System Specification for Secure IoT Solution (PoC)

1. Architecture

Components:

• Sensor Device (Virtual Device via Docker): A Docker container running a Python script to simulate sensor data and send it to the backend.

• InfluxDB (Database): Stores incoming sensor data from the sensor device.

• Grafana (Visualization): Visualizes sensor data in real-time.

• Docker Compose (Infrastructure): Manages the installation and integration of all components within a connected network.

2. Communication Flow

Data Flow:

• Sensor Device ➔ InfluxDB: The sensor device generates data and sends it via HTTP POST to the InfluxDB API.

• InfluxDB ➔ Grafana: Grafana retrieves data from InfluxDB and displays it on dashboards.

Protocols:

• HTTP: For communication between devices and services.

• Internal Docker Network: Ensures that communication occurs within a closed network without exposure to the internet.

3. Security Measures

• Isolation with Docker: Each component runs in its own Docker container, protecting against security vulnerabilities by keeping the system isolated.

• Secure Communication (HTTPS or VPN): Option to implement TLS certificates to encrypt data transmissions, preventing eavesdropping and data tampering.

• Authentication and Authorization: Grafana and InfluxDB are configured with usernames and passwords to ensure that only authorized users can access the systems.

4. Cyber Resilience Act (CRA) Requirements

Security-by-Design:

• Isolation and Containerization: The system is built with isolation of each component to prevent vulnerabilities from spreading.

• Encrypted Communication: Option to add TLS to secure data transmissions.

Updatability:

• Docker Containers: Each component can be easily updated by rebuilding the container, ensuring that the system can be kept up-to-date with security fixes without disrupting other parts.

Vulnerability Management:

• Rapid Updates via Docker Compose: If a vulnerability is discovered, new versions of each component can be quickly deployed by running new versions of the containers.

• Vulnerability Scanning: Docker has tools to scan and detect vulnerabilities in containers, ensuring you can act quickly if issues are discovered.