SISEN: Medical IoT

Security-Informed Safety in Wireless Networks

Pre-Lab Checks

Before starting this scenario, refer to the SISEN Student Guide and confirm that SISEN has been installed and configured correctly.

The student guide is available **online** and in the SISEN repository under `teaching-materials/labs/`.

Activity Worksheet

Complete the practical activity using the SISEN Hazard Analysis and Evidence Sheet.
The worksheet is available **online** and in the SISEN repository under `teaching-materials/labs/`.

Scenario Purpose

This scenario introduces a simulated medical Internet of Things environment in which patient monitoring depends on trustworthy and timely telemetry.

A wearable sensor simulator generates patient vital-sign data, including heart rate, blood oxygen saturation, and blood pressure. The data is passed through a simulated BLE-to-Wi-Fi gateway, published to an MQTT broker, and displayed on a patient monitoring dashboard.

The scenario demonstrates how a security failure affecting the integrity or availability of patient telemetry can develop into a safety-relevant clinical hazard.

Learning Outcomes

By completing this activity, you should be able to:

- explain how patient monitoring depends on trustworthy and timely telemetry;
- describe the data flow between the wearable simulator, gateway, MQTT broker, and monitoring dashboard;
- observe normal patient telemetry using the dashboard and MQTT tools;
- identify how spoofed, missing, delayed, or disrupted telemetry affects system behaviour;
- collect evidence of normal and abnormal system operation;
- distinguish between a security event, an operational effect, and a safety-relevant hazard;
- complete a structured hazard analysis for the scenario; and
- evaluate appropriate mitigation and resilience controls.

Scenario Overview

The scenario represents a simplified remote patient monitoring system.

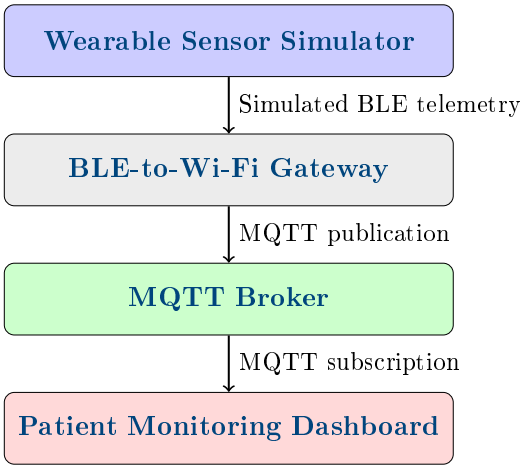
The simulated wearable device produces physiological readings and sends them to a gateway. The gateway forwards the readings through MQTT to a patient monitoring dashboard. The dashboard acts as the primary interface through which clinical staff would observe the patient's condition.

The system assumes that all received telemetry is trustworthy. It does not independently verify that the readings are authentic, current, or physiologically plausible.

- System focus: simulated remote patient monitoring
- Communication path: wearable simulator to BLE-to-Wi-Fi gateway to MQTT broker to dashboard
- Core services: wearable simulator, gateway, MQTT broker, and patient monitoring dashboard
- MQTT namespace: `patient/#`
- Safety relevance: incorrect or unavailable telemetry may result in inappropriate or delayed clinical intervention

Scenario Architecture

Shown here is the simplified system architecture. The wearable simulator generates vital-sign data. The gateway receives and forwards the data. The MQTT broker distributes the telemetry to subscribed services. The dashboard displays the latest readings and associated alert state.



Main Components

Wearable Sensor Simulator

The wearable sensor simulator generates patient vital-sign data, including:

- heart rate;
- blood oxygen saturation, or SpO_2; and
- blood pressure.

The readings vary over time to represent normal physiological variation.

BLE-to-Wi-Fi Gateway

The gateway represents the intermediary device between the wearable and the wider monitoring network.

It receives simulated wearable data and forwards the telemetry to the MQTT broker. The gateway does not perform strong authenticity, integrity, or physiological plausibility checks.

MQTT Broker

The MQTT broker receives telemetry from the gateway and distributes it to subscribed services.

The medical telemetry is published under:

```
patient/vitals/#
```

The broker provides the central communication mechanism for the scenario.

Patient Monitoring Dashboard

The dashboard displays the latest patient readings.

During normal operation, it should show continuously updating values for:

- heart rate;
- SpO_2; and
- blood pressure.

The dashboard trusts the telemetry received through MQTT and does not independently confirm whether the readings are authentic.

Safety Context

Patient monitoring systems support clinical awareness and decision-making.

If a patient begins to deteriorate, accurate telemetry may allow clinical staff to recognise the change and intervene. If the data is false, delayed, replayed, or unavailable, the dashboard may no longer reflect the patient's actual condition.

Examples of possible consequences include:

- a stable patient appearing to be in a critical condition;
- a deteriorating patient appearing to be stable;
- patient telemetry disappearing from the dashboard;
- old readings being presented as current readings; and
- clinical staff losing confidence in the monitoring system.

The security properties of the telemetry path are therefore directly connected to patient safety.

1 Start the Medical IoT Scenario

Before starting the scenario, decide how the wireless environment will be configured.

1.1 Wireless Access Point

The scenario runs through the SISEN wireless environment. When prompted, select the required Wi-Fi access point mode.

The selected access point determines how the simulated devices connect to the network and which wireless security properties are available for observation or testing.

Use the access point mode specified by the instructor.

1.2 Number of Simulated Nodes

The direct launch example creates four simulated patient wearables. The `-patients` option accepts a value from 1 to 10.

Each wearable uses the active `patient-N` identifier format, such as `patient-1`, `patient-2`, `patient-3`, and `patient-4`. Increasing the number of patients increases the amount of traffic generated and the number of patient cards and MQTT streams visible in the scenario.

Use the number of patients specified by the instructor.

1.3 Start the Scenario

From the project root directory, run:

```
python3 launch_sisen.py --scenario medical --ap-mode wep --patients 4 --capture-hints
```

2 Observe Normal Operation

Open the patient monitoring dashboard using the address shown by the scenario runner.

```
http://localhost:5000
```

Observe the dashboard for at least two minutes. During normal operation, you should see:

- continuously updating heart-rate readings;
- continuously updating SpO_2 readings;
- continuously updating blood-pressure readings;
- values changing gradually rather than remaining completely static; and
- no persistent critical or missing-data warning.

Do not perform any attack or disruption during this observation period. Record in your activity document:

- the normal range of values shown for heart rate, SpO_2, and blood pressure; and
- how frequently the dashboard appears to update.

3 Optional: Inspect MQTT Telemetry

This step is not required for the scenario to operate. The gateway and dashboard already subscribe to the relevant MQTT topics automatically.

Use this additional step to inspect the raw telemetry directly and compare it with the values displayed on the dashboard.

Open a second terminal and subscribe to the medical telemetry namespace:

```
mosquitto_sub -h localhost -t 'patient/#' -v
```

Observe the messages for at least one minute. Identify:

- the MQTT topics used;
- the values contained in each message;
- the frequency of publication;
- whether timestamps are included; and
- whether the messages are readable in plaintext.

Compare the MQTT messages with the values shown on the dashboard. Record in your activity document:

- the MQTT topics observed;
- one set of patient readings shown in both the MQTT message and the dashboard; and
- whether the MQTT data appears to be encrypted or protected from unauthorised subscribers.

4 Examine the Communication Flow

Use the relevant packet capture activity from the SISEN Attack and Disruption Guide.

Capture the traffic on the interface specified for the selected scenario and examine the MQTT communication between the gateway, broker, and dashboard.

Use Wireshark and apply the following MQTT display filter:

```
mqtt
```

Observe the traffic generated during normal operation. Identify:

- the source and destination addresses;
- the MQTT broker port;
- MQTT publish messages;
- MQTT topic names; and
- whether the payload is visible.

Record in your activity document:

- one example MQTT publish message;
- the associated topic and payload; and
- what information an unauthorised observer could learn from the captured traffic.

5 Perform a Bounded Attack or Disruption

This activity demonstrates how attacks or disruptions at different layers of the system can influence patient monitoring and create a safety-relevant clinical effect. Attacks may fall into one or more of the following categories:

- **Infrastructure Attack:** An infrastructure attack affects the supporting wireless or network environment, such as the Wi-Fi access point, BLE-to-Wi-Fi gateway, network connectivity, or MQTT infrastructure.
- **Communication or Protocol Attack:** A communication or protocol attack affects the exchange, forwarding, or delivery of data, such as BLE communication, Wi-Fi transport, MQTT publication or subscription, replay, message injection, or traffic disruption.
- **Telemetry or Application-Layer Attack:** A telemetry or application-layer attack affects the patient readings, timing, freshness, authenticity, plausibility, or availability of the data presented to the monitoring dashboard.

5.1 Select the Disruption Activity

Use the relevant activity from the SISEN Attack and Disruption Guide. Available helper-based activities can be listed from the repository root using `python3 attacks/run_attack.py -list -scenario medical`. The guide provides the available bounded attacks and disruptions, including:

- the disruption type;
- the target component or communication path;
- the affected security property;
- the expected technical effect;
- the required command or helper script; and
- the stopping and recovery procedure.

Use the disruption activity specified by the instructor.

5.2 Before Running the Disruption

Confirm that:

- the Medical IoT scenario is running;
- the dashboard is displaying normal telemetry;
- the gateway and wearable processes are active;
- normal telemetry has been observed on the dashboard; and
- any required packet capture is already running.

Record the most recent normal patient readings before continuing.

5.3 Run the Disruption

Run the selected bounded activity using the current attack runner command in the SISEN Attack and Disruption Guide. Medical IoT activities are organised under categories including authenticity, integrity, replay, and safety-case. Some activities target one active patient, while broader activities may affect several active `patient-N` identifiers.

While it is active, do not restart the scenario or manually publish additional MQTT messages unless instructed.

6 Observe the System Effect

Compare the system behaviour during the disruption with the normal baseline recorded earlier. Examine:

- the patient monitoring dashboard;
- the MQTT telemetry;
- the gateway output;
- the wearable simulator output; and
- the packet capture, where available.

Determine:

- which component first showed evidence of the disruption;
- whether the dashboard displayed false, stale, missing, or inconsistent data;
- whether the gateway detected or rejected the abnormal telemetry;
- whether the effect remained limited to one data stream or affected the wider monitoring system; and
- whether the system recovered automatically after the disruption stopped.

Record the findings in the separate activity document.

7 Stopping the Disruption

Helper-based telemetry attacks are bounded demonstrations. Where an attack needs to remain visible, the helper refreshes the injected value for approximately ten seconds and then exits. Normal scenario telemetry continues during the activity and should restore the dashboard afterwards. For interface-level or manual activities, use the stopping or recovery procedure provided in the SISEN Attack and Disruption Guide.

Confirm that the abnormal activity has stopped. Observe whether normal telemetry resumes automatically. Record in your activity document:

- whether the system recovered automatically;
- how long recovery took; and
- whether any stale or abnormal data remained visible.

8 Distinguish the Security Event from the Hazard

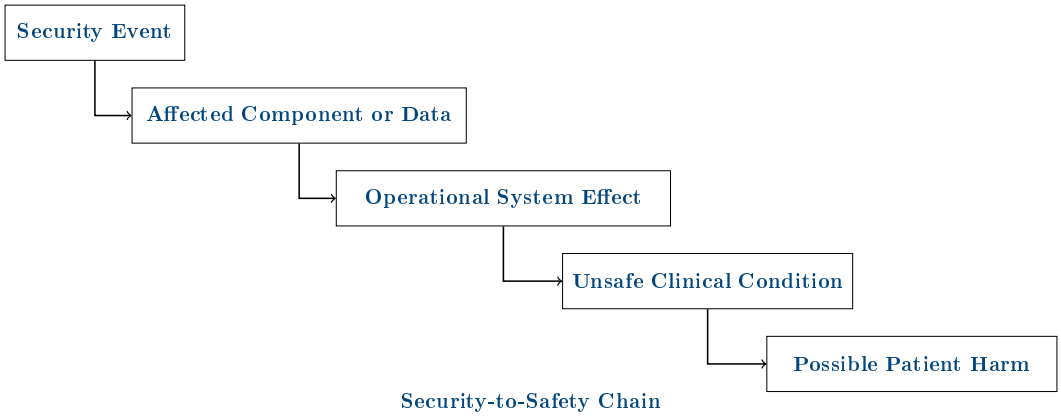
A security event and a safety hazard are not the same.

The security event is the compromise or disruption that affects the system. The operational effect is the resulting change in system behaviour.

The hazard is the condition that may contribute to patient harm.

Using the observed disruption, complete the **security-to-safety chain** in your activity document. Address the following points:

- What security event occurred?
- Which component, connection, or data item was affected?
- What immediate operational effect followed?
- What unsafe clinical condition could result?
- What patient harm could occur if the problem were not detected?



Security-to-Safety Chain

9 Hazard Analysis

Complete a structured hazard analysis for the observed event in your activity document.

Your analysis should identify:

- the initiating security event;
- the affected asset or function;
- the loss of confidentiality, integrity, availability, authenticity, or accountability;
- the immediate system effect;
- the safety-relevant hazard;
- the possible clinical consequence;
- existing controls;
- proposed mitigations; and
- remaining residual risk.

Use the following questions to guide your analysis:

- What evidence confirms that the telemetry was disrupted or falsified?
- Was integrity, availability, authenticity, or more than one property affected?
- Did the dashboard fail visibly or continue to present misleading information?
- Who would rely on the affected information?
- What incorrect decision might be made?
- What condition could expose the patient to harm?
- What control could prevent the event?
- What control could detect the event?
- What control could reduce the safety impact after detection?

10 Mitigation and Resilience

Consider how the system could be improved. Possible controls include:

- MQTT authentication;
- encrypted MQTT communication;
- topic-level access control;
- message authentication or digital signatures;
- gateway input validation;
- physiological plausibility checking;
- replay protection;
- timestamps and freshness checks;
- rate limiting;
- missing-data detection;
- independent alarm channels;
- local gateway fail-safe behaviour;
- audit logging; and
- manual clinical verification.

For each selected control, explain whether it:

- prevents the security event;
- detects the event;
- limits the operational effect;
- reduces the safety consequence; or
- supports recovery and investigation.

Select three controls and explain in your activity document:

- where each control would be implemented;
- what failure or attack it addresses;
- how it improves safety or resilience; and
- what limitation or residual risk remains.

11 Evidence to Collect

Collect the following evidence during the activity:

- scenario startup output;
- dashboard during normal operation;
- normal MQTT telemetry;
- one normal network capture observation;
- the disruption command or helper used;
- dashboard behaviour during the disruption;
- abnormal or missing MQTT telemetry;
- relevant gateway or scenario logs;
- evidence showing whether normal operation resumed or manual recovery was required;
- the completed security-to-safety chain; and
- the completed hazard analysis.

Evidence references should identify the relevant application, command, timestamp, packet number or time range, topic, filename, or reading clearly.

Do not submit screenshots without explaining what each screenshot demonstrates.

12 Reflection Questions

Answer the following questions in your activity document:

1. Why is falsified telemetry potentially more dangerous than unavailable telemetry?
2. Which component acts as the main trust boundary in this scenario?
3. Why is MQTT broker authentication alone insufficient to guarantee trustworthy patient data?
4. How could replayed telemetry cause a patient-safety hazard?
5. What should the dashboard do when telemetry becomes unavailable?
6. What should the gateway do when it receives physiologically impossible readings?
7. Which controls primarily improve security, and which primarily reduce the safety impact?
8. What evidence would be required to demonstrate that the proposed controls are effective?

13 Stopping the Scenario

Stop the Medical IoT scenario using the provided cleanup command:

```
python3 launch_sisen.py --stop
```

Confirm that:

- the wearable simulator has stopped;
- the BLE-to-Wi-Fi gateway has stopped;
- the dashboard has stopped;
- temporary processes have been removed;
- no stale MQTT publishers remain; and
- the environment is ready for another scenario run.

14 Summary

This activity demonstrated how a medical IoT monitoring system depends on the integrity, authenticity, and availability of patient telemetry.

The technical security event was examined alongside its operational effect and possible patient-safety consequence. The activity also demonstrated that protecting the communication channel alone is not sufficient. A safety-relevant system must also detect implausible, stale, missing, or unauthorised telemetry and respond in a fail-safe manner.