Constrained Application Protocol

Server

**Students:**

Chelarașu Elena-Denisa

Miron Alexandru

**Professor:**

Nicolae-Alexandru Botezatu

**Introducere**

Protocolul CoAP are la bază modelul client-server. Clientul transmite cereri serverului, iar serverul le procesează și trimite mesaje de răspuns. Tipurile de cereri ce pot fi transmise sunt asemănătoare cu cele de la HTTP: GET, POST, PUT, DELETE, PATCH, etc.

CoAP folosește datagrame UDP pentru mesaje, opțional folosind și DTLS pentru securitatea comunicațiilor.

User Datagram Protocol (UDP) este un membrul de bază al suitei de protocoale Internet. Datagramele UDP pot fi trimise fără a stabili o conexiune explicită între sursă și destinatar, însă nu oferă garanții ca protecția contra mesajelor duplicate, pierderea mesajelor, sau sosirea lor în ordine inversă.

**Scopul proiectului**

Obiectivul primar al acestui proiect este implementarea unui server CoAP capabil să furnizeze clienților CoAP informații despre fișierele disponibile pe server, precum și executarea unor operațiuni precum crearea, ștergerea, redenumirea fișierelor, etc.

Un alt obiectiv este colaborarea cu echipa ce implementează clientul CoAP. Împreună vom decide formatul mesajelor trimise, operațiile ce vor fi puse la dispoziție și alte detalii de implementare, precum și testarea aplicațiilor rezultante.

**Obiective**

* Implementarea serverului CoAP
* Implementarea operațiilor de tip browser FS
* Comunicarea cu un client CoAP
* Comunicarea cu mai mulți clienți CoAP în același timp
* Implementarea în detaliu (dacă nu deplin) a protocolului CoAP

**Tehnologii folosite**

Aplicația va fi realizată în Python 3, IDEul folosit fiind PyCharm de la JetBrains.

Ca module adiționale, vom folosi **sockets** pentru primirea și trimiterea pachetelor UDP, **tkinter** pentru interfața grafică, precum și **os** pentru operațiile FS.

**Operații filesystem suportate**

Ne propunem să suportăm următoarele operațiuni pe filesystem:

* cd - Schimbă directorul curent
* ls - Listează fișierele din directorul curent
* touch - Crează un fișier gol
* mkdir - Crează un director gol
* move - Mută / redenumește un fișier sau director
* rm - Șterge un fișier sau director
* open - Citește un fișier
* save - Salvează un fișier

În plus, vom expanda codurile de cereri CoAP prin metoda nouă SEARCH, cu codul 0.08, care va realiza o căutare din directorul curent și va returna fișierele / directoarele găsite.

**Detalii de implementare pentru server**

* Restricționarea folderelor disponibile clientului (va fi stabilit un director root)
* Implementarea codurilor de răspuns specifice serverului
* ... și altele