgi-bin/nzlicz.cgi HTTP/1.0

Referer: http://www.pajacyk.pl

Host: www.pajacyk.pl

Cookie: cc=cc

2. Odpowiedź serwera

HTTP/1.0 200 OK.

Date: Tue, 07 Jan 2003 19:34:46 GMT

Set-Cookie: Mazurek=Mazurek;

expires=Tue, 7-Jan-2003 21:59:59 GMT

Content-type: text/html



Ciasteczka - przykład

3. Kolejne odwołanie do serwera
www.pajacyk.pl (po chwili)

GET /cgi-bin/nzlicz.cgi HTTP/1.0

Referer: http://www.pajacyk.pl

Host: www.pajacyk.pl

Cookie: **Mazurek=Mazurek**; cc=cc

4. Odpowiedź serwera

HTTP/1.0 200 OK.

Date: Tue, 07 Jan 2003 19:44:11 GMT

Set-Cookie: **Mazurek=Mazurek**; **expires=Tue,**
7-Jan-2003 21:59:59 GMT

Content-type: text/html



Dziękujemy !

Już dzisiaj kliknąłeś!
Pamiętaj, odwiedź nas jutro!



Proxy HTTP

- Program pośredniczący między klientem a serwerem HTTP
 - przechwytuje żądania klienta, przekazuje do serwera, a odpowiedź kieruje do klienta
 - możliwość buforowania danych
 - funkcjonalność klienta i serwera równocześnie
- Niezbędne przy ograniczeniach dostępu
- GET musi zawierać kompletny URL
 - nie może być tak:
`GET index.html`
 - musi być tak:
`GET http://www.agh.edu.pl/index.html`

Trochę historii

- Pierwszy dokument HTTP - **HTTP v0.9** , 1991
- **HTTP v1.0** - RFC 1945, 1996
- **HTTP/1.1** - RFC 2068, styczeń 1997
- **HTTP/2** - RFC 7540 , **maj 2015**
Bazuje na eksperymentalnym protokole SPDY rozwiniętym przez Google
- **HTTP/3** - nadchodzi (Internet draft)

Utrzymywanie połączenia TCP

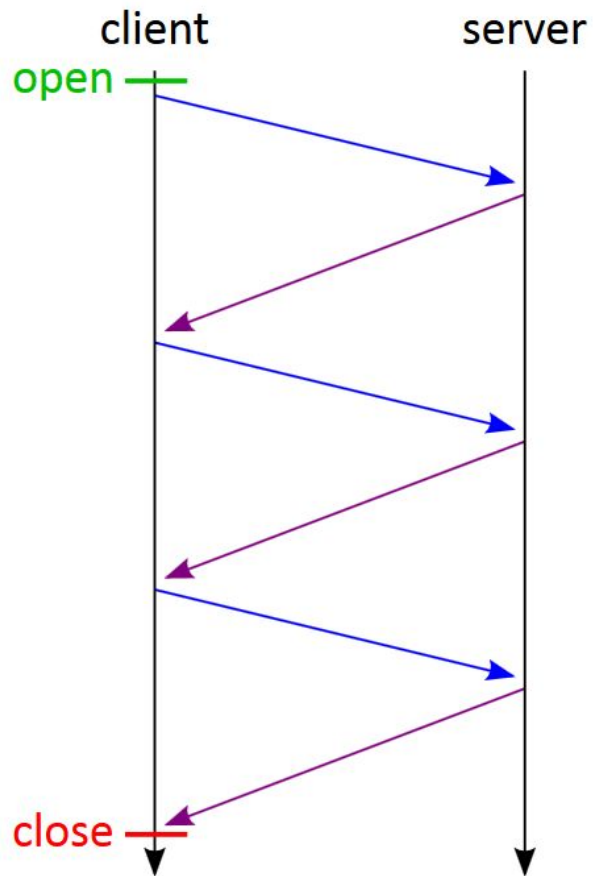
- HTTP 1.0
 - **Connection: Keep-Alive** przesyłane w żądaniu i odpowiedzi
 - domyślne lub jawne **Connection: Close** wygenerowane przez klienta lub serwer zamyka połączenie
- HTTP 1.1
 - domyślne utrzymywanie połączenia (tzw. *persistent connection*)
 - wysłanie **Connection: close** w ostatnim żądaniu
 - czasami używa go serwer - klient powinien zaprzestać używania tego połączenia
 - zamknięcie po czasie nieaktywności

HTTP, wersja 1.1

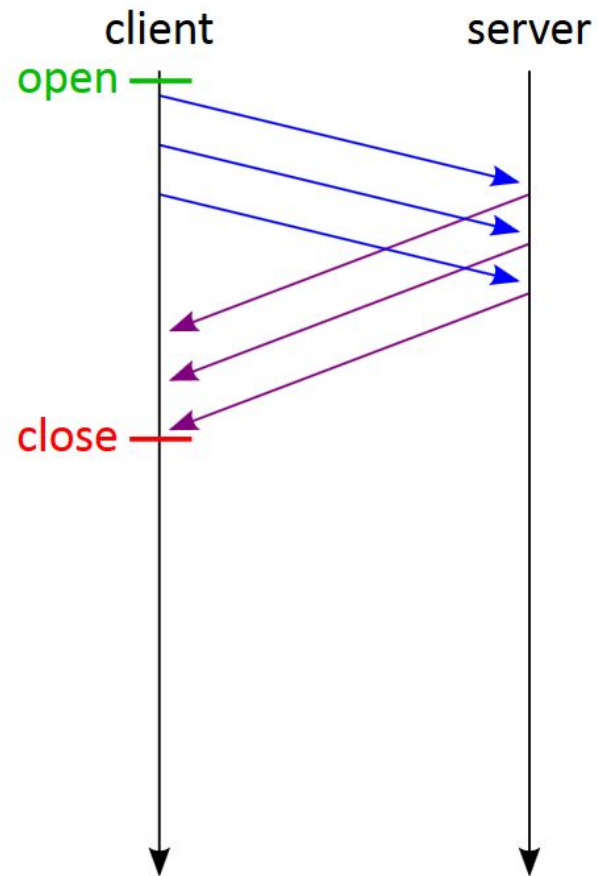
- Wiele poprawek w stosunku do wersji 1.0
 - rozbudowany mechanizm buforowania
 - lepsze wykorzystanie pasma
 - obsługa wielu domen z jednego adresu IP
 - inne
- Serwer HTTP 1.1 musi obsługiwać żądania wersji 1.0

HTTP, wersja 1.1

no pipelining



pipelining



HTTP, wersja 1.1

- Chunked Transfer-Encoding
 - Serwer nie musi czekać aż obliczy długość całego zwracanego zasobu (potrzebna do pola nagłówka Content-Length)
 - Transfer-Encoding: chunked

HTTP/1.1 200 OK.

Date: Fri, 31 Dec 1999 23:59:59 GMT

Content-Type: text/plain

Transfer-Encoding: chunked

1a; ignore-stuff-here

abcdefghijklmnopqrstuvwxy

10

1234567890abcdef

0

some-footer: some-value

another-footer: another-value

[blank line here]

HTTP, wersja 1.1

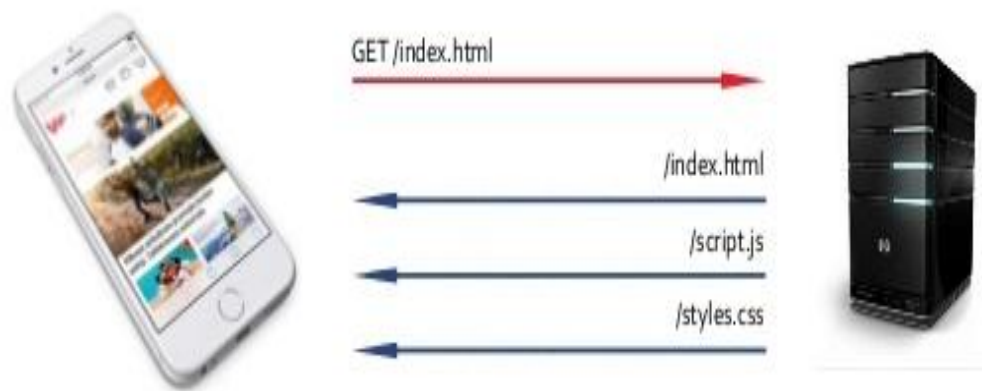
- Wiele domen może być obsługiwanych z jednego adresu IP
 - np. `www.srv-a.com` i `www.srv-b.com`
 - takie adresy są przyjaźniejsze niż rozbudowane URL
 - DNS pozwala na wiązanie różnych nazw domenowych z jednym adresem IP
 - GET nie musi zawierać pełnej nazwy (z adresem hosta) - wymóg kompatybilności z wersją 1.0
 - w żądaniu musi być zawarta informacja o hoście

```
GET /pogoda.html HTTP/1.1
```

```
Host: www.onet.pl
```

Co nowego w HTTP/2?

- Multiplexing żądań - wiele *równoległych* żądań w jednym połączeniu
- Kompresja nagłówków HTTP
- Serwer może wypchać dane do klienta (*HTTP/2 Server Push*)



HTTP/3

- **Internet draft**
“Hypertext Transfer Protocol (HTTP) over QUIC”
- supported by 71% of running web browsers
 - Chrome - September 2019 (April 2020),
 - Firefox - November 2019 (April 2021)
- QUICK (Quick UDP Internet Connections) -
a general-purpose transport layer protocol
 - designed at Google, deployed in 2012, announced in 2013.
 - used by more than half of all connections from the Chrome web browser to Google's servers.

Protokół HTTPS

- Wykorzystuje SSL (Secure Socket Layer) / TLS (Transport Layer Security)
 - otwarty standard opracowany przez Netscape
 - rozwiązanie oparte na kluczu publicznym
 - klucz o długości 128 bit (dziś sugerowana 2048 bit)
 - szyfrowanie danych na czas transmisji kluczem sesji
 - uwierzytelnianie klienta i serwera
- Domyślnie używa portu 443 TCP
- Zastosowania:
 - bankowe, biznesowe, medyczne, inne

Dokumenty dotyczące HTTP

- RFC 1945 □ HTTP/1.0
- RFC 2616 □ HTTP/1.1
- RFC 2109 □ cookie
- RFC 2660 □ HTTPS
- RFC 7540 □ HTTP/2