**UNIT-III**

**M2M Security:**

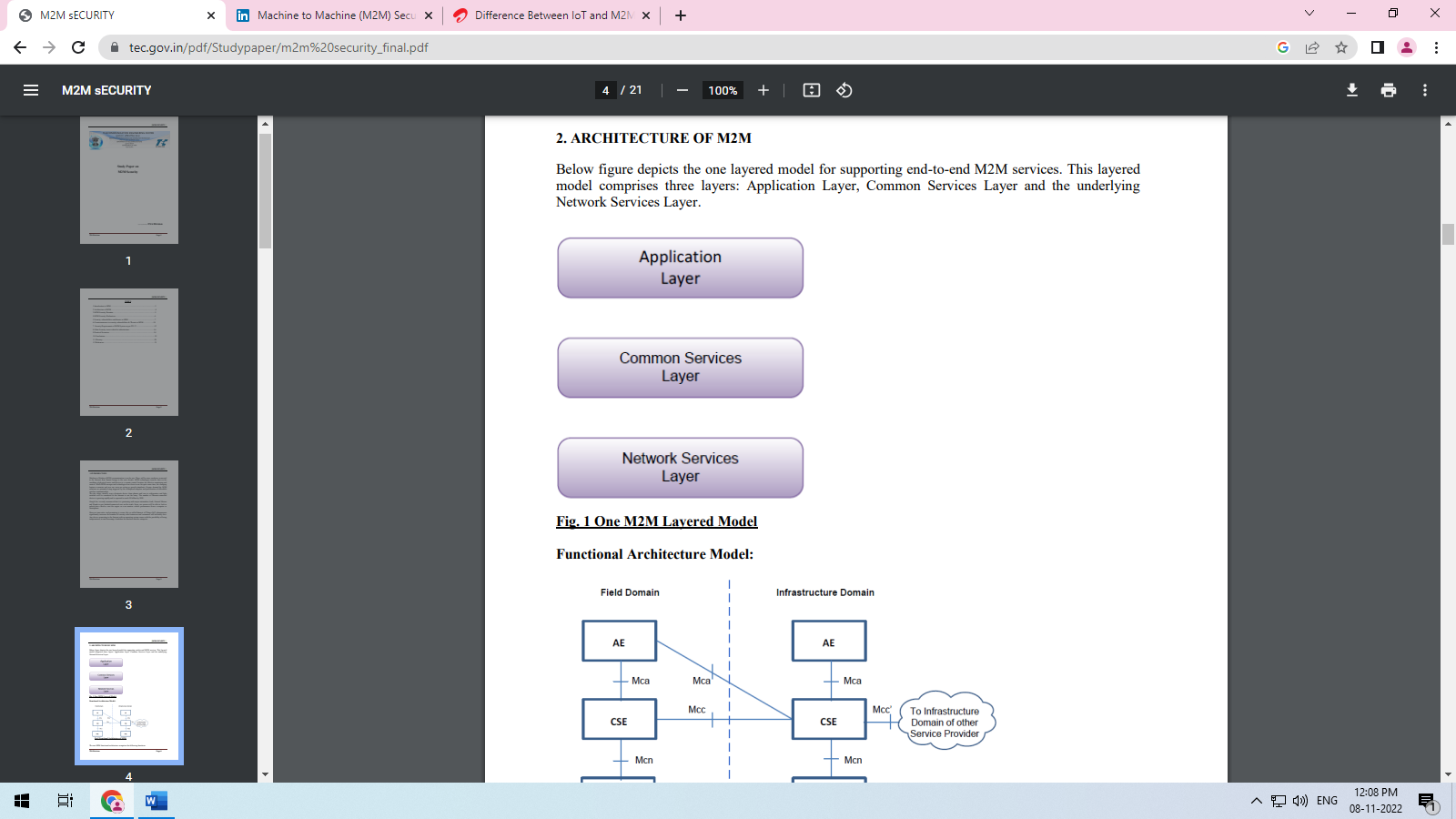
* Machine-to-machine (M2M) communications is used for **automated data transmission and measurement between mechanical or electronic devices**.
* Machine to Machine (M2M) communication is on the rise. There will be more machines connected to the Internet than human beings in the next decade.
* M2M technologies transfer data on the condition of physical assets and devices to a remote central location for effective monitoring and control.
* While M2M concepts and technologies have been in use for quite some time, the changing business scenarios and new use cases are acting as growth stimulants.

Common examples of machine-to-machine technology are controlling electrical devices like fans and bulbs using smartphone’s Bluetooth. Here, the smartphone and the electrical devices are the two interacting devices. Another example is the smart meter, which can track electric consumption in real-time. M2M technology is widely used in applications such as tracking and tracing, automation, metering, healthcare, etc.

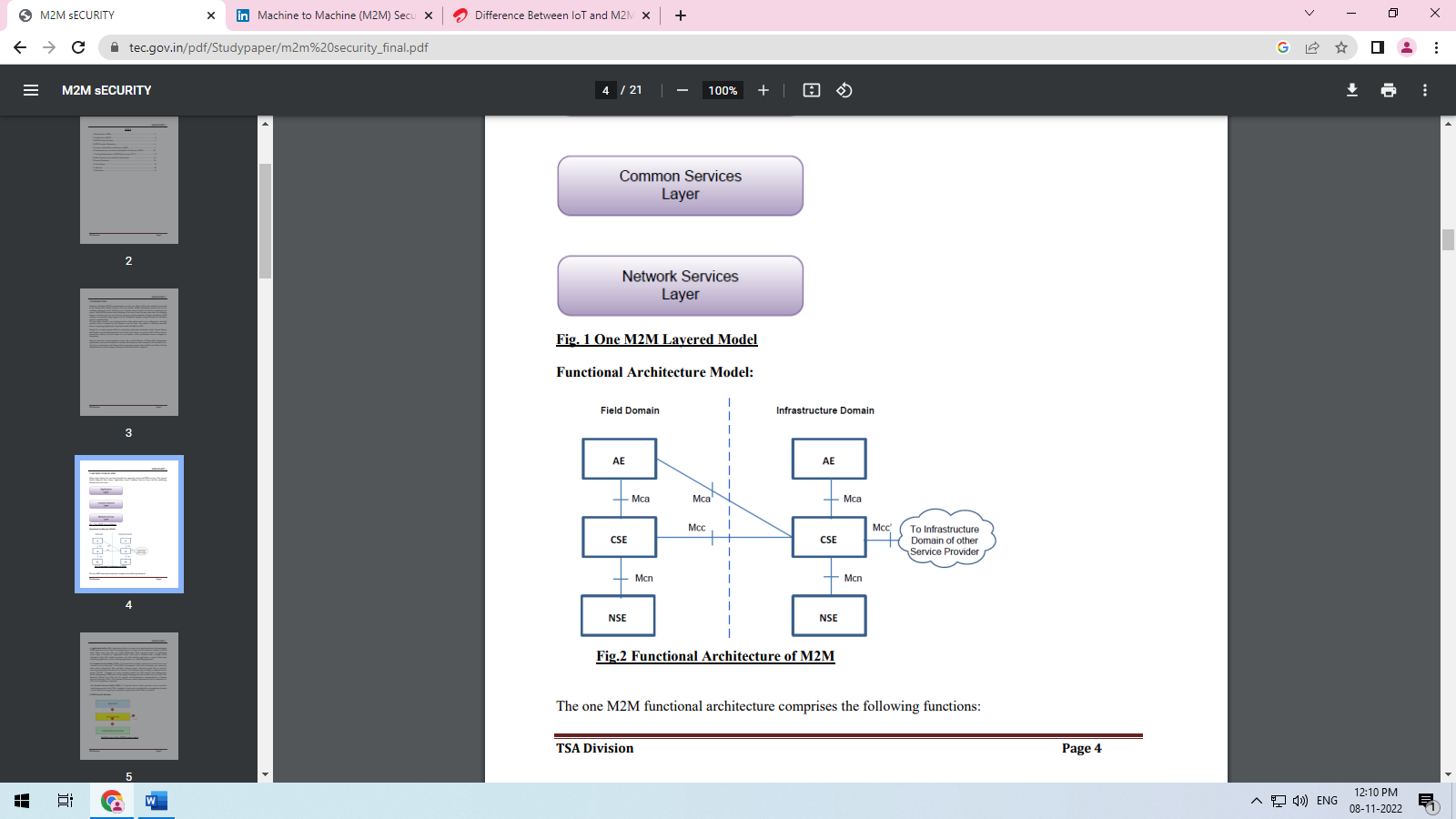
**M2M technology has many benefits, including but not limited to:**

* **cost-effective**
* **easy to maintain**
* **improves customer service by proactive monitoring and servicing. Architecture Of M2m**

Below figure depicts the one layered model for supporting end-to-end M2M services. This layered model comprises three layers: Application Layer, Common Services Layer and the underlying Network Services Layer.

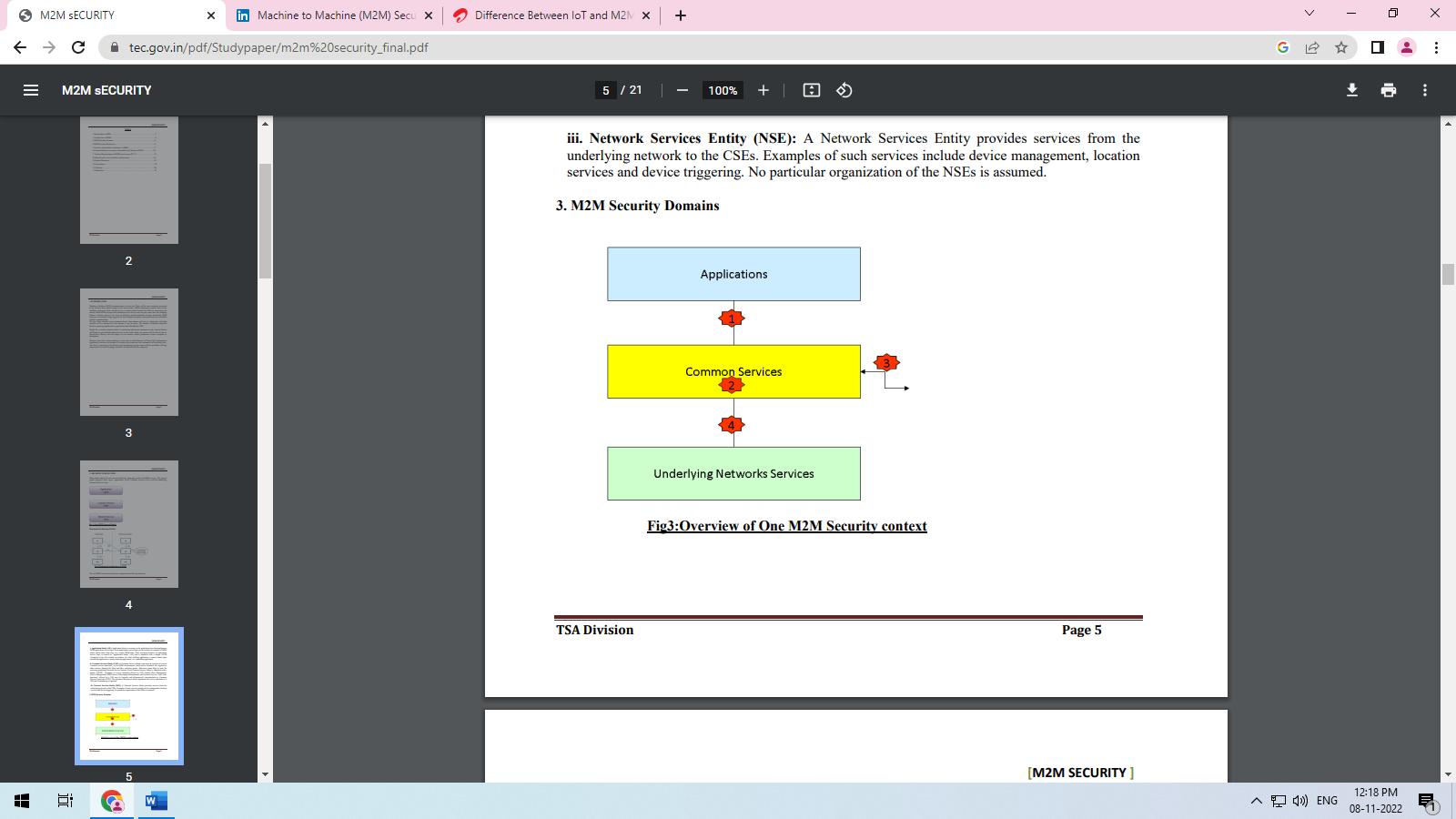


**One M2M Layered Model**

**Functional Architecture Model:**

1. **Application Entity (AE):** Application Entity is an entity in the application layer that implements M2M application service logic. Each application service logic can be resident in a number of M2M nodes and/or more than once on a single M2M node. Each execution instance of application service logic is termed an "Application Entity" (AE) and is identified with a unique AE-ID .Examples of the AEs include an instance of a fleet tracking application, a remote blood sugar monitoring application, a power metering application, or a controlling application.
2. **Common Services Entity (CSE):** A Common Services Entity represents an creation of a set of "common service functions" of the M2M environments. Such service functions are exposed to other entities through the Mca and Mcc reference points. Reference point Mcn is used for accessing underlying Network Service Entities. Each Common Service Entity is identified with a unique CSE-ID. Examples of service functions offered by CSE include: Data Management, Device Management, M2M Service Subscription Management, and Location Services. Such "subfunctions" offered by a CSE may be logically and informatively conceptualized as Common Services Functions (CSFs). The normative Resources which implement the service functions in a CSE can be mandatory or optional.
3. **Network Services Entity (NSE):** A Network Services Entity provides services from the underlying network to the CSEs. Examples of such services include device management, location services and device triggering. No particular organization of the NSEs is assumed.

**M2M Security Domains**



Based on one M2M architecture model, four security domains are identified. Each of these domains provides security features to meet certain threats and in particular protect against attacks, in associated trust scenarios.

1. **Application domain security:** The set of security features that enable Applications and Common Services to securely exchange messages and protect against attacks on the Mca Reference Points. An M2M Application Service Provider can rely on independent credentials to secure its End-to-End communications, so that application related information is exposed to either the M2M Service Provider or the underlying network operator.
2. **Intra Common Services domain security:** The set of security features that enable Common Service Functions in the Common Service Entity to securely exchange messages and which in particular protect against attacks on the CSE. In cases where the M2M Service provider is trusted to provide security to the M2M Application, the ability to secure communication between nodes for the purpose of the M2M Service Layer can be made directly available by M2M Service Providers to the M2M Applications through an API.
3. **Inter Common Services domain security:** the set of security features that enable secure exchange of messages between CSEs and protect against attacks on the Mcc Reference Points. iv. Underlying Network security: the set of security features that enable Underlying Network Services and Common Services to securely exchange messages and protect against attacks on the Mcn Reference Points.

**Key Differences between M2M and IoT**

M2M and IoT are not the same. IoT requires M2M, but M2M does not need IoT. M2M systems are often isolated, and IoT systems take M2M to an enhanced level. They bring isolated systems together into a large connected ecosystem.

**Some of the key differences between these two technologies are:**

|  |  |  |
| --- | --- | --- |
| **Basis** | **M2M** | **IoT** |
| **Connection Type Used** | Simple device-to-device communication usually within an embedded software at the client site. | Devices use IP networks to communicate. |
| **Communication** | Communication directly between machines. | IoT sensors automation. |
| **Communication Protocol Used** | Communication technology techniques and traditional protocols. | Internet protocols like HTTP, FTP and Telnet. |
| **Intelligence** | Observation of some degree of intelligence. | Objects are responsible for decision-making. |
| **Technology** | Hardware-based. | Hardware and Software-based. |
| **Data Sharing** | Data sharing among communicating parties only. | Data sharing between other applications to improve the end-user experience. |
| **Scope** | Deployed in a closed system. | Connects to a larger network. |
| **Open API Support** | No open API support. | Supports open API integration. |
| **Internet** | Devices do not rely on internet connection. | An active internet connection is required. |

**RFID Security:**

* RFID tags are a type of tracking system that uses smart barcodes in order to identify items.
* RFID is short for **“radio frequency identification**,” and as such, RFID tags utilize radio frequency technology.
* These radio waves transmit data from the tag to a reader, which then transmits the information to an RFID computer program.
* RFID tags are frequently used for merchandise, but they can also be used to track vehicles, pets, and even patients with Alzheimer’s disease.
* An RFID tag may also be called an **RFID chip.**

**Advantages of RFID:**

* RFID technology automates data collection and vastly reduces human effort and error
* RFID supports tag reading with no line-of-sight or item-by-item scans required
* RFID readers can read multiple RFID tags simultaneously, offering increases in efficiency
* All RFID tags within range can be detected instantly and matched with information in your database
* Assets can be cross-referenced against assigned locations and recorded as present, missing, or relocated
* RFID can be integrated with active scanning and fixed readers for a totally automated tracking solution
* Assets and employees can be tracked and located automatically for everything from supply chain and asset management to facility security and emergency planning
* Available scanners support both RFID and barcoding so you can upgrade at your own pace.