SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

Course Name – 19IT503 Internet of Things

III Year / V Semester

Unit 3 – EVOLVING IoT STANDARDS & PROTOCOLS

Topic 2- CONSTRAINED APPLICATION PROTOCOL

CONSTRAINED APPLICATION PROTOCOL

CoAP

- CoAP is a simple application layer protocol targeted to simple electronic devices (e.g., IoT/M2M things) to allow them to communicate interactively over the Internet.
- CoAP is designed for low power sensors and for actuators that need to be controlled or monitored remotely, using IP/Internet networks.
- It can be seen as a specialized web transfer protocol for use with constrained networks and nodes for M2M applications.
- It operates with HTTP (hypertext transfer protocol) for basic support with the web.
- CoAP makes use of two message types, requests and responses.
- CoAP is by default bound to UDP and, optionally, to transmission control protocol (TCP).

CONSTRAINED APPLICATION PROTOCOL

Main goals of CoAP is to design a generic web protocol for the

- constrained environment,
- especially considering energy,
- building automation, and
- other M2M applications

CoAP has the following main features:

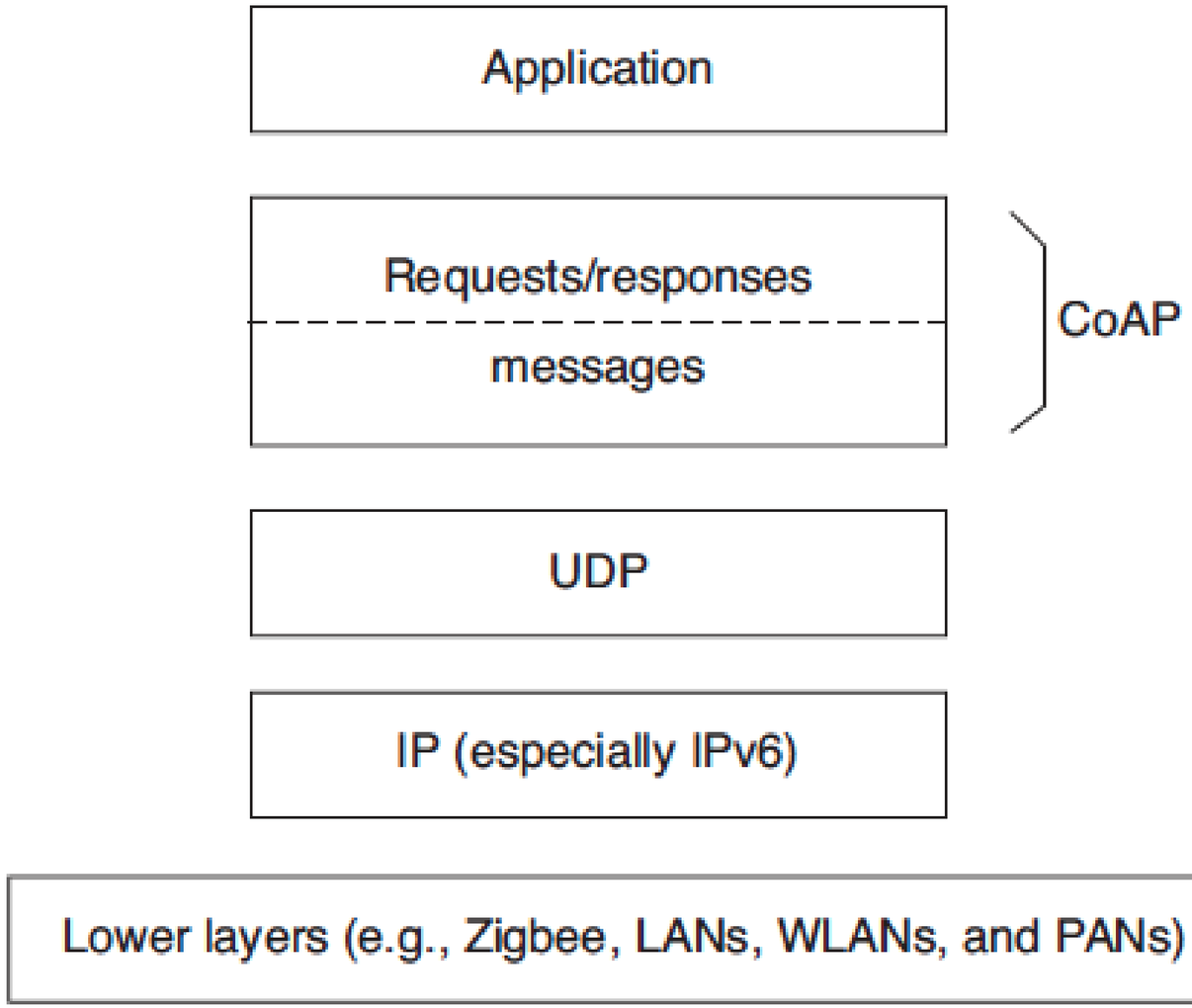
- Constrained web protocol fulfilling M2M requirements;
- UDP binding with optional reliability supporting unicast and multicast requests;
- Asynchronous message exchanges;
- Low header overhead and parsing complexity;
- URI and content-type support;
- Simple proxy and caching capabilities;
- A stateless HTTP mapping, allowing proxies to be built providing access to
- CoAP resources via HTTP in a uniform way or for HTTP simple interfaces to be realized alternatively over CoAP; and
- Security binding to datagram transport layer security (DTLS).

CONSTRAINED APPLICATION PROTOCOL

Abstract layering of CoAP

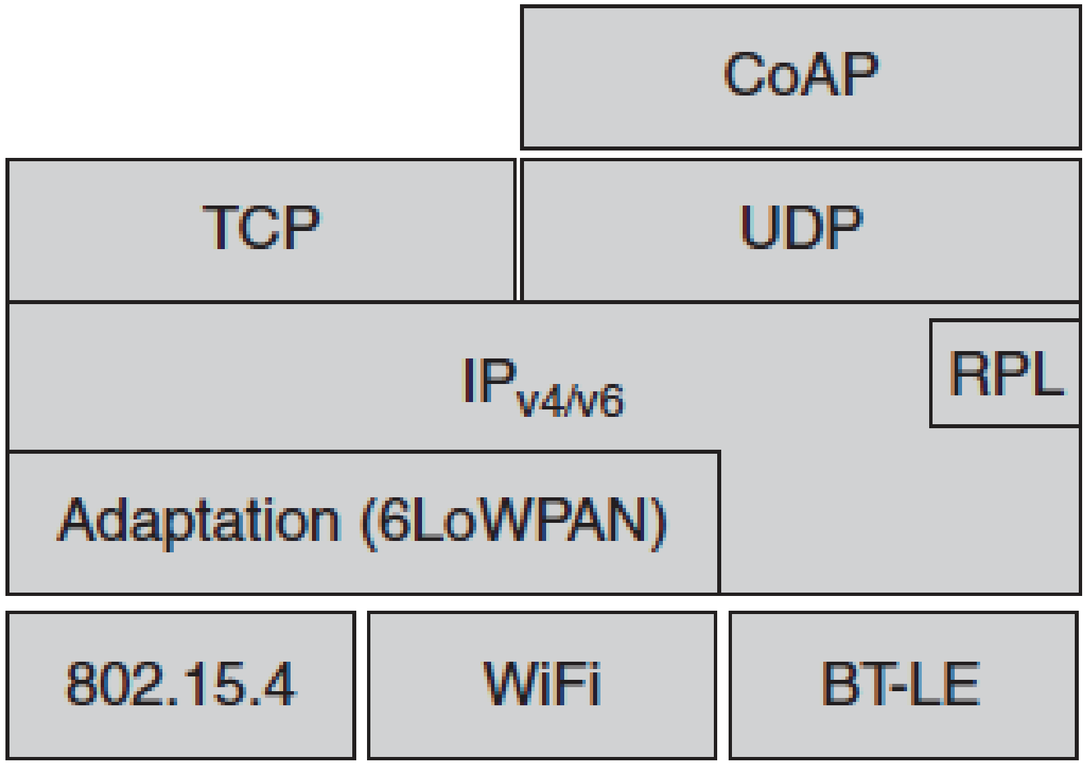
CoAP defines four types of messages:

- confirmable (CON),
- non-confirmable (NON),
- acknowledgement,
- reset



CONSTRAINED APPLICATION PROTOCOL

Protocol Stack of CoAP



CONSTRAINED APPLICATION PROTOCOL

Messaging Model

- The CoAP messaging model is based on the exchange of messages over UDP between end-points.
- It uses a short fixed-length binary header (4 bytes) that may be followed by compact binary options and a payload.
- This message format is shared by requests and responses.
- Each CoAP message contains a message ID used to detect duplicates and for optional reliability
- Reliability is provided by marking a message as CON.
- A CON message is retransmitted using a default timeout and exponential back-off between retransmissions, until the recipient sends an acknowledgement message (ACK) with the same
- message ID from the corresponding end-point.
- When a recipient is not able to process a CON message, it replies with a reset message (RST) instead of an ACK.
- A message that does not require reliable delivery, for example, each single measurement out of a stream of sensor data, can be sent as a NON message.

CONSTRAINED APPLICATION PROTOCOL

Request/Response Model

- CoAP request and response semantics are carried in CoAP messages, which include either a method code or response code, respectively
- A request is carried in a CON or NON message, and if immediately available, the response to a request carried in a CON message is carried in the resulting ACK message.
- If the server is not able to respond immediately to a request carried in a CON message, it simply responds with an empty ACK message so that the client can stop retransmitting the request.
- When the response is ready, the server sends it in a new CON message
- CoAP makes use of GET, PUT, POST, and DELETE methods in a similar manner to HTTP.

CONSTRAINED APPLICATION PROTOCOL

Intermediaries and Caching

- The protocol supports the caching of responses in order to efficiently fulfill requests.

Proxying is useful in constrained networks for several reasons, including

(i) Network traffic limiting,
(ii) to improve performance,
(iii) to access resources of sleeping devices, or
(iv) for security reasons

THANK YOU