UNIT-II

IOT & M2M

Internet of Things: IOT is known as the Internet of Things where things are said to be the communicating devices that can interact with each other using a communication media. Usually every day some new devices are being integrated which uses IoT devices for its function. These devices use various sensors and actuators for sending and receiving data over the internet. It is an ecosystem where the devices share data through a communication media known as the internet.

Machine to Machine: This is commonly known as Machine to machine communication. It is a concept where two or more than two machines communicate with each other without human interaction using a wired or wireless mechanism. M2M is a technology that helps the devices to connect between devices without using internet. M2M communications offer several applications such as security, tracking and tracing, manufacturing and facility management.

Machine-to-machine, or M2M, is a broad label that can be used to describe any technology that enables networked devices to exchange information and perform actions without the manual assistance of humans. Artificial intelligence (AI) and machine learning (ML) facilitate the communication between systems, allowing them to make their own autonomous choices.

How M2M works

The main purpose of machine-to-machine technology is to tap into sensor data and transmit it to a network. M2M systems often use public networks and access methods -- for example, cellular or Ethernet -- to make it more cost-effective.

The main components of an M2M system include sensors, RFID, a Wi-Fi or cellular communications link, and autonomic computing software programmed to help a network device interpret data and make decisions. These M2M applications translate the data, which can trigger preprogrammed, automated actions.

One of the most well-known types of machine-to-machine communication is telemetry, which has been used since the early part of the last century to transmit operational data. Pioneers in telemetric first used telephone lines, and later, radio waves, to transmit performance measurements gathered from monitoring instruments in remote locations.

The Internet and improved standards for wireless technology have expanded the role of telemetry from pure science, engineering and manufacturing to everyday use in products such as heating units, electric meters and internet-connected devices, such as appliances.

M2M applications and examples

Machine-to-machine communication is often used for remote monitoring. In product restocking, for example, a vending machine can message the distributor's network, or machine, when a particular item is running low to send a refill. An enabler of asset tracking and monitoring, M2M is vital in warehouse management systems (WMS) and supply chain management (SCM).

Utilities companies often rely on M2M devices and applications to not only harvest energy, such as oil and gas, but also to bill customers -- through the use of Smart meters -- and to detect worksite factors, such as pressure, temperature and equipment status.

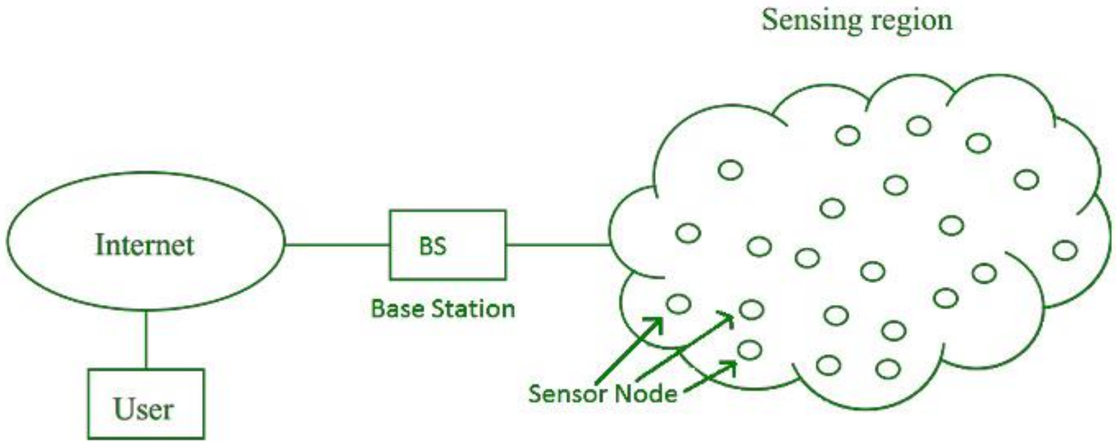
Difference between IOT and M2M:

Basis of	IOT	M2M
	Internet of Things	Machine to Machine
Intelligence	Devices have objects that are responsible for decision making	Some degree of intelligence is observed in this.
Connection type used	The connection is via Network and using various communication types.	The connection is a point to point
Communication protocol used	Internet protocols are used such as HTTP, FTP, and Telnet.	Traditional protocols and communication technology techniques are used
Data Sharing	Data is shared between other applications that are used to improve the end-user experience.	Data is shared with only the communicating parties.
Internet	Internet connection is required for communication	Devices are not dependent on the Internet.

Basis of	IOT	M2M
Type of Communication	It supports cloud communication	It supports point-to-point communication.
Computer System	Involves the usage of both Hardware and Software.	Mostly hardware-based technology
Scope	A large number of devices yet scope is large.	Limited Scope for devices.
Open API support	Supports Open API integrations.	There is no support for Open APIs
Examples	Smart wearable's, Big Data and Cloud, etc.	Sensors, Data and Information, etc.

Wireless Sensor Network (WSN) is an infrastructure-less wireless network that is deployed in a large number of wireless sensors in an ad-hoc manner that is used to monitor the system, physical or environmental conditions.

Sensor nodes are used in WSN with the onboard processor that manages and monitors the environment in a particular area. They are connected to the Base Station which acts as a processing unit in the WSN System. Base Station in a WSN System is connected through the Internet to share data.



WSN can be used for processing, analysis, storage, and mining of the data.

Applications of WSN:

1. Internet of Things (IOT)
2. Surveillance and Monitoring for security, threat detection
3. Environmental temperature, humidity, and air pressure
4. Noise Level of the surrounding
5. Medical applications like patient monitoring
6. Agriculture
7. Landslide Detection

Challenges of WSN:

1. Quality of Service

2. Security Issue
3. Energy Efficiency
4. Network Throughput
5. Performance
6. Ability to cope with node failure
7. Cross layer optimization
8. Scalability to large scale of deployment

Components of WSN:

1. **Sensors:**

   Sensors in WSN are used to capture the environmental variables and which is used for data acquisition. Sensor signals are converted into electrical signals.

2. **Radio Nodes:** It is used to receive the data produced by the Sensors and sends it to the WLAN access point. It consists of a microcontroller, transceiver, external memory, and power source.
3. **WLAN Access Point:** It receives the data which is sent by the Radio nodes wirelessly, generally through the internet.
4. **Evaluation Software:** The data received by the WLAN Access Point is processed by a software called as Evaluation Software for presenting the report to the users for further processing of the data which can be used for processing, analysis, storage, and mining of the data.

Wireless Media Access Issues in Internet of Things

When it comes to communication using a wireless medium there is always a concern about the interference due to other present wireless communication technologies. Wireless means communication and message transfer without the use of physical medium i.e., wires.

Let us understand how communication is done between them. Different Mobile stations (MS) are attached to a transmitter/receiver which communicates via a shared channel by other nodes. In this type of communication, it makes it difficult for the MAC design rather than the wireline networks.

The very important issues which are observed are: Half Duplex operation, Time-varying channel, and Burst channel errors. These are explained as following below.

1. **Half Duplex operation:**

Half-duplex transmission means when the sender and receiver both are capable of sharing data but one at a time. In wireless transmission, it is difficult to receive data when the transmitter is sending the data because during transmission a large amount or a large fraction of signal energy is leaked while broadcasting. The magnitude of the transferred signal and received signal differs a lot. Due to which collision detection is even not possible by the sender as the intensity of the transferred signal is large than the received one. Hence this causes the problem of collision and the prime focus should be to minimize the collision.

2. Time-varying channel: Time-varying channels include the three mechanisms for radio signal propagations they are Reflection, Diffraction, and Scattering.

- **Reflection** –

  This occurs when a propagating wave carrying information intrudes on an object that has very large dimensions than the wavelength of the wave.

- **Diffraction** –

  This occurs when the radio path between the transmitter and the receiver is collided by the surface with sharp edges. This is a phenomenon which causes the diffraction of the wave from the targeted position.

- **Scattering** –

  This occurs when the medium through from the wave is traveling consists of some objects which have dimensions smaller than the wavelength of the wave.

While transmitting the signal by the node these are time shifted and this is called multipath propagation. While when this node signals intensity is dropped below a threshold value, then this is termed as fade. As a result, Handshaking strategy is widely used so as a healthy communication can be set up.

2. **Burst channel errors:** Burst channel errors are called as a contiguous sequence of symbols, which are received in a communication channel, in which the first and last symbols has an error and there is no evidence of contiguous sub-sequence of corrected received symbols. When time-varying channels are used then signals strengths are introduced due to which errors are observed in transmission. For these channels in wireline networks, the Bit rate is high as 10^{-3}.

Sensor Node Deployment Models

Sensor network is capable of monitoring the real-world phenomena at a large scale and embeds the sensor nodes of wireless network in the real world. Deployment is concerned about the set-up process of an operational sensor network in the real world. Deployment is cumbersome and labor-intensive as bugs may be triggered or the performance is degraded that was not observed at the time pre-deployment. The real world has an influence on the functionality of sensor network by influencing the quality of communication links, tracking the output of sensors, and by having physical strain on wireless sensor nodes. Deploying sensors by covering the complete area is considered as a design problem in various WSN applications. The three common deployment approaches are random deployment, deterministic deployment and grid (pattern) based deployment. Deterministic deployment is suitable for small-scale applications because of deliberate location of sensors. Non-deterministic deployment is commonly suitable for large-scale applications.

- Random deployment: It is quite challenging to locate the spot for each device since there is no prior configuration in randomized sensor deployment. In uniform random deployment, the ‘n’ number of sensors has an opportunity to place the sensor node at any point in a given field. WSN applications prefer uniform random method because of ease deployment and cost-effective.
- Grid Deployment: The most popular grid layouts followed in grid deployment are a unit square, triangle, hexagon etc. Grid deployment is suitable for several WSN applications due to its coverage performance. This kind of deployment is performed by placing sensors row-by-row based on moving carrier. The time interval is maintained between consecutive droppings to obtain the desired distance. However, this deployment model is not ideal because of placement errors.

Data aggregation & dissemination

Data aggregation with IoT gathers data from various kinds of devices/ objects and shared with multiple devices for processing in several IoT applications. The reliable and efficient data aggregation techniques generally increase the network lifetime using appropriate sensors. The data acquisition process is facilitated in different technologies that consist of sensors, actuators, camera, RFID, GPS etc. The short-range communications enables information sharing among heterogeneous devices in IoT environment. IoT devices cover a wide range of applications across the globe, such as agriculture, transportation, healthcare, market, industry, smart school, smart home, smart city etc.

The information about the particular location is significant to know the current situation. Localization in WSN is the process of determining the location of unknown network sensor nodes. Localization plays role in many practical applications by determining the location of patients, equipments or personnel in a hospital, swarm of robots work together toward a common goal. GPS is the straightforward solution for localization but it is not suitable for all applications. Router communicates the identity of its location and it can be realized using the two most significant bits of one byte data structure. The following possibilities are enabled: indoor and outdoor in a public area, personal indoor area, restricted area.

In indoor systems, there is no direct connection from GPS signal to satellites. However, there are obstacles in many situations where it blocks the direct communication with the GPS satellites in outdoor systems. Some other issues of GPS in WSN nodes are cost, size, and power consumption. Moreover, in various WSN applications, the localization problem is solved using network parameters, features of the received radio signal and location of fixed nodes (known as anchors or beacons). The localization problem is categorized into two levels:

i. Distance between two nodes are estimated

ii. Location of all unknown nodes is determined.

Routing Protocols

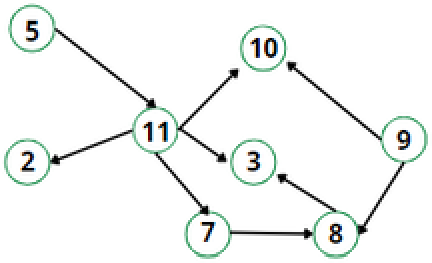
In this section, we discuss some standard and non-standard protocols that are used for routing in IoT applications. It should be noted that we have partitioned the network layer in two sublayers: routing layer which handles the transfer the packets from source to destination, and an encapsulation layer that forms the packets. Encapsulation mechanisms will be discussed in the next section.

RPL

Routing Protocol for Low-Power and Lossy Networks (RPL) is distance-vector protocol that can support a variety of datalink protocols, including the ones discussed in the previous section. RPL stands for Routing Protocol for Low Power and Lossy Networks for heterogeneous traffic networks. It is a routing protocol for Wireless Networks.

It is a Distance Vector Routing Protocol that creates a tree-like routing topology called the Destination Oriented Directed Acyclic Graph (DODAG), rooted towards one or more nodes called the root node or sink node.

The Directed Acyclic Graphs (DAGs) are created based on user-specified specific Objective Function (OF). The OF defines the method to find the best-optimized route among the number of sensor devices.



In an RPL Network, each node acts as a router and becomes part of a mesh network. Routing is performed at the IP Layer. Each node examines every received IPv6 packet and determines the next-hop destination based on the information contained in the IPv6 header. No information from the MAC layer header is needed to perform the next determination.

Modes of RPL:

This protocol defines two modes:

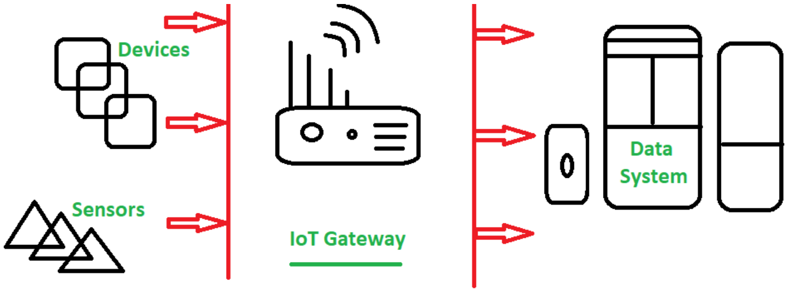
1.**Storing mode:** All modes contain the entire routing table of the RPL domain. Every node knows how to reach every other node directly.

2**. Non-Storing mode:** Only the border router(s) of the RPL domain contain(s) the full routing table. All other nodes in the domain maintain their list of parents only and use this as a list of default routes towards the border router. The abbreviated routing table saves memory space and CPU. When communicating in non-storing mode, a node always forwards its packet to the border router, which knows how to ultimately reach the final destination.

RPL is the most commonly used one. It is a distance vector protocol designed by IETF in 2012. CORPL is a non-standard extension of RPL that is designed for cognitive networks and utilizes the opportunistic forwarding to forward packets at each hop. On the other hand, CARP is the only distributed hop based routing protocol that is designed for IoT sensor network applications. CARP is used for underwater communication mostly. Since it is not standardized and just proposed in literature, it is not yet used in other IoT applications.

IoT Gateways

Gateway provides bridge between different communication technologies which means we can say that a Gateway acts as a medium to open up connection between cloud and controller (sensors / devices) in Internet of Things (IoT). By the help of gateways it is possible to establish device to device or device to cloud communication. A gateway can be a typical hardware device or software program. It enables a connection between sensor network and Internet along with enabling IoT communication, it also performs many other tasks such as this IoT gateway performs protocol translation, aggregating all data, local processing and filtering of data before sending it to cloud, locally storing data and autonomously controlling devices based on some inputted data, providing additional device security.



Key functionalities of IoT Gateway:

- Establishing communication bridge
- Provides additional security.

- Performs data aggregation.
- Preprocessing and filtering of data.
- Provides local storage as a cache/ buffer.
- Ability to manage entire device.
- Device diagnostics.
- Verifying protocols.

Working of IoT Gateway:

1. Receives data from sensor network.
2. Performs Preprocessing, filtering and cleaning on unfiltered data.
3. Transports into standard protocols for communication.
4. Sends data to cloud.

IoT Gateways are key element of IoT infrastructure as Gateways establish connection for communication and also performs other task.

Communication Protocols in IoT

IoT is based on networking of things where smart devices communicate with each other by sending and receiving data. So for that several network protocols (Communication protocols) are used to connect the IoT enabled devices and to establish communication.

1.13 COMMUNICATION PROTOCOLS

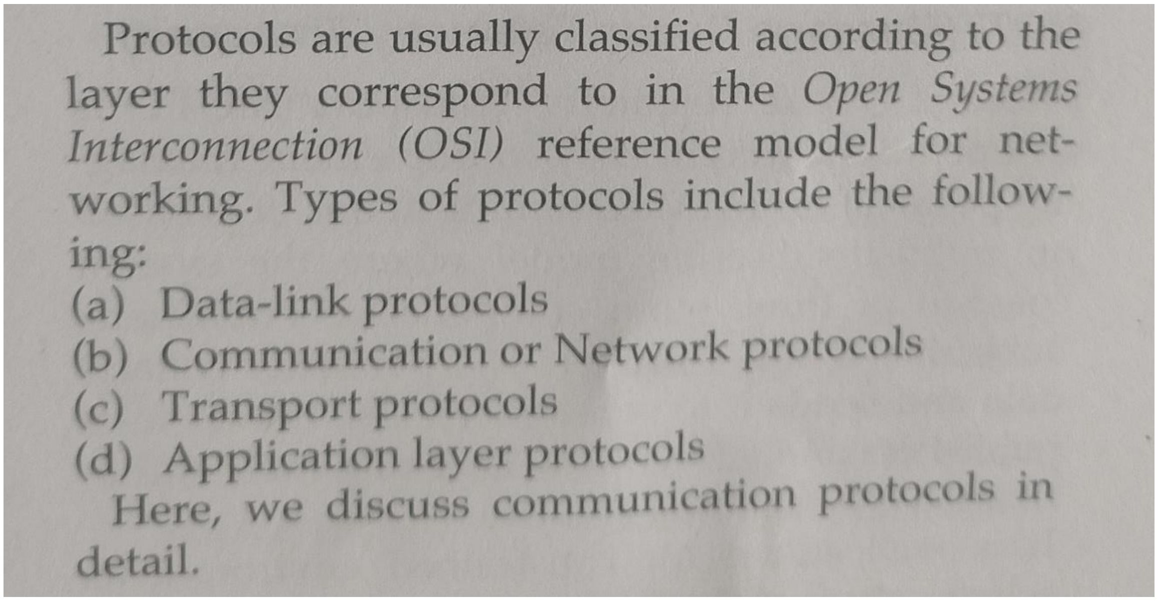
Communication protocols form the backbone of IoT systems and enable network connectivity. Communication protocols allow devices to exchange data over the network.

By the term protocol, we mean the set of rules or standards designed to enable computers to connect with one another and to exchange information. Protocol can describe low-level details of machine-to-machine interfaces (e.g. the order in which bits and bytes are sent across a wire) or high-level exchange between allocation programs (e.g. the way in which two programs transfer a file across the Internet).

An example will be able to make the idea clear. Suppose, there are two people one speaks Hindi and the other French. Since they have no common language, they each engage a translator whose common language is English.

Protocols can include rules concerning any or all of the following functions:

- Data transmission mechanisms.
- Communication session initialization and termination.
- Addressing and routing.
- Authentication and verification.
- Encryption and compression.
- Error correction.



Protocols are usually classified according to the layer they correspond to in the *Open Systems Interconnection (OSI)* reference model for networking. Types of protocols include the following:

(a) Data-link protocols
(b) Communication or Network protocols
(c) Transport protocols
(d) Application layer protocols

Here, we discuss communication protocols in detail.

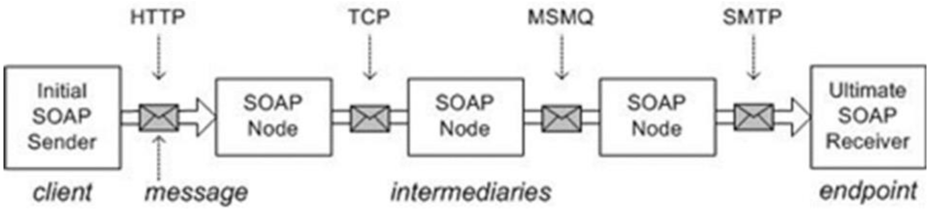
Simple Object Access Protocol (SOAP)

- SOAP is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment.
- SOAP uses XML technologies to define an extensible messaging framework, which provides a message construct that can be exchanged over a variety of underlying protocols.
- SOAP is a way for a program running in one operating system to communicate with a program running in either the same or a different operating system, using HTTP.

SOAP Messaging Framework

- XML-based messaging framework that is
    - 1) extensible: SOAP defines a communication framework that allows for features such as security, routing, and reliability to be added later as layered extensions

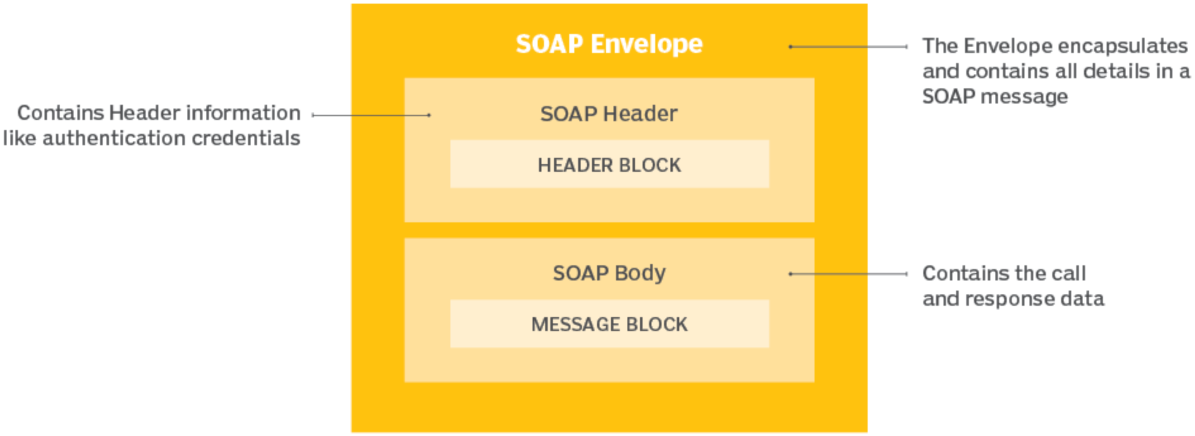
- 2) interoperable: SOAP can be used over any transport protocol such as TCP, HTTP, SMTP



- 3) independent: SOAP defines a model for processing individual, one-way messages. SOAP also allows for any number of message exchange patterns

SOAP Message Format

- SOAP message consists of three parts:
  - SOAP Envelope: A SOAP message can be defined as an XML document containing header and body encapsulated in the envelope.
  - SOAP Header (optional): This part is not mandatory. But when it is present it can provide crucial information about the applications.
  - SOAP Body: It contains the actual message that is being transmitted.

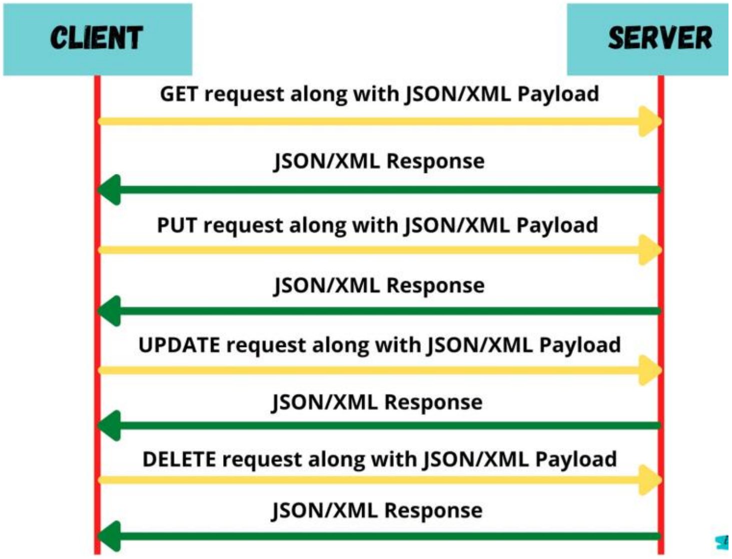

SOAP Envelope
The Envelope encapsulates and contains all details in a SOAP message
Contains Header information like authentication credentials
SOAP Header
HEADER BLOCK
SOAP Body
Contains the call and response data
MESSAGE BLOCK

Advantages of SOAP

- SOAP is a light weight data interchange protocol because it is based on XML.
- SOAP was designed to be OS and Platform independent.
- It is built on top of HTTP which is installed in most systems.
- It is suggested by W3 consortium which is like a governing body for the Web.
- SOAP is mainly used for Web Services and Application Programming Interfaces.

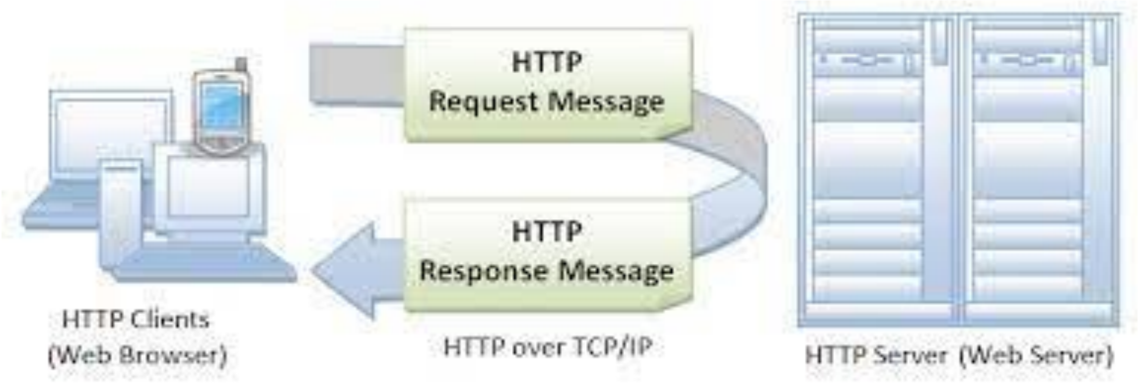
REST

- REST stands for Representational State Transfer. It’s an architectural style for developing web services. A lot of people believe that there is a REST protocol in IoT. However, REST itself is a concept, not an IoT protocol.
- REST is the basis for the most widely used form of API and is designed to be used over any protocol.


CLIENT
SERVER
GET request along with JSON/XML Payload
JSON/XML Response
PUT request along with JSON/XML Payload
JSON/XML Response
UPDATE request along with JSON/XML Payload
JSON/XML Response
DELETE request along with JSON/XML Payload
JSON/XML Response

HyperText Transfer Protocol (HTTP)

HTTP (Hypertext Transfer Protocol) is perhaps the most popular application protocol used in the Internet (or The WEB). HTTP is an asymmetric request-response client-server protocol as illustrated. An HTTP client sends a request message to an HTTP server. The server, in turn, returns a response message. In other words, HTTP is a pull protocol, the client pulls information from the server (instead of server pushes information down to the client). HTTP is a stateless protocol. In other words, the current request does not know what has been done in the previous requests. HTTP permits negotiating of data type and representation, so as to allow systems to be built independently of the data being transferred. The HTTP is an application-level protocol for distributed, collaborative, hypermedia information systems. It is a generic, stateless, protocol which can be used for many tasks beyond its use for hypertext, such as name servers and distributed object management systems, through extension of its request methods, error codes and headers.



To access the web pages, a client normally uses web browsers, who are responsible for initiating, maintaining, and closing TCP connections. HTTP is a stateless protocol, which means the Server maintains no information about earlier requests by clients.

• This is a protocol used mainly to access data on the World Wide Web (www).

• The Hypertext Transfer Protocol (HTTP) the Web's main application-layer protocol although current browsers can access other types of servers

• A repository of information spread all over the world and linked together.

• The HTIP protocol transfer data in the form of plain text, hyper text, audio, video and so on.

• HTTP utilizes TCP connections to send client requests and server replies.

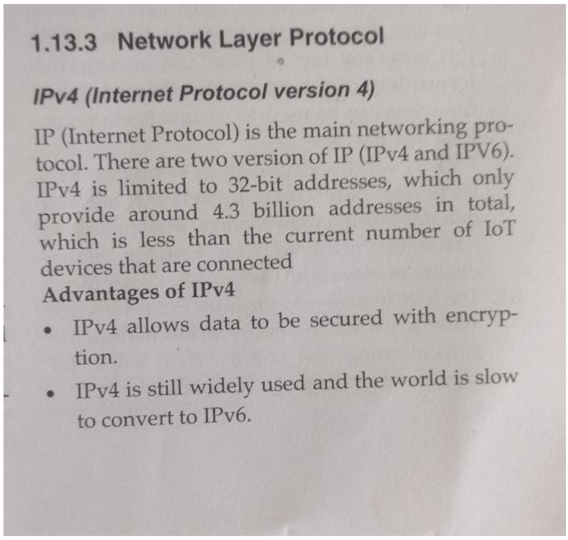
• it is a synchronous protocol which works by making both persistent and non-persistent connections.

The set of common methods for HTTP is defined below and this set can be expanded based on requirements. These method names are case sensitive and they must be used in uppercase.

S.N.	Method and Description
1	**GET** The GET method is used to retrieve information from the given server using a given URI. Requests using GET should only retrieve data and should have no other effect on the data.
2	**HEAD** Same as GET but transfers the status line and header section only.
3	**POST**

	A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.
4	**PUT** Replaces all current representations of the target resource with the uploaded content.
5	**DELETE** Removes all current representations of the target resource given by a URI.
6	**CONNECT** Establishes a tunnel to the server identified by a given URI.
7	**OPTIONS** Describes the communication options for the target resource.
8	**TRACE** Performs a message loop-back test along the path to the target resource.

IP addressing of IoT (Network layer protocol)



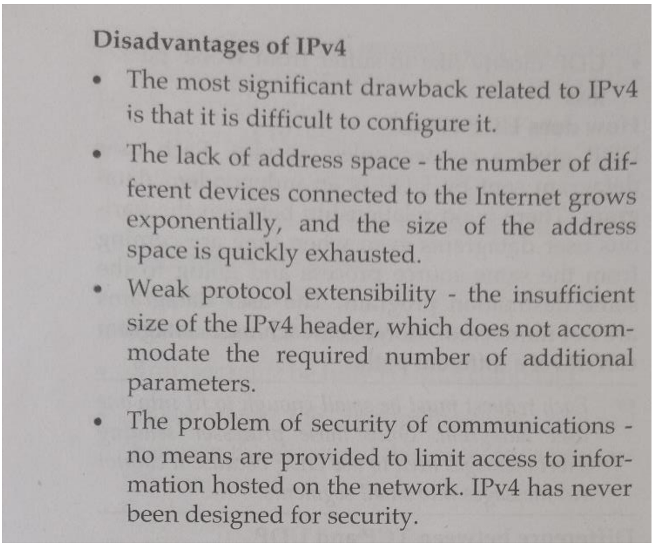
1.13.3 Network Layer Protocol

IPv4 (Internet Protocol version 4)

IP (Internet Protocol) is the main networking protocol. There are two version of IP (IPv4 and IPV6). IPv4 is limited to 32-bit addresses, which only provide around 4.3 billion addresses in total, which is less than the current number of IoT devices that are connected

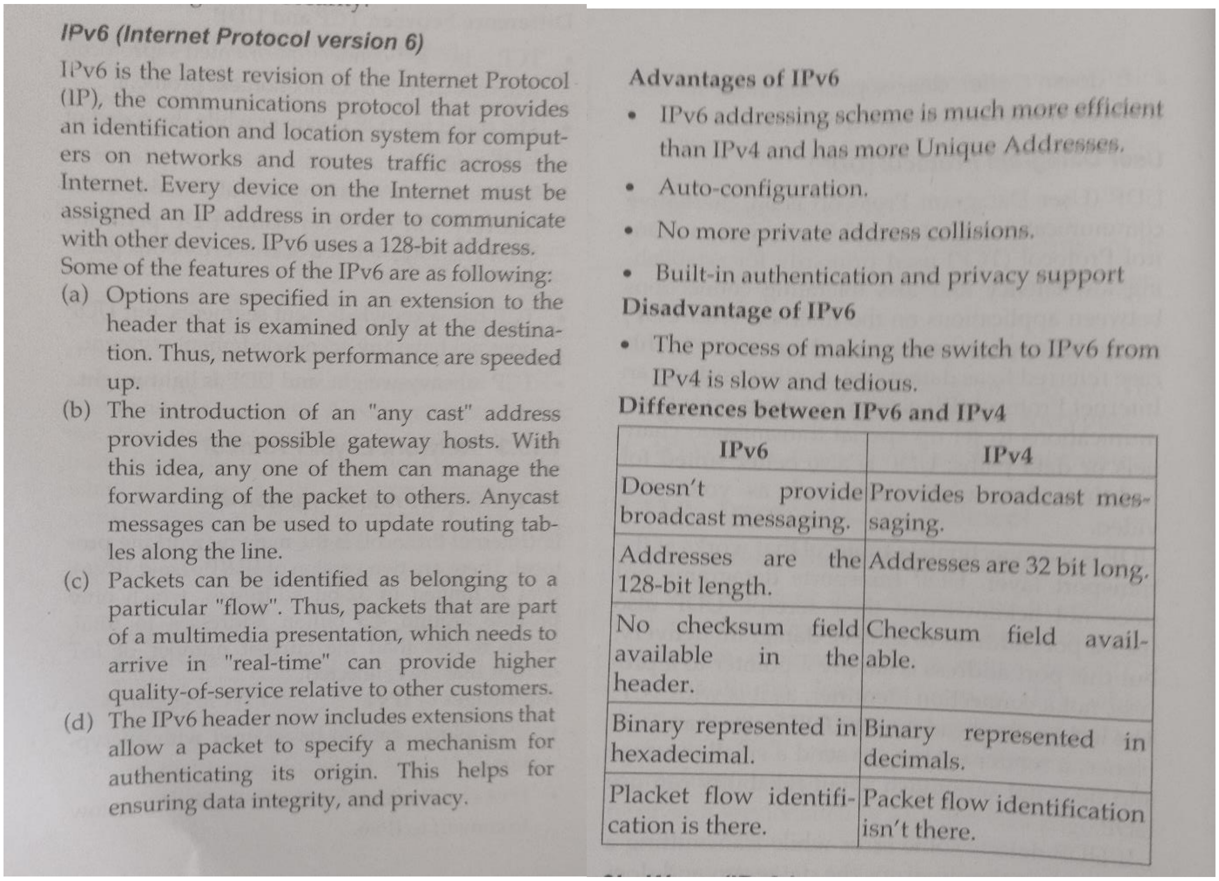
Advantages of IPv4

- IPv4 allows data to be secured with encryption.
- IPv4 is still widely used and the world is slow to convert to IPv6.



Disadvantages of IPv4

- The most significant drawback related to IPv4 is that it is difficult to configure it.
- The lack of address space - the number of different devices connected to the Internet grows exponentially, and the size of the address space is quickly exhausted.
- Weak protocol extensibility - the insufficient size of the IPv4 header, which does not accommodate the required number of additional parameters.
- The problem of security of communications - no means are provided to limit access to information hosted on the network. IPv4 has never been designed for security.



IPv6 (Internet Protocol version 6)

IPv6 is the latest revision of the Internet Protocol (IP), the communications protocol that provides an identification and location system for computers on networks and routes traffic across the Internet. Every device on the Internet must be assigned an IP address in order to communicate with other devices. IPv6 uses a 128-bit address.

Some of the features of the IPv6 are as following:

(a) Options are specified in an extension to the header that is examined only at the destination. Thus, network performance are speeded up.

(b) The introduction of an "any cast" address provides the possible gateway hosts. With this idea, any one of them can manage the forwarding of the packet to others. Anycast messages can be used to update routing tables along the line.

(c) Packets can be identified as belonging to a particular "flow". Thus, packets that are part of a multimedia presentation, which needs to arrive in "real-time" can provide higher quality-of-service relative to other customers.

(d) The IPv6 header now includes extensions that allow a packet to specify a mechanism for authenticating its origin. This helps for ensuring data integrity, and privacy.

Advantages of IPv6

- IPv6 addressing scheme is much more efficient than IPv4 and has more Unique Addresses.
- Auto-configuration.
- No more private address collisions.
- Built-in authentication and privacy support

Disadvantage of IPv6

- The process of making the switch to IPv6 from IPv4 is slow and tedious.

Differences between IPv6 and IPv4

IPv6	IPv4
Doesn't provide broadcast messaging.	Provides broadcast messaging.
Addresses are the 128-bit length.	Addresses are 32 bit long.
No checksum field available in the header.	Checksum field available.
Binary represented in hexadecimal.	Binary represented in decimals.
Placket flow identification is there.	Packet flow identification isn't there.

Medium Access Control

- In most networks, multiple nodes share a communication medium for transmitting their data packets
- The medium access control (MAC) protocol is primarily responsible for regulating access to the shared medium
- The choice of MAC protocol has a direct bearing on the reliability and efficiency of network transmissions λ due to errors and interferences in wireless communications and other challenges
- Energy efficiency also affects the design of the MAC protocol λ trade energy efficiency for increased latency or a reduction in throughput.
    - Responsibilities of MAC layer include:

- decide when a node accesses a shared medium
- resolve any potential conflicts between competing nodes
- correct communication errors occurring at the physical layer
- perform other activities such as framing, addressing, and flow control
- used in data link layer as divides data link layer into logical link control and medium access control layer)

When sending data to another device on the network, the MAC sublayer encapsulates higher-level frames into frames appropriate for the transmission medium, adds a frame check sequence to identify transmission errors, and then forwards the data to the physical layer as soon as the appropriate channel access method permits it. For topologies with a collision domain (bus, ring, mesh, point-to-multipoint topologies), controlling when data is sent and when to wait is necessary to avoid collisions. Additionally, the MAC is also responsible for compensating for collisions by initiating retransmission if a jam signal is detected. When receiving data from the physical layer, the MAC block ensures data integrity by verifying the sender's frame check sequences, and strips off the sender's preamble and padding before passing the data up to the higher layers.

Topics

definition IOT with a suitable example, IoT protocols and layers, Characteristics & functional block of IOT, communication models, M2M, difference between IoT and M2M, communication protocol in IOT, SOAP, REST, exclusive pair, WebSocket, MAC, WSN, HTTP,