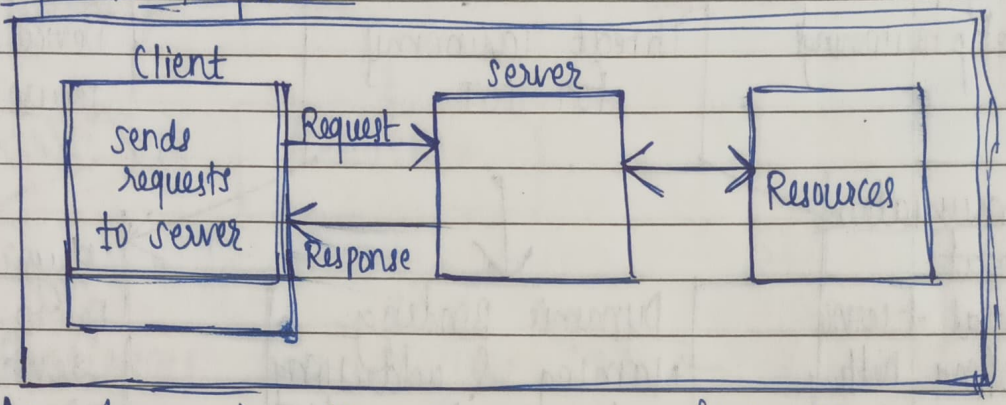
IOT Unit-3

Quick Work
Page No.:
Date:
M T W T F S S

Q. Communication model :-

→ Communication model tells how data moves from one IOT device to another.
There are 4 types of comm^n model in IOT.

1 Request Response -

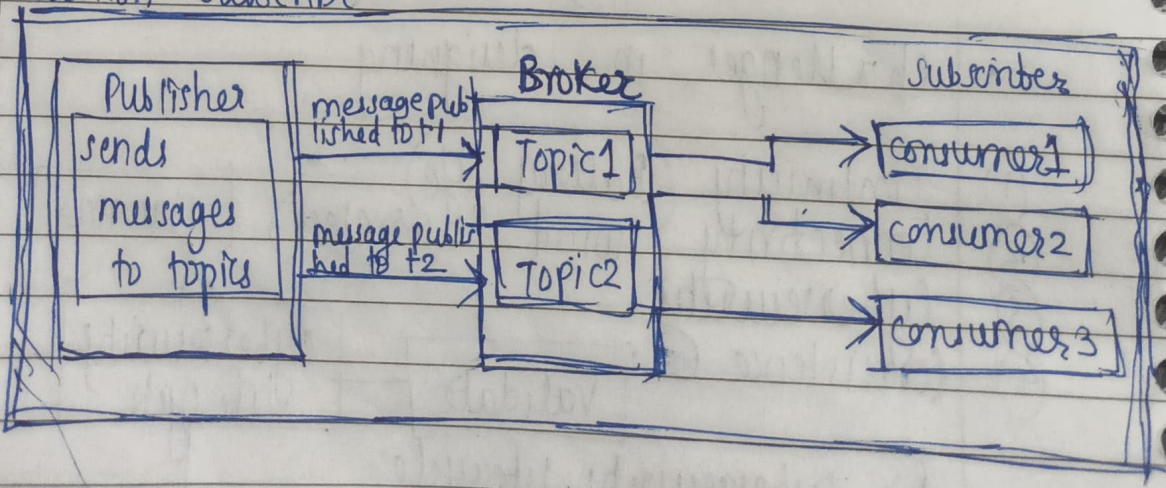


Working -
- As shown in the diagram, first the one device (client) sends a request
- Another device or server (resource) sends a response
- It basically works just like HTTP (web)

eg. temperature sensor sends a request
→ Server replies with latest temperature.

— Usage/ - synchronous (wait for reply)
adv - simple & reliable
- used when immediate data is needed.

2 Publish - Subscribe



Quick Work

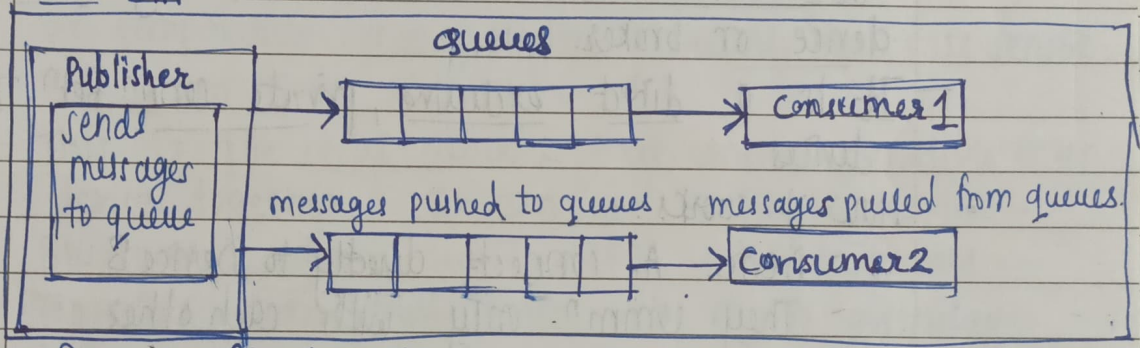
Page No.:
Date:
M T W T F S S

Working - Devices here do not communicate directly
- Publisher sends the data to the broker
- Subscribers receives the data if they have subscribed to that topic

eg - A smart home temperature sensor publishes data to the MQTT broker.
Mobile app subscribed to "temperature" receives updates

- Key Points/adv = Loosely coupled.
  - Scalable for many devices.
  - Uses MQTT in IoT
  - Best for real time updates.

3 Push - Pull -



Working - Producer (Push) It continuously puts data into a queue & the consumer (Pull) takes data from the queue whenever needed.
So, Push = sending data
Pull = receiving data (not at the same time)
- IOT sensor pushes data to a queue ~~or data buffer~~
- The queue temporarily stores the data.
- Consumer pulls the data when it is already to process.
- Both sides are independent

eg. Sensor pushes data to a message queue → Analytics system pulls data for processing.

KP/Adv - Handles big data streams, avoid overload. Asynchronous

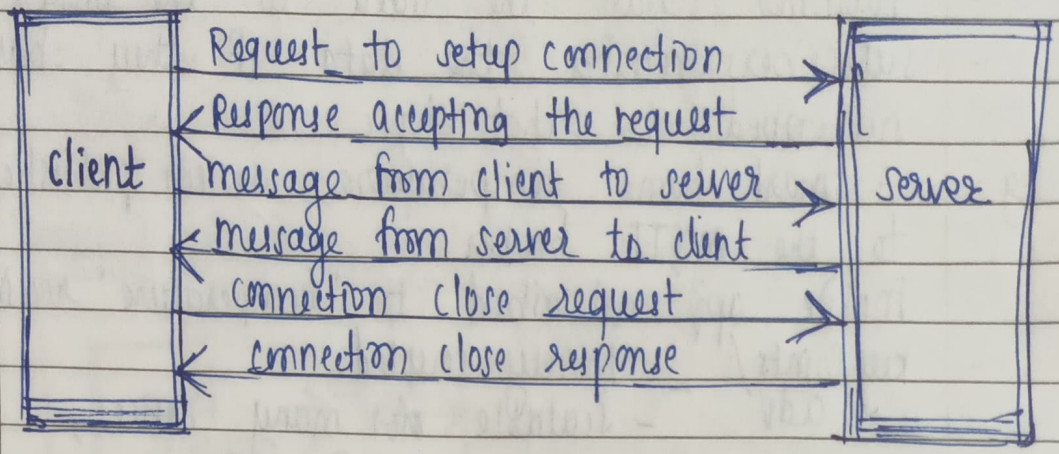
Quick Work

Page No.:

Date:

M T W T F S S

14 Exclusive Pair Model



- In this model, two IOT devices connect directly to each other and form a private, one to one communication link without any third device or broker.
- It is a direct, exclusive, private comm^n betn two devices
- How it works:-
  - Device A connects directly to Device B
  - They comm^n only with each other
  - No server, no broker, no queue
  - The link stays active until comm^n ends

eg. smartwatch paired with a phone using Bluetooth

Characteristics:- one to one direct comm^n.
- Private & secure link.
- Low latency (fast).

Technologies used:- Bluetooth.
- Zigbee direct links.
- Wifi direct.

Quick Work

Page No.:
Date:
M T W T F S S

• Pillars of IoT. -

1] M2M - Machine to Machine Communication

- M2M refers to direct communication between machines / devices without human involvement.
- In simple words, M2M allows devices to "talk to each other", share info, & perform action independently.
- M2M architecture consists of :-

i] M2M Devices -

- These are the sensors, actuators, meters, machines or embedded devices that collect data from the environment.

eg. temperature sensors, electricity meters, heart rate sensor.

ii] M2M Area Network -

- This is the local network that connects multiple M2M device together. It can use technologies like - Bluetooth, zigbee, wifi, NFC, wired networks
- Transfers data from devices → M2M gateway

iii] M2M gateway -

- This acts as a bridge betn device-level networks and communication networks.
- Collects data from devices, converts device protocols to Internet protocols & sends data to the M2M communication network.

eg. smart home hub, IOT gateway, router with IOT support

iv] M2M Commn Networks

- These are the wide-area networks that carry data from gateway to cloud/servers.
- Technologies used - 3G/4G/5G, LTE, Ethernet, Fiber

v] M2M Applications - These are the end-user appln that display processed data & allow monitoring or control

eg - smart home apps, smart city dashboards, Healthcare monitoring systems, Industrial automation platforms.

Quick Work

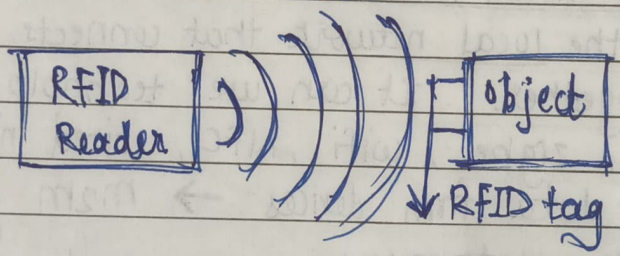
Page No.:

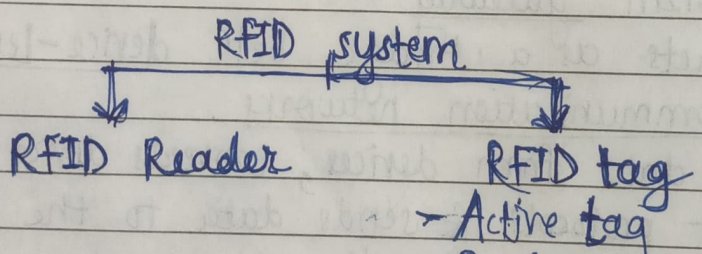
Date:

M T W T F S S

RFID - Radio Frequency Identification

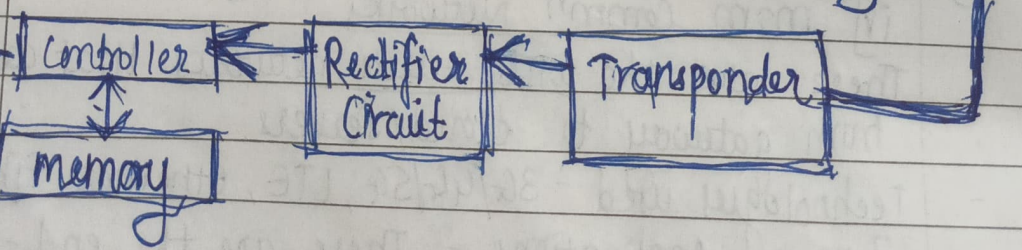
- RFID is a wireless identification technology that uses radio waves to identify, track and manage objects, people, or animals automatically.
- It uses RFID uses the radio frequency to read & capture info stored on a tag attached to the object.
- A tag can be read from up to several feet away & does not need to be within the direct line of an object
- It uses NCF & IC card.
- It consists of two main components -
  1. RFID Tag
  2. RFID Reader



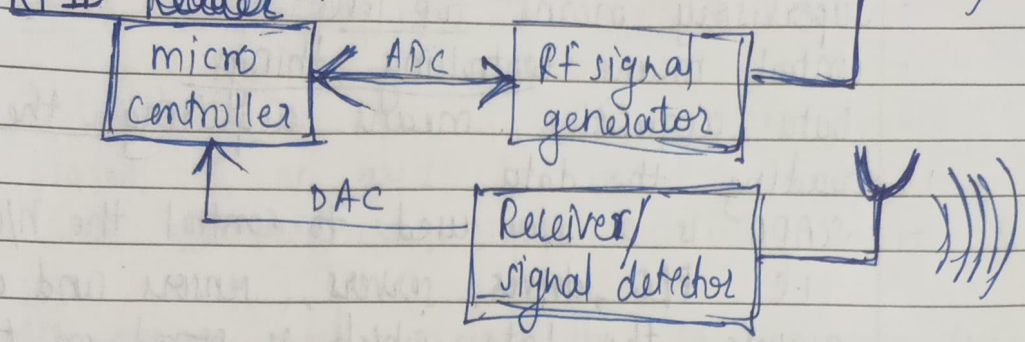


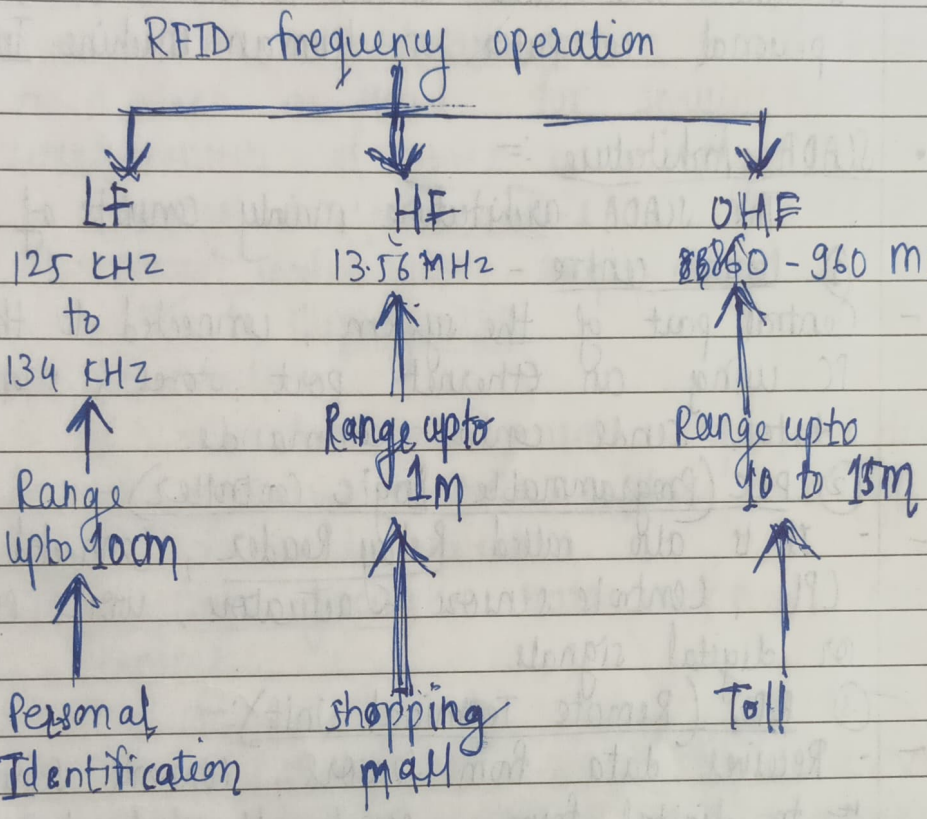
- Active tag
- Passive tag
- Semi Passive tag

RFID tags :-



- **RFID Reader -**





- **SCADA** - **Supervisory Control & Data Acquisition.**
  - SCADA is a system used to monitor & control equipment using computers, networked data communication, & graphical user interfaces.
  - It is widely used inside buildings or industrial plants
  - It uses BACnet communication protocol (LANBUS) & supports both wired & wireless commn
  - SCADA works at the supervisory level, above PLC (Programmable logic Controller) & RTU (Remote Terminal unit)

Applications - food processing industry, water treatment plants, chemical industries

Quick Work
Page No.:
Date:
M T W T F S S

- supervisory means top level.
- control means controlling things.
- Data acquisition means acquiring the data/ reading the data.
- SCADA is a s/w used to control the h/w i.e PLC, drives, servers, sensors and also acquire the data which is stored on the personal computer or Human Machine Interface (HMI)

• SCADA Architecture :=

The SCADA architecture mainly consists of -

① Control centre -

- Central part of the system, connected to the main PC using an Ethernet port, stores & monitors all data, sends control commands.

② PLC (Programmable Logic Controller) -

- It is also called Relay Reader, connected to the CPU, controls sensors & actuators, works on analog or digital signals.

③ RTU (Remote Terminal Unit) :-

- Receives data from sensors, converts sensor data to digital form, sends the data to scada system

④ Sensors & Actuators :-

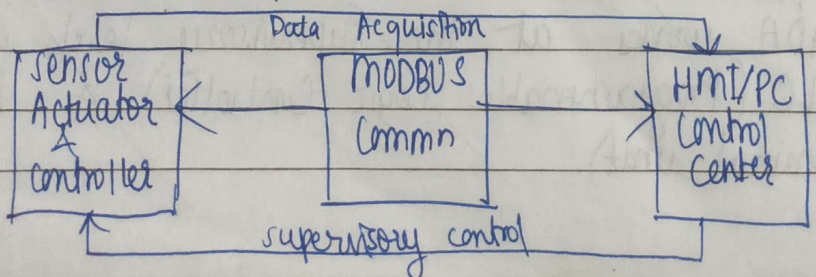
- measures temp, pressure, humidity, etc.

⑤ HMI (Human machine Interface) :-

- Graphical interface for operators, shows real-time data, alarms, & system status, allows operators to control process

⑥ Communication Network :-

- Modbus commn, connects PLC → Controller → HMI, can be wired wireless



Applications - Agriculture, smart cities, Healthcare, Home Automation

Adv - wireless, low cost, No human needed Disadv - Battery drains, limited range, can get hacked.

Quick Work
Page No.:
Date:
M T W T F S S.

- **WSN :- Wireless Sensor Network.**

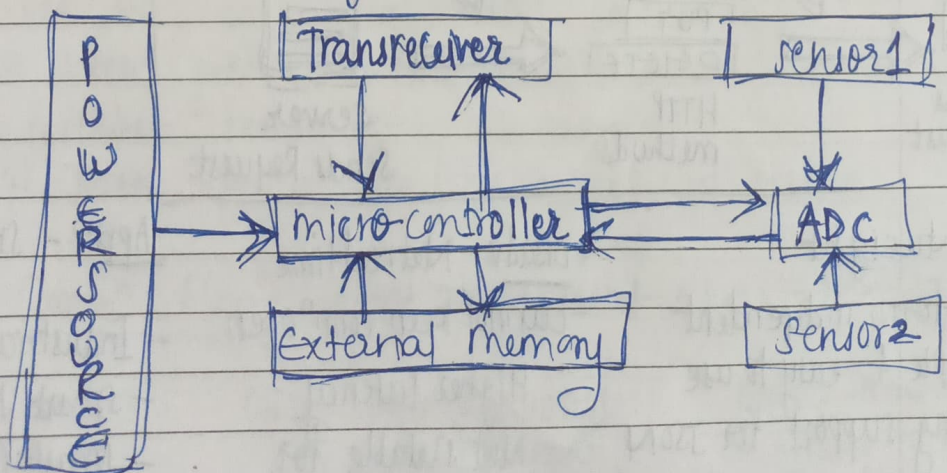
= A WSN is a n/w of many small wireless sensors placed in an area to collect information like temperature, light, humidity, movement, etc.

- All sensors send their data to one main device called the sink/Base station, which then sends it to a server or cloud for analysis.
- WSN consists of three basic things.
  - sensors to send values
  - Commⁿ model using any protocol
  - API to display data

- WSN elements -
  1) Node - Autonomous sensor - equipped device
  2) Data gatherers - Data capture and gateway to external systems.
  3) External systems - Data storing & managing centres.

- Working of WSN -
  - The sensor (nodes) are sensing the device values
  - These transmit the info to the measuring device (data gatherer) which transmit the values to external systems using Ethernet, Wifi or GPRS

- Parts of WSN -



Quick Work
Page No.:
Date:
M T W T F S S

API → Application Programming Interface

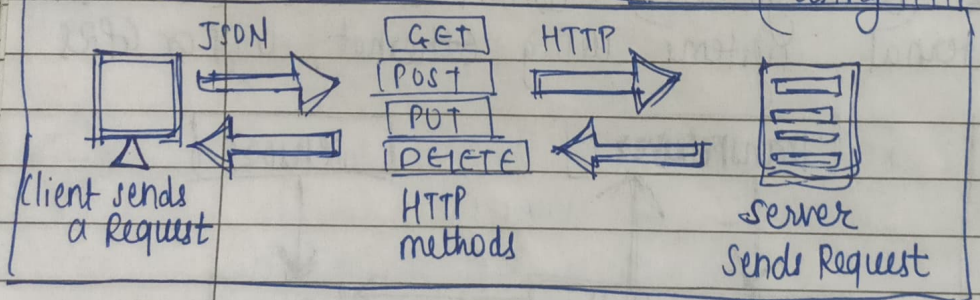
API is set of rules that allow two s/w appln or devices to commn with each other

- **REST API** :-
  - Representational state Transfer.
  - REST is a request-response communication method used in IOT.
  - Devices communicate using HTTP methods like -
    - GET → Read data
    - POST → Send/store data
    - PUT → Update data
    - DELETE → Remove data
  - REST is stateless, simple and best suited for IOT systems that send data at intervals (not continuously).

Example:- Smart Home Temperature Monitoring System.

Working →
1. A temperature sensor sends data to the IOT cloud.
2. The cloud exposes a REST API like -
   - GET /temperature
   - POST /temperature
3. mobile app or web app uses these APIs to read or update temperature settings.

REST Communication Flow - Sensor → IOT Cloud → Mobile App (using HTTP requests)



Adv - lightweight
- Platform independent
- simple & easy to use
- Strong support for JSON

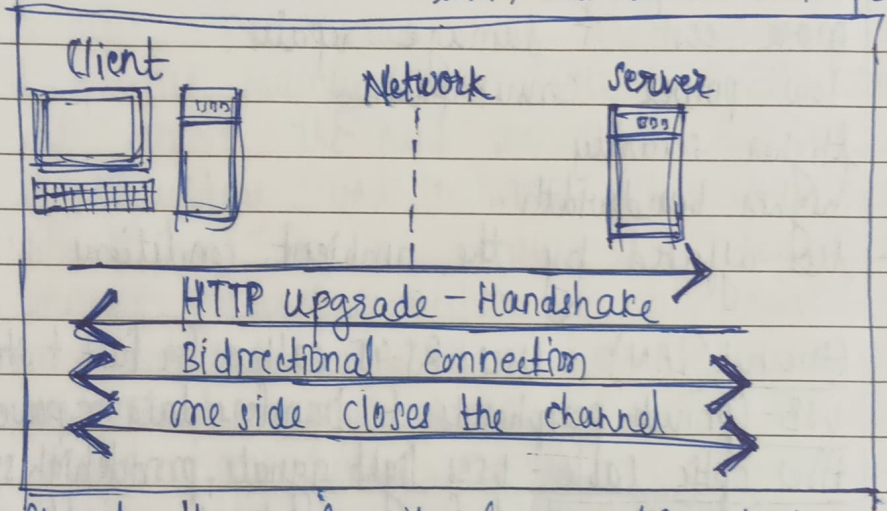
Disadv - Not real time
- Cannot keep conn^n open
- Higher latency
- Not suitable for continuous streaming sensor data

Appln - Smart Home system
- Industrial IOT
- Smart Agriculture
- Smart City management.

Quick Work

Page No.:
Date:
M T W T F S S

- **Websocket API** :-
  - Websocket provides full-duplex, two way continuous commⁿ between IOT device and server.
  - Unlike REST, Websocket keeps the connection always open.
  - Best for real-time IOT systems.
  - Once connected, both side can send and receive data anytime. without needing repeated requests.
  - Websocket is needed because IOT devices need - Real time data, low latency, Continuous connection, fast updates.
  - REST is slow for real time, but Websocket is perfect.



Example :- Smart Home Security Camera / Smart Door system.

Working - 1. Smart door sensor is connected to a websocket server
2. When door opens, sensor adds sends data instantly via the open websocket connection.
3. User gets real-time notification on mobile without sending any request.
4. Both device and server can talk any time.

Websocket Commⁿ Flow - Sensor ⇄ websocket server ⇄ mobile app.

Adv - Real time commⁿ
- full duplex
- low latency

Disadv - more complicated to implement
- Not ideal for small updates
- Requires continuous connection

Applⁿ - Smart Home monitoring
- Smart Factory
- Real time dashboards
- live weather stations

Quick Work
Page No.:
Date:
M T W T F S S

Connectivity Technologies in IOT :-

- IOT systems use different communication technologies to connect sensors, devices, gateways, and cloud platforms.
- These technologies are mainly classified into wired & wireless connectivity.

(1) Wired Connectivity -

- Wired connectivity means devices are connected using physical cables to communicate.

features -
- more expensive
- more cost of damage repairs.
- low power consumption
- Higher security
- higher bandwidth.
- less affected by the ambient conditions.

eg - Ethernet (LAN) - Uses RJ-45 cables for fast & stable data comm
USB - Connects peripherals & transfers data or power betn devices
Fiber optic cable - uses light signals, provide high speed comm
Coaxial cable - Used for TV networks
Serial cables - older methods for comm with printers & devices

(2) Wireless Connectivity -

- means devices communicate using electromagnetic waves (radio waves, infrared, satellite signals) without cables.

features -
- Higher reliability & longevity
- higher power consumption
- higher range
- higher bandwidth

Quick Work

Page No.:
Date:
M T W T F S S

eg. Wifi- wireless internet access through access points within limited range
Bluetooth - short range wireless commn between devices
Cellular N/w (4G/5G) - long range wireless commn using mobile power
Satellite commn - long distance commn via satellites
Infrared - line-of-sight data transfer, used in TV remotes.
NFC - very short range commn used in mobile payments

Q. explain any one in brief.

- Wifi -

- Wifi is a wireless communication technology that allows devices to access the internet using radio waves.
- It operates mainly in 2.4 GHz & 5 GHz frequency bands
- Wifi removes the need for cables & allow mobility
- It is widely used in homes, offices, and public hotspots
- Wifi is used in IoT because IoT devices need constant internet connection & wifi gives - good speed, easy installation, works in home/office environments, can connect many devices & that is why smart home devices mostly use wifi.

Adv - No cables required
fast internet
easy to install
Connect multiple IoT devices
Support mobility

Disadv - limited range
- can face interference
- less secure than wired
- more power usage

Applications of Wifi - Smart home devices
- Laptops & mobile internet
- Smart CCTV cameras
- Smart speakers
- Home routers
- Smart meters
- IoT appliances

Quick Work

Page No.:
Date:
M T W T F S S

IOT Design Methodology for smart Irrigation system :-

(1) Purpose & Requirements -

- Define purpose : automate Irrigation & save water
  Requirements : measure soil moisture, temp, humidity; automate pump control; remote monitoring

(2) Process model specification -

- Identify use cases -
  - sense moisture
  - Auto pump ON/OFF
  - sends alerts
  - manual control from app.

(3) Domain model specification -

- Define entities -
  Physical - soil, crops, pump, sensors
  Devices - moisture sensor, DHT11, ESP32/Arduino, Relay
  Services - Weather service, pump control, monitoring

(4) Info model specification -

- Define data structure -
  - moisture %, temp, humidity, pump status
  - Threshold values, timestamps
    Relationship - moisture < threshold → pump ON

(5) Service Specification -

- Services Include -
  - Monitoring service
  - Actuation service
  - Alert / notification service
  - Data Analytics service

Quick Work
Page No.:
Date:
M T W T F S S

⑥ IOT level specification
- Smart irrigation uses IoT level 3 with cloud based monitoring
Sensor → Gateway → Cloud → App

⑦ Functional view specification
Map functions of the system -
- Sensing - moisture, temp / humidity
- Comm^n - wifi / LORa
- Control - relay + pump
- Appln - mobile / web dashboard
- Cloud processing - decision-making, storage

⑧ Operational view specification
Define operational aspects -
- Comm^n - wifi / LAN / LORaWAN
- Storage - cloud database
- Hosting - cloud / server
- Device options - ESP32, NodeMCU, moisture sensor, relay

⑨ Device & Component Integration -
All hardware is connected & integrated -
- Moisture sensor + DHT11 connected to ESP32
- Relay module connected to pump motor
- Power supply, wiring, calibration, testing done
- Cloud API integrated for weather prediction

⑩ Appln Development - Develop the appln
- Real time monitoring screen
- Auto/manual pump control
- Alerts & ~~comm~~ notifications
- Data visualization (graphs, log)

Unit 4 - IOT Protocols

Quick Work
Page No.:
Date:
M T W T F S S

- Issues with standardization of IOT protocols -
  - IOT consists of many heterogeneous devices, networks, and platforms.
  - Due to this diversity, several issues arise in standardizing IOT protocols.

① Heterogeneity of Devices:-
- IOT devices differ in h/w, os, commⁿ range, power and processing capability.
- This makes it difficult to create one common protocol suitable for all devices

② Lack of Universal standards:-
- many organizations (IEEE, IETF, ITU, OneM2M, ETSI) define their own standards
- multiple standards cause confusion & interoperability problems.

③ Interoperability challenges :-
- Devices from different vendors often cannot communicate easily because they follow different protocols (MQTT, COAP, Zigbee, Bluetooth, LORa)
- Achieving seamless commⁿ becomes difficult.

④ Security & Privacy Issues -
- Different IOT protocols offer diff security levels
- lack of unified security guidelines increases risks like data theft, spoofing & unauthorized access

⑤ Scalability Problems
- Protocols may not scale uniformly when the numbes of IOT devices increases
- some protocols work well for small networks but not for large deployments

Quick Work

Page No.:
Date:
M T W T F S S

(6) Power & Resource Constraints :-
- IoT devices often have limited battery & memory
- standard protocols may be too heavy for low power devices, creating compatibility issues.

(7) Rapid Technological Changes :-
- IoT technology evolves very fast.
- Protocols quickly become outdated & cannot keep up, making long-term standardization difficult.

(8) Fragmented Ecosystem -
- manufacturers create proprietary protocols for their products, increasing fragmentation & reducing standard adoption.

• LoRa protocol - (long-range)
- LoRa is a wireless technology that helps devices talk to each other over very long distance using very little battery.

• Use of LoRa protocol in Smart Irrigation System Development

- In smart irrigation system, LoRa plays a major role in connecting sensors spread across the farm to a central gateway.

• Steps :-

(1) Sensors measure farm data :-
- soil moisture sensor, temp sensor, humidity sensor are placed in diff parts of the field

Quick Work
Page No.:
Date:
M T W T F S S

② Sensors send data using LoRa -
- Each sensor has a small LoRa module.
- It sends the data wirelessly for many kilometers.

③ LoRa Gateway receives the data -
- One gateway is placed at the farmer's house or near the farm
- It collects data from all sensors

④ Gateway sends data to the mobile App -
- The gateway forwards data to the cloud → which the farmer can see on a mobile app

⑤ Pump is controlled automatically
- If soil moisture is low → Pump turns ON
- If soil moisture is enough → Pump turns OFF

- Farmers get alerts on the phone.

- LoRa is used in smart Irrigation because -
- Very long range → sensors can send data from far away in the field
- Very low battery use → sensors can work for months or years
- Cheap → No sim card or wifi needed
- Perfect for farms → Works even where n/w is weak.

- LORa is used in smart irrigation to send sensor data over long distance using very low power, so the pump can be controlled automatically and farmers can monitor their fields easily.

Quick Work

Page No.:

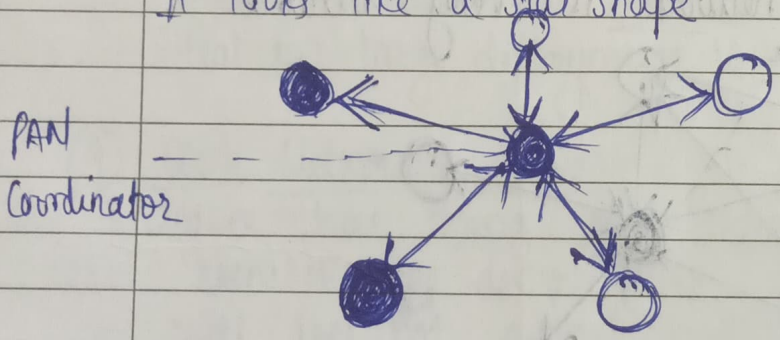
Date:

M T W T F S S

- **IEEE 802.15.4** :- Is a wireless comm^n standard specially designed for low power, low cost short range IoT devices
  - It is mainly used in - ZigBee, 6LoWPAN, Thread, wireless Sensor network
  - IEEE 802.15.4 supports three types of n/w topologies

(1) **Star Topology** -

- In star topology, there is one main central device called the PAN coordinator.
- All other devices (end devices) are connected only to this central device, not to each other
- Comm^n flow - All comm^n, even betn two devices, must pass through the PAN Coordinator
  - Device to PAN Coordinator - Single hop
  - Device to Device - Two hops
- It looks like a star shape



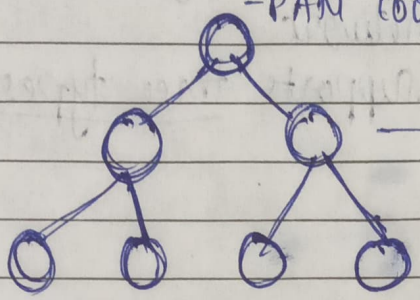
- Simple, low power, easy to manage

(2) **Tree Topology (Cluster-tree)** :-

- In Tree topology, devices are arranged in a hierarchical (tree like) structure
- At the top is the PAN coordinator
- Below it the coordinators, which manages small groups of devices

Quick Work
Page No.:
Date:
M T W T F S S

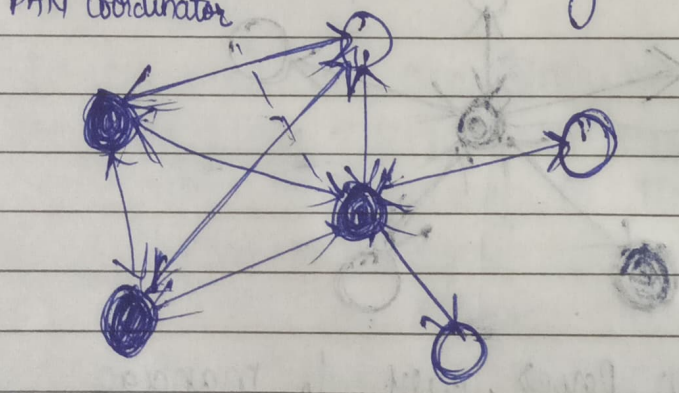
- At the bottom the end devices.
- It is used for large IOT networks because it can grow easily like branches of a tree.



- covers large area, easy to expand, organized comm^n.

③ Mesh Topology :-

- In mesh topology, every device can connect to multiple nearby devices.
- There is no central device like in star
- Data can travel through many paths, so the network is very reliable



- Highly reliable, self healing n/w, large coverage area
- Best for large & reliable IOT n/w
- If one path fails, data takes another route (multi-hop)

(6) Suitable for sensor data
- IOT sends small data
- Zigbee's low speed is enough & saves power

Quick Work
Page No.:
Date:
M T W T F S S

Q. Zigbee is more popular than wifi & Bluetooth in IOT.

(1) Low Power Consumption (Main Reason) -
- Zigbee uses very less power, so sensors can run on a small battery for months or years
- Wifi consumes high power
- Bluetooth consumes more power than Zigbee

→ So, IOT = battery devices → Zigbee is ideal.

(2) Supports large Number of devices -
- Zigbee can connect 65,000+ devices using mesh topology
- Wifi supports only 20-30 devices effectively
- Bluetooth supports 7-8 devices

→ for smart home or agriculture → Zigbee is better

(3) lower cost Hardware
- Zigbee modules are cheap compared to wifi & bluetooth
- Ideal for large deployments like farms, industries, sensors

(4) Mesh Networking support -
- Zigbee has strong mesh topology, so data reaches even if one device fails
- wifi does not support mesh by default
- Bluetooth mesh exists but is not as stable.

→ large IOT n/w need mesh, so zigbee is preferred

(5) Better for long Battery life IOT appln
- Zigbee is designed specially for - smart home devices, sensors, Industrial IOT, smart agriculture
- wifi & Bluetooth were designed for - Internet browsing, Audio browsing, file transfer.
- So, Zigbee matches IOT needs more closely

Quick Work

Page No.:

Date:

M T W T F S S

Classify betn m2m & SCADA Protocols.

	m2m	SCADA
Meaning	Commn between two machines without human involvement	System used to monitor & control industrial processes.
Purpose	Mainly for data exchange betn devices.	for real-time monitoring, control & automation
Area of use	Smart homes, IOT devices, appliances, meters	Industries, power plants, water treatment
Commn Type	Mostly device to device	Central system to field devices (RTUs, PLCs)
Protocol Eg.	MQTT, CoAP, HTTP	Modbus, DNP3, Profibus
Network type	Works on Internet, Cellular, Wifi	Uses industrial networks
Human Involvement	No human needed	Operator monitors data on SCADA screen
Real-time Control	Limited real-time control	Strong real-time control & automation
Data Volume	Small packets (sensor data)	Larger data + Commands + alarms
Appln	Smart meters, smart irrigation, wearable devices	Industrial automation, electricity grids, refineries

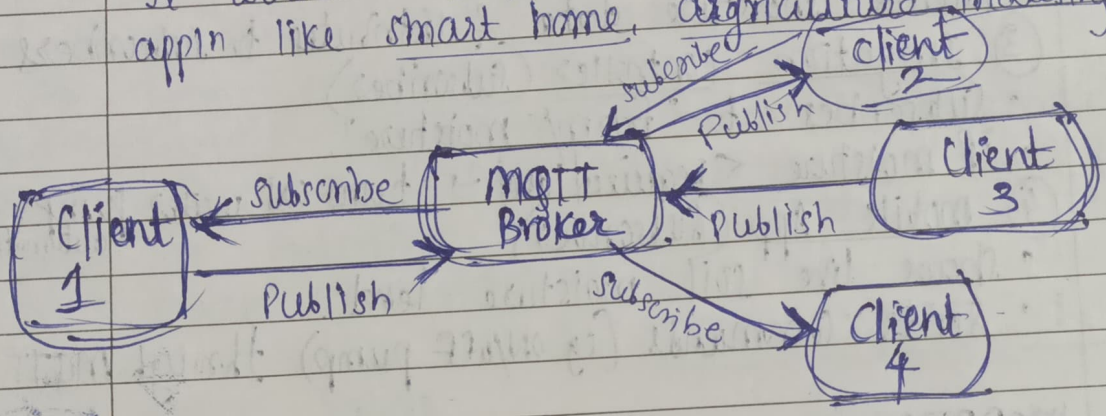
Quick Work

Page No.:
Date:

M T W T F S S

MQTT (Message Queuing Telemetry Transport)

- MQTT is a lightweight messaging protocol designed for sending data between devices using a publish subscribe model.
- It is specially designed for low power, low bandwidth, & unreliable networks.
- It works on TCP/IP & is very popular in IOT appln like smart home, agriculture, industry, etc.



- MQTT has three main components-

① Publisher -
- ~~A Device that~~ sends data.
- A device publishes data to a topic.

eg. soil moisture sensor → publishes to topic "farm/moisture"

② Broker
- Receives data.
- The MQTT Broker is the main server.
- It receives all messages from publishers.

eg. Mosquitto, HiveMQ.

③ Subscriber -
- Device or apps that need the data subscribe to the same topic.
- The broker forwards messages only to the subscribers.

- Features of MQTT - lightweight, low power usage, Publish Subscribe model, Reliable, scalable.

Quick Work
Page No.:
Date:
M T W T F S S

IOT Appln of MQTT - Smart Irrigation System -

- MQTT is widely used in smart irrigation because it handles sensor data efficiently.

Working :- eg

1. Soil Moisture Sensor (Publisher)
   - Measures soil moisture
   - Publishes data to topic "farm/moisture"
2. MQTT broker
   - Receives moisture data & distributes to subscribers.
3. Irrigation Controller (Subscriber)
   - Subscribes to "farm/moisture".
   - If moisture < required level → turns on water pump automatically
4. Mobile App (Subscriber)
   - Shows live soil moisture level.
   - Sends commands (eg ON/OFF pump) through MQTT.

- MODBUS -
- Modbus is a serial comm protocol used in industries to allow controllers, sensors, & machines to comm with each other
- It is simple, open-source, & widely used in Industrial IOT (IIOT) system.
- Originally developed for (PLCs)
- It uses a client/server or master/slave model & comes in two main versions:
  - Modbus RTU for serial comm
  - Modbus TCP for Ethernet.

- Uses RS-485 RS-232 most common in factories.
- Uses TCP/IP used in modern industrial n/w.

Working :- Client/Server Model - In a Modbus n/w, one device requests data from other devices.

- Messaging Structure - A Modbus message includes the slave device's address, the specific command or fun code, the data, & a checksum for error checking.

Master Device
PLC/RTU/HMI
↕ Req/Response
[Slave 1 (Sensor/RTU)] [Slave 2 (Meter/Drive)] [Slave 3 Actuator]

Quick Work
Page No.:
Date:
M T W T F S S

- Tranmission - The protocol can be transmitted over serial line or Ethernet networks

- Modbus usage in Industrial IOT :-

(1) Remote monitoring of Industrial machines :-
modbus is used to collect data like - temperature, pressure, vibration, RPM, current & voltage.
This data is sent to an IIOT gateway, then to the cloud for monitoring

(2) Energy monitoring systems :-
Most industrial energy meters comm^n using modbus
IIOT uses this data to track - Power usage, Load analysis, energy loss, predictive maintenance.

(3) Process Automation -
In factories, modbus connects - sensors, PLCs, SCADA, motor drives, valves.
IIOT uses this data to automate manufacturing & reduce manual work.

(4) Building Automation -
used for - HVAC control, Lighting, Chillers, Air Handling units
All monitored and controlled via Modbus + IOT dashboard

(5) Water & wastewater Treatment Plants
Modbus devices measure -
- flow
- level
- pump status
- chemical dosing.
- IIOT monitors these values and alerts if any fault occurs

IP based Protocols

Quick Work
Page No.:
Date:
M T W T F S S

IP based protocols such as MQTT, 6LOWPAN, & LORAWAN are widely used to develop diff IOT applⁿ. These protocols enable low power, long range, & reliable commⁿ over IP Networks.

① Smart Home (Using MQTT) - MQTT is used for home automation devices like smart lights, door sensors, alarms, & temp sensors.
- Device publish data to an MQTT broker & mobile apps subscribe to control or monitor them.

② Smart Agriculture / Irrigation (Using LORAWAN) - LORAWAN is used for long-range commn betn soil moisture sensors, weather sensors and irrigation pumps.
Data is sent from LoRa nodes → LORA gateway → Cloud using IP

③ Smart Metering (Using 6LOWPAN)
Electricity, gas, & water meters use 6LOWPAN to send readings over IPV6 N/w with very low energy consumption.
- Utility servers receive data in real-time

④ Industrial IOT (Using MQTT / 6LOWPAN)
Machines use MQTT for real time status updates & alerts.
Sensors on factory floors use 6LOWPAN for energy efficient IPV6 commn

⑤ Smart City Applⁿ (Using LORAWAN + MQTT)
- Streetlight monitoring, garbage bin monitoring, air quality measurement, & smart parking use LORAWAN for field data Collection and MQTT for cloud reporting.

⑥ Healthcare / Wearables (Using MQTT)
Wearable devices & send patient data like heart rate, temp, & emergency alerts to cloud servers using MQTT.

IoT Unit-06

- **Threat model** -
  - A threat model is a systematic method used to identify, analyze, and understand all possible security threats to an IoT system.
  - It helps us know - who can attack?
    What can be attacked?
    How can the attack happen?
    What damage it can cause?
    How to defend against it?

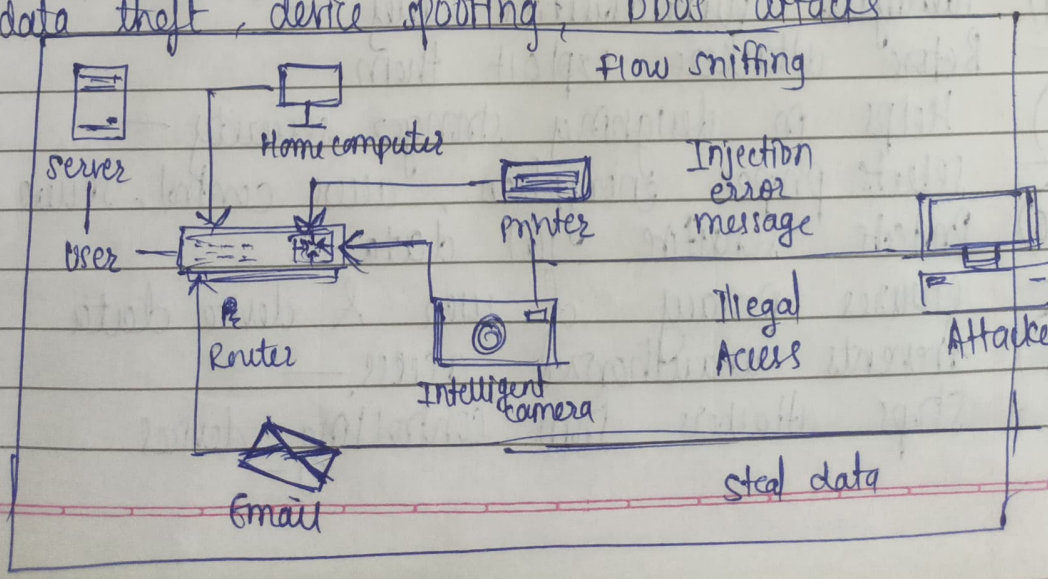
- Components of Threat model :-
  IoT threat modeling focuses on four main elements -

(a) **Assets** - Things to protect - sensors, smart appliances, user data, network traffic, cloud storage, control commands

(b) **Entry points** = Places where an attacker can enter - wifi/bluetooth, mobile app, Router, cloud API, physical ports, firmware updates.

(c) **Threat Actors** - People who can attack - Hackers, malware, insider users, fake devices, malicious third-party apps.

(d) **Possible Attacks** - sniffing, Illegal access, malware injection data theft, device spoofing, DDoS attacks



20-fi

- The IOT system contains devices like - Homecomputer, printer, router, smart camera, email server, cloud/servers
- An attacker can target any of these points -
- The diagram highlights four major threats -

① Flow sniffing - Attackers capture data flowing betn devices (eg- printer ⟷ home computer)
They use tools like packet sniffers to read sensitive info
Risk : Attack on confidentiality

② Injection of exact messages - Attackers send fake messages to the printer or IOT device..
Device thinks it is a normal command
Risk : Attack on Integrity

③ Illegal Access - Attackers gain unauthorized access to IOT devices like smart camera
They can watch video, control the device or change setting
Risk : Attack on authorization & privacy

④ Stealing Data - Attacker steal personal data from email comm, home n/w, camera storage, etc
Risk : Attack on confidentiality & privacy.

How Threat model helps in securing IOT appln.

a). Identifies vulnerabilities early —
- Before attackers exploit them

b) Helps in designing stronger security —
- selects proper encryption, access control, secure comm.

c) Protects sensitive IOT data —
- ensures privacy of user & device data

d) Prevents unauthorized access —
- stops attackers from controlling devices

e) Reduces risk of attacks
- By understanding all possible attack paths
f) Supports compilance
- Many IoT standards require threat modeling

• Classic pillars of Information Assurance while securing IoT applications -

- Information Assurance (IA) is the practice for protecting info by ensuring that it is secure, reliable & available when needed.
- In IoT, millions of small devices communicate with each other, so ensuring security becomes difficult
- To secure IoT, we follow the classic pillars of IA (also called CIA + 2 pillars).
- These principles work together to form a comprehensive security framework for IoT systems, which involve connected devices, data collⁿ, & commⁿ networks.

5 Pillars -

(1) Confidentiality - (Protecting data from Unauthorized Access)
- Confidentiality ensures that IoT data is accessed Only by authorized users

How it applies in IoT -
- Encrypt data collected by sensors.
- Use authentication methods (passwords, OTP)
- Prevents eavesdropping on IoT commⁿ (wifi, Bluetooth attacks)
eg - smart home camera feed is encrypted so outsiders cannot view the video

(2) Integrity (Ensuring data is not modified or tampered)
- Integrity makes sure data stays accurate & unchanged during transmission or storage

How it applies on IoT -
- Use message authentication codes (HMAC).
- Use digital signatures to verify device identity.
- Protect against man-in-the-middle & data injection attacks.

eg - In smart irrigation, moisture sensor readings should not be altered by an attacker.

(3) Availability - (Ensuring IoT services are working when needed)
Availability means IoT devices and services must remain accessible & work continuously.

How it applies in IoT -
- Use redundancy (backup servers)
- Prevent DoS/DDoS attacks on IoT gateways
- Regular updates & maintenance

eg - A smart health monitoring device must always send patient data to the doctor in real-time.

(4) Authentication (Verifying Identity of User or Device)
Authentication confirms "who is accessing the IoT system?"

How it applies in IoT -
- Device-to-device authentication
- multi-factor authentication for IoT apps
- Pairing procedures for Bluetooth devices

eg - Only the owner's smartphone should pair with the smart door lock.

(5) Non-Repudiation (Proof of actions cannot be Denied)
Non-Repudiation ensures that no user or device can deny performing an action.

How it applies in IoT -
- Maintain secure logs of device actions
- Digital signatures to validate actions

- Blockchain to maintain tamper-proof records
  e.g. If a smart meter reports electricity readings, the consumer cannot deny recorded usage

- Use security concepts to identify diff threats —
  - IoT appln comm using sensors, controllers, cloud platforms, mobile apps & networks
  - Due to wireless connectivity and limited device security, each IoT appln faces diff threats
  - Using basic security concepts (CIA + Authentication Authorization properties) the threat can be identified as follows.

① Smart Home Automation - Threats
i) Unauthorized access (Confidentality threat)
- Hackers may gain access to smart locks, cameras, or appliances due to weak passwords or insecure wifi
ii) Device manipulation (Integrity threat)
- Attacks can change thermostat settings, turns devices ON/off
iii) Botnet Attacks (Availability threat)
- Smart home devices can be hijacked & used in DDos attacks making them unavailable to the user

② Smart Parking system - Threats
i) false Parking Availability Data (Integrity Threat)
- Attackers can inject fake sensor data showing wrong slot availability
ii) Unauthorized charging/Billing manipulation (Integrity & Confidentality)
- Payment data can be altered or stolen during transmission
iii) system downtime (Availability)
- DDos attackers can shut down parking servers, causing chaos

③ Smart Irrigation System - Threats

i) Unauthorized control of water flow (Integrity Threat)
- Attackers can turn ON/OFF irrigation remotely causing crop damage

ii) Network Jamming (Availability Threat)
- Wireless comm^n can be jammed to stop irrigation commands

iii) Sensor data tampering (Integrity T.)
- Fake soil moisture data can mislead the system resulting in over watering or under-watering

④ Smart Surveillance System - Threats

i) Camera Hijacking (Confidentiality)
- Attackers gain access to live video feeds

ii) Video tampering (Integrity)
- Recorded video can be deleted, paused or modified.

iii) Denial of service Attack (Availability)
- Surveillance cameras or NVRs can be overloaded to stop recording

• Predict the possible vulnerabilities in designing smart home intrusion detection system.
- A smart home intrusion detection system (IDS) uses IOT devices such as motion sensors, CCTV cameras, door sensors, smart locks, alarms, wifi modules etc.
- While designing such an IDS, several vulnerabilities may occur due to limited device capability, weak security & wireless comm^n

① Weak Authentication & Passwords
- many smart home devices use default username/password, weak PINs or password protection
- Attackers can easily guess the credentials & gain control of sensors or cameras

(2) Unencrypted Comm^n
- If data betn the sensors → hub → cloud is not encrypted, attackers can intercept signals.
- They can view live camera feeds, disable alarms, or replay commands

(3) Vulnerable wireless protocols (wifi, Zigbee, BLE)
- Smart home IDs devices often use low power protocols.
- These protocols may be vulnerable to: jamming, spoofing, eavesdropping. This can disable intrusion system.

(4) Privacy leakage - Smart cameras & microphones may leak personal info if hacked. The attacker can watch the home activity patterns such as: when the house is empty, user routines, daily movements.

(5) Poor Physical security of devices - Sensors placed outside the house can be: removed, reset, damaged, reprogrammed physically. Thus IDS fails to detect intrusion.

(6) Firmware vulnerabilities
- many IoT sensors run outdated firmware. If patches are not applied, attackers can exploit known weakness & take over.

(7) Cloud Server Vulnerability -
- most smart IDS soln depends on cloud dashboards
- If the cloud server is attacked, the entire home IDS becomes unavailable

(8) Device Cloning & spoofing -
- An attacker may clone a legitimate sensor or send fake sensor alerts to confuse the system.
- Fake devices can register themselves into the home n/w.

- Different vulnerabilities of IOT & how to handle them
- IOT system consists of sensors, devices, networks & cloud platforms.
- Due to limited resources & wide connectivity, they face several vulnerabilities.
- The major IOT vulnerabilities & their soln are:-

(1) Weak or No Authentication :-
- Many IOT devices use default passwords, weak login systems or even no authentication, allowing attackers to easily access devices.

Handle - use strong authentication (OTP, biometrics, certificates)
- Enforce password changes & use MFA.

(2) Insecure Comm^n
- Data is often sent without encryption b/t devices, gateways, and cloud. Attackers can perform eavesdropping or man-in-the-middle attacks.

Handle - use end to end encryption
Secure comm^n protocols

(3) Poor software / firmware security
- IOT devices often run outdated firmware, making them vulnerable to exploits & malware.

Handle - Regular firmware updates & secure OTA updates
- Signed updates to avoid tampering.

(4) Lack of Physical security -
- IOT devices deployed in public can be stolen, reset or physically tampered

Handle - Tamper-resistant casing
Disable unused ports. Physical locks & secure mounting

(5) Cloud & API vulnerabilities -
- Weak APIs, poor access policies, or insecure cloud storage can leak sensitive data.

Handle - Strong API authentication. Role-based access control.
Secure cloud configurations

- Security Requirement of IOT appln-

- IOT systems connect many devices, sensors, n/w & cloud platforms
- To protect data & devices, the following security requirements must be fulfilled:

(1) Confidentiality - ensures data is not accessed by authorized user.
Technique - AES, TLS, VPN, encryption keys

(2) Integrity - ensures data is accurate, unchanged, trustworthy
Technique - Hashing, digital signature, MAC

(3) Availability - ensures IOT services remain usable at all times.
Technique - Redundancy, Backup power, load balancing

(4) Authentication - ensures only verified users/device can access IOT system
Technique - digital certificates, OTP, biometrics.

(5) Authorization / Access control - ensures each device/user gets only the required level of access.
Technique - Role based access control, Attribute based access control

(6) Non Repudiation - Ensures a user cannot deny sending a message or performing an action
Technique - Digital signatures, Blockchain logs.

(7) Data privacy - protects personal info of users.
Technique - Data minimization, privacy policies, anonymization

(8) Secure Comm^n - ensures safe data transfer betn sensors, gateways, cloud.
Technique - TLS/SSL, DTLS, encrypted CoAP/MQTT.

(9) Device security - Protects the IOT device itself from attacks.
Technique - Secure boot, firmware updates, patching anti-tampering

- Lightweight Cryptography -
- lightweight cryptography refers to special cryptographic algorithms designed for low power, low cost, resource-constrained IoT devices such as RFID tags, sensors, wearables, and embedded systems.
- These devices cannot run traditional cryptography (AES, RSA) because they have limited mem, battery, processing power and storage.
- lightweight Cryptography need -
IoT devices face : small CPU & RAM • Very low battery • Need fast encryption • Low n/w bandwidth
so lightweight algo provide security with minimum resource storage
- Charactoristics :-
i) small key size & block size - less mem needed
ii) low power consumption - suitable for battery powered IoT nodes
iii) low computational cost - works on microcontrollers with low clock speed
iv) smaller code size - can fit into tiny ROM/flash memory
v) fast encryption & decryption - Imp for real-time IoT appln
- examples -
1. PRESENT - lightweight block cipher used in RFID tags
2. SPECK & SIMON - designed for simple IoT h/w
3. HIGHT - for low power sensors
4. KATAN/KATANTAN - very low h/w footprint
- Applications in IoT -
- Smart home devices
- RFID based tracking
- Wearable health sensors
- Smart agriculture nodes
- Wireless sensor n/w
- Industrial IoT systems
- lightweight cryptography ensures that even small devices can maintain Confidentiality, Integrity, & Authentication

IOT Unit-6

- Challenges in Designing / Securing IOT applications -
- IOT systems connect many devices, sensors, networks, and cloud platforms.
- This make IOT powerful, but also difficult to secure.
- The major challenges are -

① Resource constraints - / limited device Resources -
- IOT devices have low CPU, small memory, & limited battery
- So strong algo cannot run easily
- This makes devices weak & easily attacked.

② Heterogeneous Devices -
- IOT uses different h/w, OS, protocols, & comm^n types
- Because of this standard security solution cannot fit all devices, which creates security gaps.

③ Large Attack Surface -
- Millions of devices are connected.
- more devices = more entry points for attackers.

④ Weak or NO Physical Security -
- many IOT devices are kept in open or outdoor areas
- Attackers can physically access, steal, or tamper with the device.

⑤ Insecure Comm^n -
- attackers can capture data because many IOT devices use weak encryption, default passwords. Unsecured wireless channels
- This leads to eavesdropping & data theft.

⑥ Poor software updates –
- Many IOT devices –
  - Do not receive updates, receive update late, have no update mechanism
- This leaves old vulnerabilities unpatched, causing attacks

⑦ Data Privacy Issues –
- IOT collects personal & sensitive data
- If not stored or transmitted securely → privacy leakage can occur

⑧ Cloud & N/w Dependencies –
- IOT relies heavily on cloud platforms & networks
- If cloud or n/w is attacked the entire IOT appln fails.

⑨ Weak Authentication & Default Passwords –
- Many IOT devices come with –
  - Default usernames, weak passwords, no multi-factor authentication
- Attackers easily guess passwords & take control