

Protocolul HTTP

Protocolul HTTP reprezintă baza/fundația schimbului de date pentru World Wide Web. Este un protocol de tip client-server, după crearea unei conexiuni de tipul TCP urmează un schimb de date prin **request – response**. Comunicarea se face prin text simplu ASCII.

Când clientul vrea să comunice cu un server se deschide o nouă conexiune TCP (sau mai multe), se trimite un mesaj HTTP, se citește răspunsul de la server, se închide conexiunea sau se reutilizează.

Structura unui mesaj **request**: metode (Post, Get, Put, Delete), URI -Uniform Resource Identifier (care poate fi URL sau URN), header (proprietăți cheie-valoare), empty line(separator), body (mesajul trimis care poate avea mai multe forme).

Structura unui mesaj **response**: versiunea protocolului, un cod de succes (2xx) sau de eroare (4xx – eroare la client, 5xx – eroare la server) și un mesaj, header (proprietăți cheie-valoare), empty line (separator), body.

Metodele Get, Put, Delete sunt “**idempotent**”, adică mai multe request-uri identice ar trebui să aibă același efect ca un singur request, iar starea sistemului să rămână aceeași.

Protocolul HTTP este “**stateless**”, adică serverul nu trebuie să rețină date, fiecare request este independent, iar pentru a menține coerența unor aplicații web se folosesc cookies sau diferite variabile ascunse.

Proxy-uri – Între user și server stau o serie de calculatoare, routere și alte diferite instrumente. Datorită structurii pe straturi (layers) a protocolului majoritatea acestora nu sunt vizibile, ele aparțin de nivelele de transport, network sau fizic, dar aduc un aport semnificativ performanței. Calculatoarele, router-ele ce sunt la nivelul **aplicație** se numesc **Proxy-uri** și pot efectua: caching, filtrare (antivirus), load balancing (permite serverelor să efectueze task-uri multiple), autentificare, logging (stocarea informațiilor privind istoricul).

HTTP poate “**controla**” o serie de caracteristici pentru o mai bună funcționalitate a WEB-ului: cum și pentru cât timp sunt cache-uite diferite documente, slăbește constrângerile legate de origine conform cărora doar paginile cu aceeași origine pot accesa toată informația de pe o pagină web, autentificarea, proxy și tunneling, sesiuni.

În mod uzual se foloseau 2 API-uri bazate pe HTTP. Cel mai popular dintre ele, **XMLHttpRequest**, se folosește pentru schimbul de date între client și server. API-ul **server-sent events** este folosit pe o singură direcție pentru a trimite un eveniment unui client de către un server, folosind protocolul ca mecanism de transport.

Un minus al protocolului HTTP este acela că acesta nu este sigur, schimbul de date poate fi interceptat. Acest lucru reprezintă o problemă mare deoarece se pot fura date importante (personale, bancare, etc.). **HTTPS** rezolvă această problemă prin certificate **SSL** (secure socket layer) care creează o conexiune criptată. În cazul în care se fură totuși date, acestea vor fi criptate și nu vor putea fi descifrate. Pe lângă SSL, HTTPS folosește și protocolul **TLS (Transport Layer Security)** care ajută la menținerea datelor integre. Port-ul pentru HTTP este 80, iar pentru HTTPS este 443. O altă diferență importantă este faptul că HTTPS activează pe nivelul transport.