CMR
INSTITUTE OF
TECHNOLOGY

USN

Sub:	Internet of Things – IAT – 1 Answer Key	Code:22MCA32

	Answer Any 5 **QUESTIONS**	Marks
1)	**Define IOT. Explain the genesis of IoT**	10

IoT is a technology transition in which devices will allow us to sense and control the physical world by making objects smarter and connecting them through an intelligent network.

IoT is a technology transition in which devices will allow us to sense and control the physical world by making objects smarter and connecting them through an intelligent network.

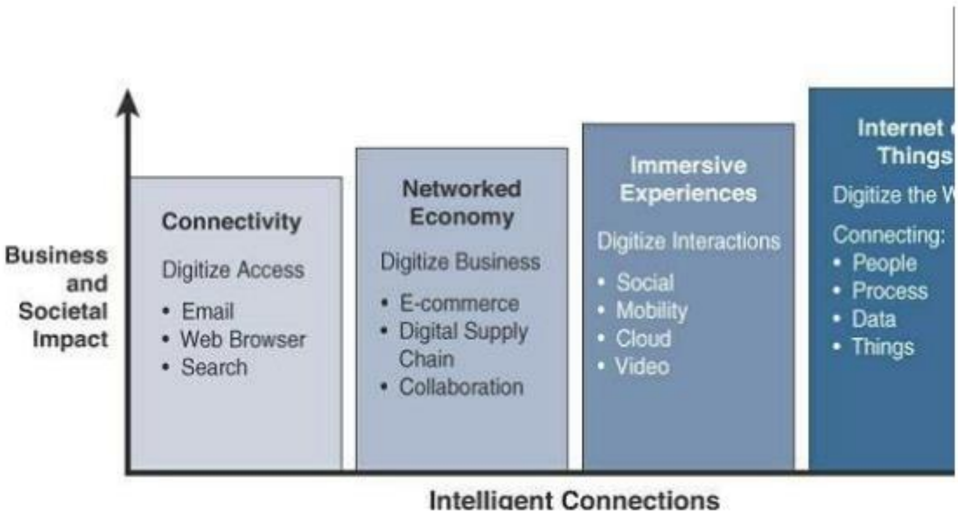
GOAL: The basic premise and goal of IoT is to "connect the unconnected." This means thatobjects that are not currently joined to a computer network, namely the Internet, will be connected so that they can communicate and interact with people and other objects.

When objects and machines can be sensed and controlled remotely across a network, a tighterintegration between the physical world and computers is enabled.

This allows for improvements in the areas of efficiency, accuracy, automation, and the enablement of advanced applications.

GENESIS OF IOT

The person credited with the creation of the term "Internet of Things" is Kevin Ashton. Whileworking for Procter & Gamble in 1999, Kevin used this phrase to explain a new idea related to linking the company's supply chain to the Internet.



the evolution of the Internet can be categorized into four phases. Each of these phases has had a profound impact on our society and our lives. These four phases are further defined inTable below.

Internet Phase	Definition
Connectivity (Digitize access)	This phase connected people to email, web services, and search so that information is easily accessed.
Networked Economy (Digitize business)	This phase enabled e-commerce and supply chain enhancements along with collaborative engagement to drive increased efficiency in business processes.
Immersive Experiences (Digitize interactions)	This phase extended the Internet experience to encompass widespread video and social media while always being connected through mobility. More and more applications are moved into the cloud.
Internet of Things (Digitize the world)	This phase is adding connectivity to objects and machines in the world around us to enable new services and experiences. It is connecting the unconnected.

2) **Explain in detail about the core functional stack of simplified architecture.** 10

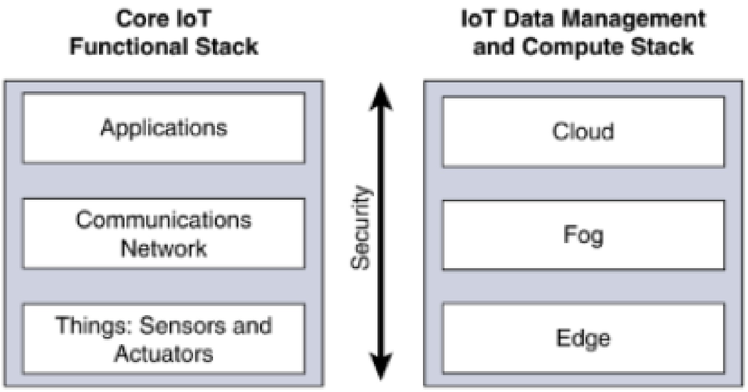


Figure 2-6 *Simplified IoT Architecture*

The Core IoT Functional Stack

IoT networks are built around the concept of "things," or smart objects performing functions and delivering new connected services. These objects are "smart" because they use a combination of contextual information and configured goals to perform actions. These actions can be self-contained (that is, the smart object does not rely on external systems for its actions); however, in most cases, the "thing" interacts with an external system to report information that the smart object collects, to exchange with other objects, or to interact with a management platform. In this case, the management platform can be used to process data collected from the smart object and also guide the behavior of the smart object. From an architectural standpoint, several components have to work together for an IoT network to be operational:

- **"Things" layer:** At this layer, the physical devices need to fit the constraints of the environment in which they are deployed while still being able to provide the information needed.
- **Communications network layer:** When smart objects are not self-contained, they need to communicate with an external system. In many cases, this communication uses a wireless technology. This layer has four sublayers:
  - **Access network sublayer:** The last mile of the IoT network is the access network. This is typically made up of wireless technologies such as 802.11ah, 802.15.4g, and LoRa. The sensors connected to the access network may also be wired.

- **Gateways and backhaul network sublayer:** A common communication system organizes multiple smart objects in a given area around a common gateway. The gateway communicates directly with the smart objects. The role of the gateway is to forward the collected information through a longer-range medium (called the backhaul) to a headend central station where the information is processed. This information exchange is a Layer 7 (application) function, which is the reason this object is called a gateway. On IP networks, this gateway also forwards packets from one IP network to another, and it therefore acts as a router.
- **Network transport sublayer:** For communication to be successful, network and transport layer protocols such as IP and UDP must be implemented to support the variety of devices to connect and media to use.
- **IoT network management sublayer:** Additional protocols must be in place to allow the headend applications to exchange data with the sensors. Examples include CoAP and MQTT.

- **Application and analytics layer:** At the upper layer, an application needs to process the collected data, not only to control the smart objects when necessary, but to make intelligent decision based on the information collected and, in turn, instruct the "things" or other systems to adapt to the analyzed conditions and change their behaviors or parameters.

3) **Explain with the neat diagram of one M2M IOT standardized architecture.** 10

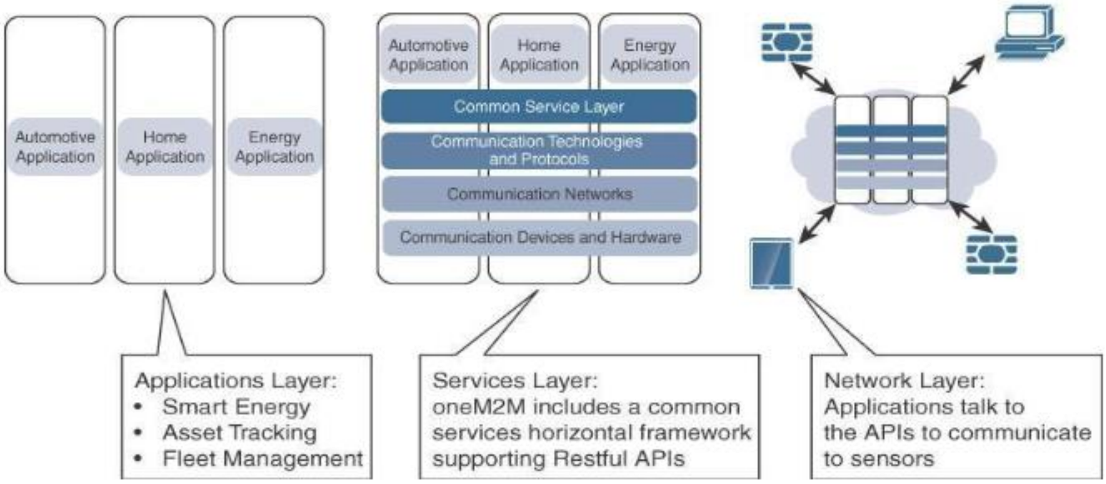
In an effort to standardize the rapidly growing field of machine-to-machine (M2M) communications, the European Telecommunications Standards Institute (ETSI) created the M2M Technical Committee in 2008. The goal of this committee was to create a common architecture that would help accelerate the adoption of M2M applications and devices. Over time, the scope has expanded to include the Internet of Things.

One of the greatest challenges in designing an IoT architecture is dealing with the heterogeneity of devices, software, and access methods. By developing a horizontal platform architecture, oneM2M is developing standards that allow interoperability at all levels of the IoT stack

- **Applications layer:** The oneM2M architecture gives major attention to connectivity between devices and their applications. This domain includes the application-layer protocols and attempts to standardize northbound API definitions for interaction with business intelligence (BI) systems. Applications tend to be industry-specific and have their own sets of data models, and thus they are shown as vertical entities.

- **Services layer:** This layer is shown as a horizontal framework across the vertical industry applications. At this layer, horizontal modules include the physical network that the IoT applications run on, the underlying management protocols, and the hardware. Examples include backhaul communications via cellular, MPLS networks, VPNs, and so on. Riding on top is the common services layer.

- **Network layer:** This is the communication domain for the IoT devices and endpoints. It includes the devices themselves and the communications network that links them. Embodiments of this communications infrastructure include wireless mesh technologies, such as IEEE 802.15.4, and wireless point-to-multipoint systems, such as IEEE 801.11ah.



4) **Explain the challenges of IoT.**

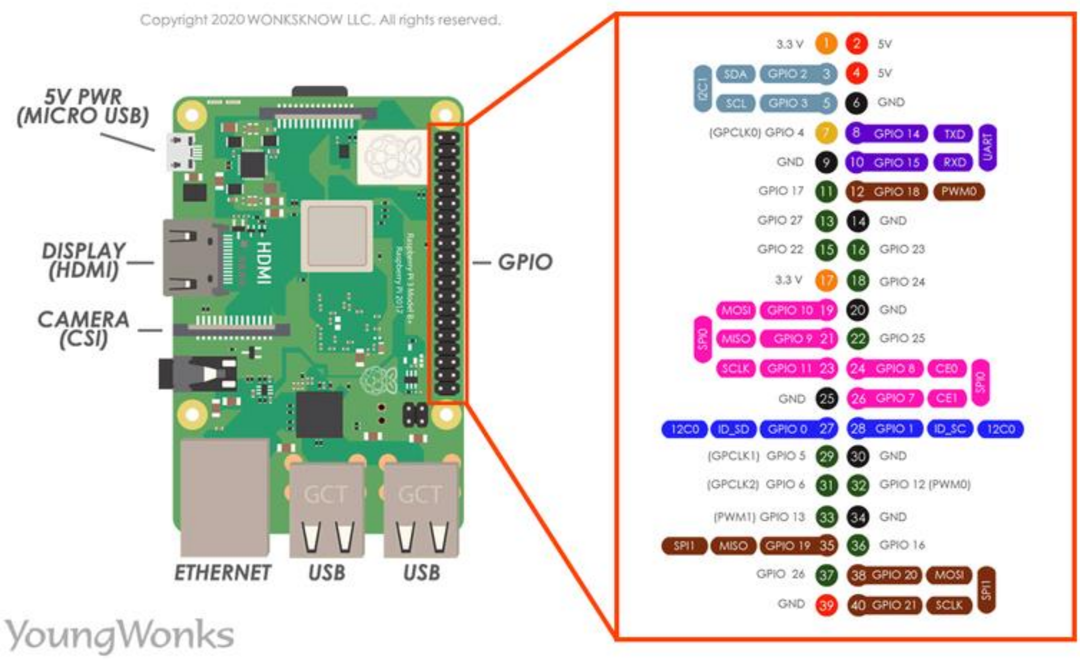
Challenge	Description
Scale	While the scale of IT networks can be large, the scale of OT can be several orders of magnitude larger. For example, one large electrical utility in Asia recently began deploying IPv6-based smart meters on its electrical grid. While this utility company has tens of thousands of employees (which can be considered IP nodes in the network), the number of meters in the service area is tens of millions. This means the scale of the network the utility is managing has increased by more than 1,000-fold! Chapter 5, "IP as the IoT Network Layer," explores how new design approaches are being developed to scale IPv6 networks into the millions of devices.
Security	With more "things" becoming connected with other "things" and people, security is an increasingly complex issue for IoT. Your threat surface is now greatly expanded, and if a device gets hacked, its connectivity is a major concern. A compromised device can serve as a launching point to attack other devices and systems. IoT security is also pervasive across just about every facet of IoT. For more information on IoT security, see Chapter 8, "Securing IoT."
Privacy	As sensors become more prolific in our everyday lives, much of the data they gather will be specific to individuals and their activities. This data can range from health information to shopping patterns and transactions at a retail establishment. For businesses, this data has monetary value. Organizations are now discussing who owns this data and how individuals can control whether it is shared and with whom.
Big data and data analytics	IoT and its large number of sensors is going to trigger a deluge of data that must be handled. This data will provide critical information and insights if it can be processed in an efficient manner. The challenge, however, is evaluating massive amounts of data arriving from different sources in various forms and doing so in a timely manner. See Chapter 7 for more information on IoT and the challenges it faces from a big data perspective.
Interoperability	As with any other nascent technology, various protocols and architectures are jockeying for market share and standardization within IoT. Some of these protocols and architectures are based on proprietary elements, and others are open. Recent IoT standards are helping minimize this problem, but there are often various protocols and implementations available for IoT networks. The prominent protocols and architectures—especially open, standards-based implementations—are the subject of this book. For more information on IoT architectures, see Chapter 2, "IoT Network Architecture and Design." Chapter 4, "Connecting Smart Objects," Chapter 5, "IP as the IoT Network Layer," and Chapter 6, "Application Protocols for IoT," take a more in-depth look at the protocols that make up IoT.

10

5)	**Compare and contrast IOT world forum Standardized architecture & one M2M IOT standardized architecture** **From your understanding – Write by own.** **Key elements for answering this answer:** • Using IoT – WF reference model, we are able to achieve the following: 1. Decompose the IoT problem into smaller parts 2. Identify different technologies at each layer and how they relate to one another 3. Define a system in which different parts can be provided by different vendors 4. Have a process of defining interfaces that leads to interoperability 5. Define a tiered security model that is enforced at the transition points between levels • Layers specifications of both architectures	10
6)	**Differentiate IT and OT and explain the convergence of IT and OT** Until recently, information technology (IT) and operational technology (OT) have for the most part lived in separate worlds. IT supports connections to the Internet along with related data and technology systems and is focused on the secure flow of data across an organization. OT monitors and controls devices and processes on physical operational systems. These systems include assembly lines, utility distribution networks, production facilities, roadway systems, and many more. Typically, IT did not get involved with the production and logistics of OT environments. Management of OT is tied to the lifeblood of a company. For example, if the network connecting the machines in a factory fails, the machines cannot function, and production may come to a standstill, negatively impacting business on the order of millions of dollars. On the other hand, if the email server (run by the IT department) fails for a few hours, it may irritate people, but it is unlikely to impact business at anywhere near the same level. Table below highlights some of the differences between IT and OT networks and their various challenges.	10

Criterion	Industrial OT Network	Enterprise IT Network
Operational focus	Keep the business operating 24x7	Manage the computers, data, and employee communication system in a secure way
Priorities	1. Availability 2. Integrity 3. Security	1. Security 2. Integrity 3. Availability
Types of data	Monitoring, control, and supervisory data	Voice, video, transactional, and bulk data
Security	Controlled physical access to devices	Devices and users authenticated to the network
Implication of failure	OT network disruption directly impacts business	Can be business impacting, depending on industry, but workarounds may be possible
Network upgrades (software or hardware)	Only during operational maintenance windows	Often requires an outage window when workers are not onsite; impact can be mitigated
Security vulnerability	Low: OT networks are isolated and often use proprietary protocols	High: continual patching of hosts is required, and the network is connected to Internet and requires vigilant protection

7) **With a neat diagram, Explain Raspberry Pi learning board with pin configuration** 10

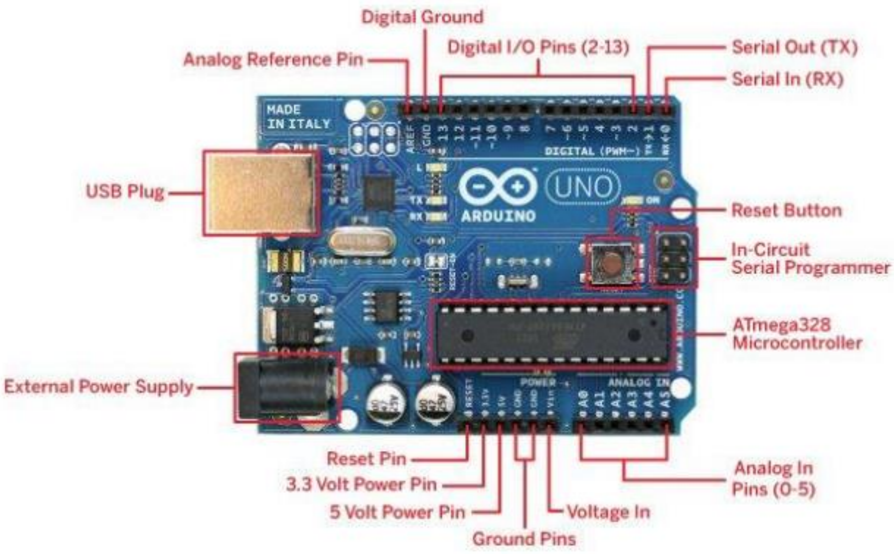


- USB Port- to connect a mouse, a keyboard or other peripherals.
- Ethernet Port- to connect to the internet using an Ethernet cable.
- Audio Jack- to connect an audio device.
- CSI Connector- to connect a camera with a CSI(Camera Serial Interface) ribbon.
- HDMI Connector- to connect a monitor or TV.
- Power Port- to power up your Pi.
- DSI Connector- to connect DSI compatible Display.

8) **Write a raspberry pi program for controlling 3 led's using switches** 10

```
import RPi.GPIO as GPIO
import time

# Pin Definitions
LED_PINS = [17, 27, 22]  # LED GPIO pins
SWITCH_PINS = [5, 6, 13]  # Switch GPIO pins

# Setup GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

# Setup LED pins as output
for pin in LED_PINS:
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, GPIO.LOW)

# Setup switch pins as input with pull-down resistors
for pin in SWITCH_PINS:
    GPIO.setup(pin, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)

try:
    while True:
        for i in range(3):
            if GPIO.input(SWITCH_PINS[i]) == GPIO.HIGH:
                GPIO.output(LED_PINS[i], GPIO.HIGH)  # Turn on LED
            else:
                GPIO.output(LED_PINS[i], GPIO.LOW)  # Turn off LED
        time.sleep(0.1)  # Small delay to avoid bouncing issues

except KeyboardInterrupt:
    print("Program terminated")
    GPIO.cleanup()  # Cleanup GPIO on exit
```

9) **Explain Arduino UNO learning board with its technical specification** 10



The Arduino Uno is a popular open-source microcontroller board that is widely used for prototyping and DIY electronics projects. Here are the technical specifications for the Arduino Uno:

Microcontroller: ATmega328P

Clock Speed: 16 MHz

Flash Memory: 32 KB (2 KB used by the bootloader)

SRAM: 2 KB

EEPROM: 1 KB

Operating Voltage: 5V

Input Voltage (recommended): 7-12V

Input Voltage (limits): 6-20V

Digital I/O Pins: 14 (of which 6 provide PWM output)

PWM Digital I/O Pins: 6

Analog Input Pins: 6

DC Current per I/O Pin: 20 mA

DC Current for 3.3V Pin: 50 mA

Voltage Regulator: AMS1117 5.0V

USB Interface: ATmega16U2

Communication:

Serial Communication: Yes (via USB and hardware UART)

I2C: Yes

SPI: Yes

Clock Source: 16 MHz Crystal Oscillator

Size: 68.6 mm x 53.4 mm

Weight: 25 g

LEDs:

13: Digital Pin 13 (default built-in LED)

TX, RX: Serial communication LEDs

Reset Button: Yes

Power Jack: 2.1mm center-positive

In-Circuit Serial Programming (ICSP) Header: Yes

Operating Temperature Range: -40°C to +85°C

USB Connector: Type-B

Programming: Via USB or ICSP

Bootloader: Yes (Optiboot)

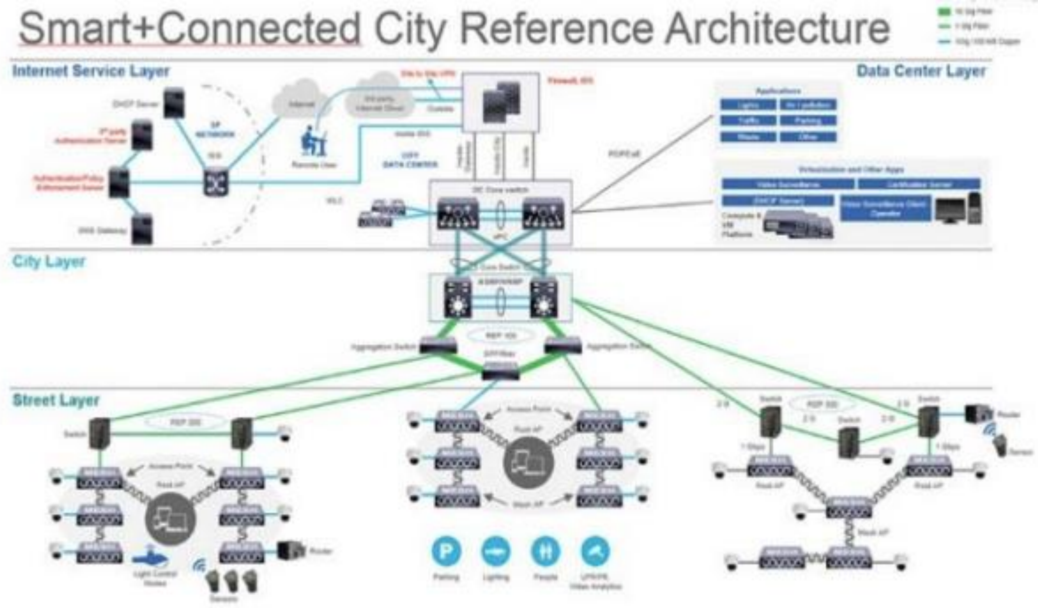
Board Type: Digital

Compatible Shields: Yes (with the standard Arduino form factor)

Open-Source: Yes (Schematics and design files are available for free)

IDE Compatibility: Arduino IDE (and other compatible IDEs)

10)	**Explain smart city Security architecture.** 5.11) Smart City Security Architecture • A serious concern of most smart cities and their citizens is data security. • Vast quantities of sensitive information are being shared at all times in a layered, realtime architecture, and cities have a duty to protect their citizens' data from unauthorized access, collection, and tampering. • In general, citizens feel better about data security when the city itself, and not a private entity, owns public or city-relevant data. • It is up to the city and the officials who run it to determine how to utilize this data. • When a private entity owns city-relevant data, the scope of the ownership may initially be very clear. • However, later considerations or changes in the private entity strategy may shift the way the data is used. • It may then be more difficult for city authorities or the citizens to oppose this new direction, simply because they do not have any stake in the decision-making process ofthe private entity. • For example, suppose that a private contractor is in charge of collecting and managing parking sensor data. • One possible way to increase the profitability of such data is to sell it to insurance companies looking to charge an additional premium to car owners parking in the street (vs. in a covered and secured garage). • Such deviations from the original mandate are less likely to happen when cities own the data and when citizens have a way to vote against such usages. • A security architecture for smart cities must utilize security protocols to fortify each layer of the architecture and protect city data. • Figure 5.15 shows a reference architecture, with specific security elements highlighted. Security protocols should authenticate the various components and protect data transport throughout. • For example, hijacking traffic sensors to send false traffic data to the system regulating the street lights may result in dramatic congestion issues. • The benefit for the offender may be the ability to get "all greens" while traveling, but the overall result would typically be dangerous and detrimental to the city. • The security architecture should be able to evolve with the latest technology and incorporate regional guidelines (for example, city by-laws, county or regional security regulations).	10



The following are common industry elements for security on the network layer:

- **Firewall:** A firewall is located at the edge, and it should be IPsec- and VPN-ready, and include user- and role-based access control.
- It should also be integrated with the architecture to give city operators remote access to the city data center.
- **VLAN:** A VLAN provides end-to-end segmentation of data transmission, further protecting data from rogue intervention. Each service/domain has a dedicated VLAN for data transmission.
- **Encryption:** Protecting the traffic from the sensor to the application is a common requirement to avoid data tampering and eavesdropping.
- In most cases, encryption starts at the sensor level. In some cases, the sensor-to gateway link uses one type of encryption, and the gateway-to-application connection uses another encryption (for example, a VPN).