Constrained Application Protocol

Server

**Students:**

Chelarașu Elena-Denisa

Miron Alexandru

**Professor:**

Nicolae-Alexandru Botezatu

**Co-AP Server Introduction**

Co-AP follows a client-server model. The clients send requests to the server and the server responds to tose requests. Usually, clients get to GET, PUT, POST ad DELETE resources from the server. This protocol is designed to fulfill the needs of constrained devices.

Clients and servers communicate through connectionless datagrams and it runs over UDP datagrams. Because of this, SSL/TLS are not available to provide security.

User Datagram Protocol (UDP) is a core member of the Internet protocol. It is a simple message-oriented transport layer, documented in RFC 768. A downside of it is that it provides no guarantee to the upper layer protocol for message delivery and it retains no state of UDP messages once sent.

**Objective**

The primary objective of this project is the implementation of a Co-AP Server (Constrained Application Protocol). This is a specialized protocol usually used for constrained devices.

**Objectives**

-implementation of the server

-being able to connect to multiple clients

-being able to operate in constrained environments

-provide file system resources to the client

**Technologies used**

The programming language this server is developed in is Python 3.

The used libraries/ modules for this project:

* sockets - this module is used for networking , for the receiving/ sending packets
* tkinter – is used to the graphical interface of the program
* os – for file system

The used IDE is PyCharm from JetBrains.

### COAP SERVER

The primary objective of this project is the implementation of a Co-AP Server (Constrained Application Protocol).

This is a specialized protocol usually used for constrained devices.