MODULE 4

CHAPTER 4

Application Protocols for IoT

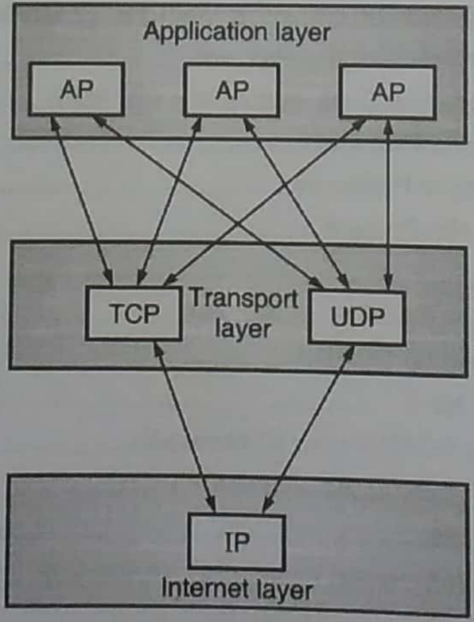
Syllabus

The Transport Layer, IoT Application Transport Methods, Application Layer Protocol Not Present SCADA - Background on SCADA, Adapting SCADA for IP, Tunneling Legacy SCADA over IP Networks, SCADA Protocol Translation, SCADA Transport over LLNs with MAP-T, Generic Web-Based Protocols, IoT Application Layer Protocols – CoAP and MQTT.

4.1 TRANSPORT LAYER

GQ. Explain transport Layer. **(4 Marks)**

- The transport layer is a 4th layer from the top.
- The main role of the transport layer is to provide the communication services directly to the application processes running on different hosts.
- The transport layer provides a logical communication between application processes running on different hosts. Although the application processes on different hosts are not physically connected, application processes use the logical communication provided by the transport layer to send the messages to each other.
- The transport layer protocols are implemented in the end systems but not in the network routers.
- A computer network provides more than one protocol to the network applications. For example, TCP and UDP are two transport layer protocols that provide a different set of services to the network layer.
- All transport layer protocols provide multiplexing/ demultiplexing service. It also provides other services such as reliable data transfer, bandwidth guarantees, and delay guarantees.



(1D1) **Fig. 4.1.1 Transport layer**

- Each of the applications in the application layer has the ability to send a message by using TCP or UDP. The application communicates by using either of these two protocols.
- Both TCP and UDP will then communicate with the internet protocol in the internet layer.

- The applications can read and write to the transport layer. Therefore, we can say that communication is a two-way process.

(1) The Transport Layer

- This section reviews the selection of a protocol for the transport layer as supported by the TCP/IP architecture in the context of IoT networks.
- With the TCP/IP protocol, two main protocols are specified for the transport layer:

(2) Transmission Control Protocol (TCP)

- This connection-oriented protocol requires a session to get established between the source and destination before exchanging data.
- You can view it as an equivalent to a traditional telephone conversation, in which two phones must be connected and the communication link established before the parties can talk.

(3) User Datagram Protocol (UDP)

- With this connectionless protocol, data can be quickly sent between source and destination but with no guarantee of delivery.
- This is analogous to the traditional mail delivery system, in which a letter is mailed to a destination. Confirmation of there ception of this letter does not happen until another letter is sent in response.
- With the predominance of human interactions over the Internet, TCP is the main protocol used at the transport layer.
- This is largely due to its inherent characteristics, such as its ability to transport large volumes of data into smaller sets of packets.
- In addition, it ensures reassembly in a correct sequence, flow control and window adjustment, and retransmission of lost packets.
- These benefits occur with the cost of overhead per packet and per session, potentially impacting overall packet per second performances and latency.

⏭ 4.2 IOT APPLICATION TRANSPORT METHODS

GQ. Explain IOT Application of Transport Methods. **(2 Marks)**

- Because of the diverse types of IoT application protocols, there are various means for transporting these protocols across a network.

- Sometimes you may be dealing with legacy utility and industrial IoT protocols that have certain requirements, while other times you might need to consider the transport requirements of more modern application layer protocols.
- To make these decisions easier, it makes sense to categorize the common IoT application protocols and then focus on the transport methods available for each category.
- The following categories of IoT application protocols and their transport methods are explored in the following sections:

4.2.1 Application Layer Protocol Not Present

In this case, the data payload is directly transported on top of the lower layers. No application layer protocol is used.

4.2.2 Supervisory control And Data Acquisition (SCADA)

SCADA is one of the most common industrial protocols in the world, but it was developed long before the days of IP, and it has been adapted for IP networks.

4.2.3 Generic Web-Based Protocols

Generic protocols, such as Ethernet, Wi-Fi, and 4G/LTE, are found on many consumer- and enterprise-class IoT devices that communicate over non-constrained networks.

4.2.4 IoT Application Layer Protocols

- IoT application layer protocols are devised to run on constrained nodes with a small compute footprint and are well adapted to the network bandwidth constraints on cellular or satellite links or constrained 6LoWPAN networks.
- Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP), covered later in this chapter, are two well-known examples of IoT application layer protocols.

4.3 APPLICATION LAYER PROTOCOL NOT PRESENT

- As introduced in IETF RFC 7228 devices defined as class 0 send or receive only a few bytes of data.
- For myriad reasons, such as processing capability, power constraints, and cost, these devices do not implement a fully structured network protocol stack, such as IP, TCP, or UDP, or even an application layer protocol.

- Class 0 devices are usually simple smart objects that are severely constrained. Implementing a robust protocol stack is usually not useful and sometimes not even possible with the limited available resources.
- For example, consider low-cost temperature and relative humidity (RH) sensors sending data over an LPWA LoRaWAN infrastructure. is represented as 2 bytes and RH as another 2 bytes of data.
- Therefore, this small data payload is directly transported on top of the LoRaWAN MAC layer, without the use of TCP/IP.

4.4 SCADA

GQ. Write Short Note on SCADA. **(2 Marks)**

- In the world of networking technologies and protocols, IoT is relatively new. Combined with the fact that IP is the de facto standard for computer networking in general, older protocols that connected sensors and actuators have evolved and adapted themselves to utilize IP.
- A prime example of this evolution is supervisory control and data acquisition (SCADA). Designed decades ago, SCADA is an automation control system that was initially implemented without IP over serial links, before being adapted to Ethernet and IPv4.

4.4.1 A Little Background on SCADA

- For many years, vertical industries have developed communication protocols that fit their specific requirements.
- Many of them were defined and implemented when the most common networking technologies were serial link-based, such as RS-232 and RS-485.
- This led to SCADA networking protocols, which were well structured compared to the protocols described in the previous section, running directly over serial physical and data link layers.
- At a high level, SCADA systems collect sensor data and telemetry from remote devices, while also providing the ability to control them.
- Used in today's networks, SCADA systems allow global, real-time, data-driven decisions to be made about how to improve business processes.
- SCADA networks can be found across various industries, but you find SCADA mainly concentrated in the utilities and manufacturing/industrial verticals.

- Within these specific industries, SCADA commonly uses certain protocols for communications between devices and applications.
- For example, Modbus and its variants are industrial protocols used to monitor and program remote devices via a master/slave relationship. Modbus is also found in building management, transportation, and energy applications.
- The DNP3 and International Electrotechnical Commission (IEC) 60870-5-101 protocols are found mainly in the utilities industry, along with DLMS/COSEM and ANSI C12 for advanced meter reading (AMR).
- Both DNP3 and IEC 60870-5-101 are discussed in more detail later in this chapter. As mentioned previously, these protocols go back decades and are serial based. So, transporting them over current IoT and traditional networks requires that certain accommodations be made from both protocol and implementation perspectives. These accommodations and other adjustments form various SCADA transport methods that are the focus of upcoming sections.

4.4.2 Adapting SCADA for IP

- In the 1990s, the rapid adoption of Ethernet networks in the industrial world drove the evolution of SCADA application layer protocols.
- For example, the IEC adopted the Open System Interconnection (OSI) layer model to define its protocol framework. Other protocol user groups also slightly modified their protocols to run over an IP infrastructure.
- Benefits of this move to Ethernet and IP include the ability to leverage existing equipment and standards while integrating seamlessly the SCADA subnetworks to the corporate WAN infrastructures. To further facilitate the support of legacy industrial protocols over IP networks, protocol specifications were updated and published, documenting the use of IP for each protocol.
- This included assigning TCP/UDP port numbers to the protocols, such as the following :

  DNP3 (adopted by IEEE 1815-2012) specifies the use of TCP or UDP on port 20000 for transporting DNP3 messages over IP.
- The Modbus messaging service utilizes TCP port 502. IEC 60870-5-104 is the evolution of IEC 60870-5-101 serial for running over Ethernet and IPv4 using port 2404 DLMS User Association specified a communication profile based on TCP/IP in the DLMS/COSEM Green Book (Edition 5 or higher), or in the IEC 62056-53 and IEC 62056-47 standards, allowing data exchange via IP and port 4059.

4.4.3 Tunneling Legacy SCADA over IP Networks

- Deployments of legacy industrial protocols, such as DNP3 and other SCADA protocols, in modern IP networks call for flexibility when integrating several generations of devices or operations that are tied to various releases and versions of application servers. Native support for IP can vary and may require different solutions.
- Ideally, end-to-end native IP support is preferred, using a solution like IEEE 1815-2012 in the case of DNP3.

4.4.4 SCADA Transport over LLNs with MAP-T

- Due to the constrained nature of LLNs, the implementation of industrial protocols should at a minimum be done over UDP. This in turn requires that both the application servers and devices support and implement UDP.
- While the long-term evolution of SCADA and other legacy industrial protocols is to natively support IPv6, it must be highlighted that most, if not all, of the industrial devices supporting IP today support IPv4 only.
- When deployed over LLN subnetworks that are IPv6 only, a transition mechanism, such as MAP-T (Mapping of Address and Port using Translation, RFC 7599), needs to be implemented. This allows the deployment to take advantage of native IPv6 transport transparently to the application and devices. depicts a scenario in which a legacy endpoint is connected across an LLN running 6LoWPAN to an IP-capable SCADA server.
- The legacy endpoint could be running various industrial and SCADA protocols, including DNP3/IP, Modbus/TCP, or IEC 60870-5-104. In this scenario, the legacy devices and the SCADA server support only IPv4 (typical in the industry today).
- However, IPv6 (with 6LoWPAN and RPL) is being used for connectivity to the endpoint. 6LoWPAN is a standardized protocol designed for constrained networks, but it only supports IPv6. In this situation, the end devices, the endpoints, and the SCADA server support only IPv4, but the network in the middle supports only IPv6.

4.5 GENERIC WEB-BASED PROTOCOLS

GQ. Explain Generic Web-Based Protocols. (4 Marks)

- Over the years, web-based protocols have become common in consumer and enterprise applications and services.

- Therefore, it makes sense to try to leverage these protocols when developing IoT applications, services, and devices in order to ease the integration of data and devices from prototyping to production.
- The level of familiarity with generic web-based protocols is high. Therefore, programmers with basic web programming skills can work on IoT applications, and this may lead to innovative ways to deliver and handle real-time IoT data.
- For example, an IoT device generating an event can have the result of launching a video capture, while at the same time a notification is sent to a collaboration tool, such as a Cisco Spark room. This notification allows technicians and engineers to immediately start working on this alert.
- In addition to a generally high level of familiarity with web-based protocols, scaling methods for web environments are also well understood and this is crucial when developing consumer applications for potentially large numbers of IoT devices.
- Once again, the definition of constrained nodes and networks must be analyzed to select the most appropriate protocol. On non-constrained networks, such as Ethernet, Wi-Fi, or 3G/4G cellular, where bandwidth is not perceived as a potential issue, data payloads based on a verbose data model representation, including XML or JavaScript Object Notation (JSON), can be transported over HTTP/HTTPS or WebSocket. This allows implementers to develop their IoT applications in contexts similar to web applications.
- The HTTP/HTTPS client/server model serves as the foundation for the World Wide Web. Recent evolutions of embedded web server software with advanced features are now implemented with very little memory (in the range of tens of kilobytes in some cases). This enables the use of embedded web services software on some constrained devices MQTT and CoAP both are the most popular Internet of Things protocols. During the next post, we will talk about pros and cons of each one.

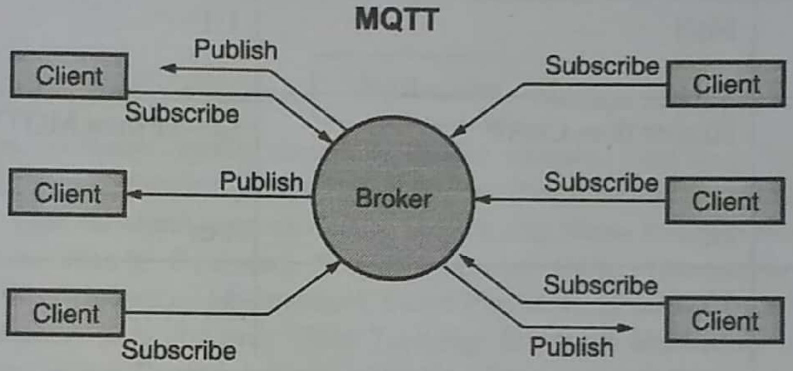
⏭ 4.6 IOT APPLICATION LAYER PROTOCOLS

- When considering constrained networks and/or a large-scale deployment of constrained nodes, verbose web-based and data model protocols, as discussed in the previous section, may be too heavy for IoT applications.
- To address this problem, the IoT industry is working on new lightweight protocols that are better suited to large numbers of constrained nodes and networks. Two of the most popular protocols are CoAP and MQTT

☞ What is MQTT?

GQ. Write a short note on MQTT. **(2 Marks)**

- Message Queue Telemetry Transport (MQTT), is a publish-subscribe protocol that facilitates one-to-many communication mediated by brokers.
- Clients can publish messages to a broker and/or subscribe to a broker to receive certain messages. Messages are organized by topics, which essentially are "labels" that act as a system for dispatching messages to subscribers.

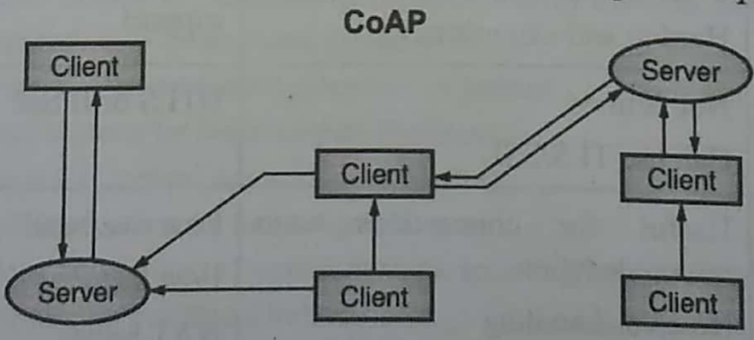


(1D2)**Fig. 4.6.1 MQTT**

4.6.1 What is CoAP?

GQ. Write a short note on COAP. **(2 Marks)**

- Constrained Application Protocol (CoAP), is a **client-server protocol** that, unlike MQTT, is not yet standardized. With CoAP, a **client node can command another node** by sending a CoAP packet.
- The CoAP server will interpret it, extract the payload, and decide what to do depending on its logic. The server does not necessarily have to acknowledge the request.



(1D3)**Fig. 4.6.2 COAP**

The following table **compares** different features and shows the **strengths and debilities** of each protocol :

Features	MQTT	CoAP
Base protocol	TCP	UDP
Model used for communication	Publish-Subscribe	Request-Response Publish-Subscribe
Communication node	M:N	1:1
Power consumption	Higher than CoAP	Lower than MQTT
RESTful	No	Yes
Number of messages type used	16	4
Header size	2 Bytes	4 Bytes
Messaging	Asynchronous	Asynchronous & Synchronous
Reliability	3 Quality of service levels QoS 0: Delivery not guaranteed QoS 1: Delivery confirmation QoS 2: Delivery double confirmation	Confirmable messages Non-confirmable messages Acknowledgements Retransmissions
Implementation	Easy to implement Hard to add extensions	Few existing libraries and support
Security	Not defined Can use TLS/SSL	DTLS or IPSec
Other	Useful for connections with remote location No error-handling	Low overhead Low latency NAT issues

...Chapter ends

□□□