Hazard Analysis and Assurance

Dr Ayman El Hajjar

CyBOK Mapping

The lecture maps to the following CyBOK Knowledge Areas:

Full Coverage:

- **Human, Organisational & Regulatory Aspects → Risk Management & Governance**

Partly Covered:

- **Infrastructure Security → Cyber Physical Systems**
- **Human, Organisational & Regulatory Aspects → Human Factors**

Outline

Understanding Hazards

Hazard Analysis Techniques

Building Assurance

Evaluating Assurance

Understanding Hazards

What is a Hazard?

A hazard is a condition, event, or circumstance that has the potential to cause harm to people, systems, property, or the environment.

- Hazards represent **potential sources of harm**
- A hazard does not necessarily result in an incident
- Hazard analysis aims to identify hazards before harm occurs

Hazard Example

Connected System

- Sensor battery failure
- Wireless communication loss
- Incorrect data displayed
- Device reprogrammed by an attacker

Hazard Interpretation Each situation has the potential to cause harm and should therefore be considered during hazard analysis.

From Hazard to Control

Core Concepts

- **Hazard**: the potential source of harm
- **Risk**: the likelihood and severity of harm occurring
- **Control**: a measure used to reduce the risk

Hazard Analysis Hazard analysis identifies hazards, assesses risks, and determines appropriate controls.

Hazard, Risk, and Control Example

Connected System

- **Hazard**: wireless interface lacks authentication
- **Risk**: an attacker modifies system data, causing unsafe outcomes
- **Control**: authentication and integrity checks

Reasoning Pattern This example shows how hazards create risks and how controls reduce potential harm.

Hazard Analysis Techniques

Hazard Analysis Techniques

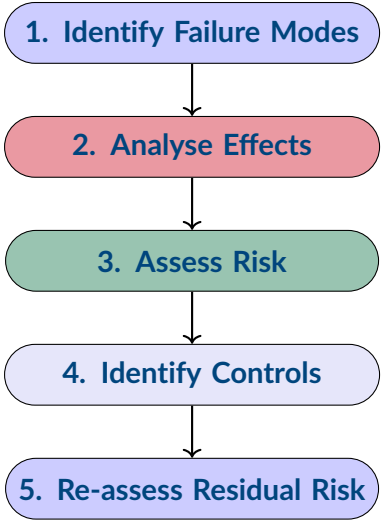
Structured techniques are used to identify hazards, understand their causes, and determine appropriate controls.

- **Preliminary Hazard Analysis (PHA)**
  - Early identification of potential hazards during system design.
- **Failure Modes and Effects Analysis (FMEA)**
  - Examines component failures and their consequences.
- **Fault Tree Analysis (FTA)**
  - Works backwards from an undesired event to determine possible causes.
- **Hazard and Operability Study (HAZOP)**
  - Identifies deviations from intended operation using structured guidewords.

Failure Mode and Effects Analysis (FMEA)

FMEA is a systematic method used to identify failure modes, assess their impact, and determine appropriate controls.

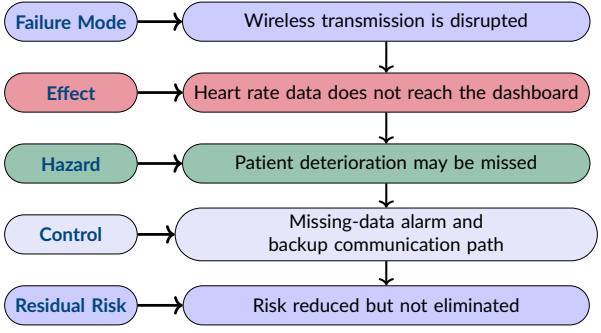
FMEA Output **FMEA** produces a prioritised list of failures and associated controls.



FMEA Example: Medical Sensor Network

Scenario

A wearable heart rate sensor sends patient data to a monitoring dashboard via a wireless link, where one failure is traced from technical cause to safety impact and control.



Fault Tree Analysis (FTA)

FTA is a top-down deductive method used to identify causes of system failures.

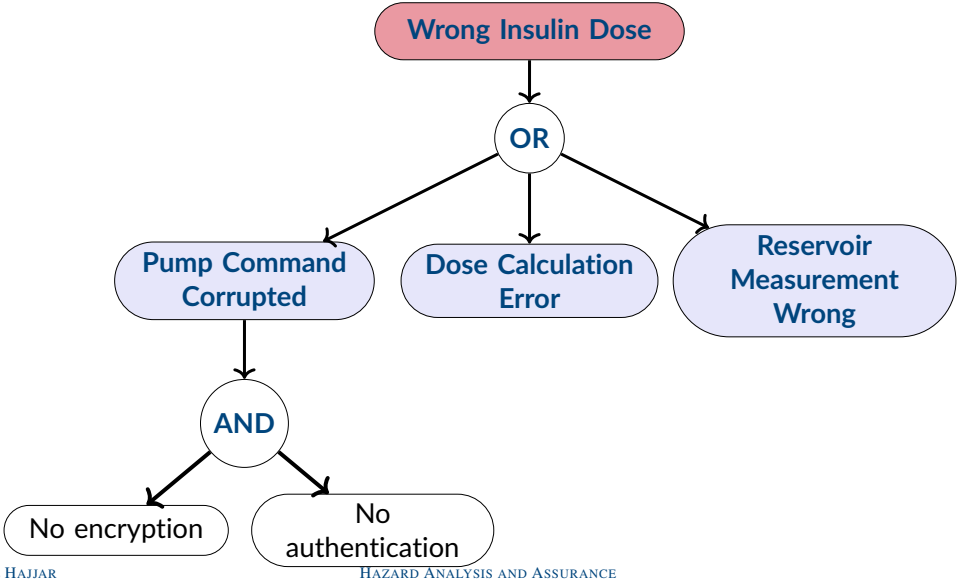
- Start with the undesired top event
- Identify events that could cause it
- Use **AND** and **OR** gates to model relationships
- Determine critical failure combinations (**minimal cut sets**)

Medical IoT Example

Patient Receives Wrong Insulin Dose

Possible causes include corrupted pump commands, incorrect dose calculations, or faulty reservoir measurements.

FTA Diagram Example



Hazard and Operability Study (HAZOP)

HAZOP is a structured technique that uses guide words to identify deviations from intended system behaviour.

Common Guide Words

- **NO/NOT**: Complete failure of a function
- **MORE/LESS**: Higher or lower than intended
- **EARLY/LATE**: Timing deviation
- **REVERSE**: Opposite action occurs
- **OTHER**: Unexpected behaviour or mode

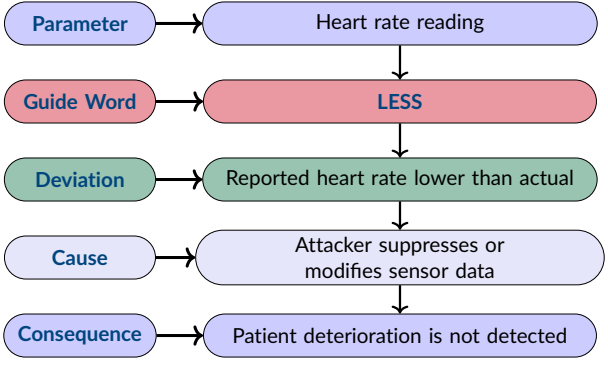
Heart Rate Monitoring Example

A wearable heart rate sensor sends data to a dashboard. HAZOP guide words help identify hazardous deviations.

HAZOP Example: Medical Sensor Network

Scenario

A wearable heart rate sensor sends patient data to a monitoring dashboard via a wireless link.



Common Pitfalls in Hazard Analysis

- **Ignoring Security Threats**: Assuming internal networks are safe or attackers cannot be insiders
- **Underestimating Attack Likelihood**: Assuming medical devices are unlikely targets
- **Weak Evidence**: Relying on limited testing or vendor claims
- **Forgotten Assumptions**: Assuming firmware or supply chains are trustworthy without evidence
- **Changing Threat Landscape**: Failing to account for new attack techniques

Building Assurance

From Hazard Analysis to Assurance

After identifying **hazards**, analysing **risks**, and introducing **controls**, how do we demonstrate that a system is safe and secure?

Assurance Case An assurance case provides **claims**, **arguments**, and **evidence** that identified risks have been adequately addressed.

Building Assurance Cases

An assurance case documents why identified risks have been adequately addressed.

Core Components

- **Claim**: What we want to demonstrate
- **Argument**: Why the claim should be believed
- **Evidence**: Information supporting the argument

Medical IoT Example

- **Claim**: The system will not deliver an incorrect insulin dose due to attack
- **Argument**: Wireless communications are authenticated and protected
- **Evidence**: Penetration testing and security audit reports

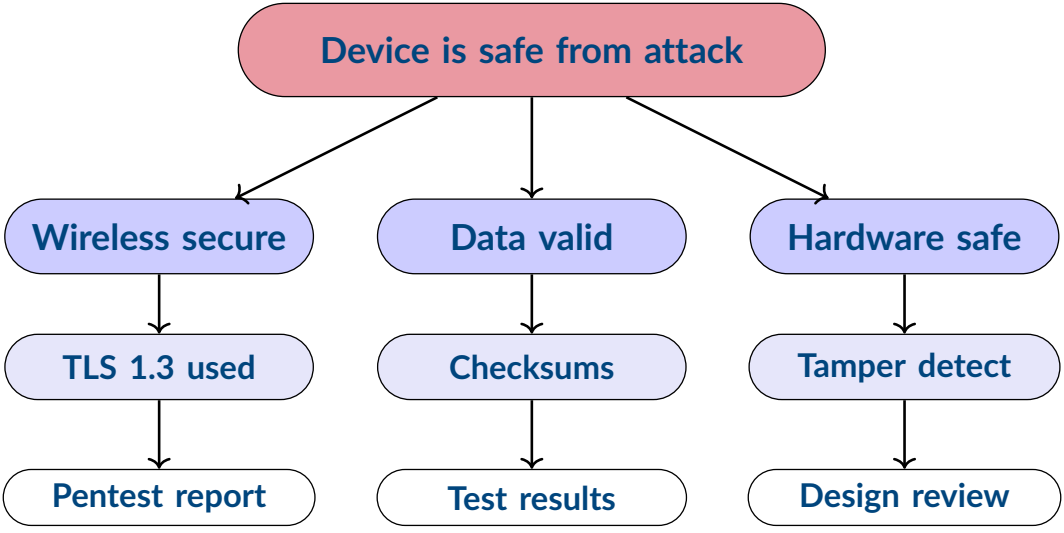
Evidence Types

Evidence provides confidence that claims made in an assurance case are valid.

Common Evidence Types

- **Design Evidence**: Architecture documents, threat models, design reviews
- **Implementation Evidence**: Code reviews, static analysis reports
- **Testing Evidence**: Functional testing, penetration testing, fuzzing
- **Operational Evidence**: Monitoring, logging, incident response procedures
- **Certification Evidence**: Audits, standards compliance, regulatory approval

Assurance Case Structure



Evaluating Assurance

Evaluating Assurance Cases

How do we know whether an assurance case is convincing?

Key Evaluation Criteria

- **Completeness**: Are all critical claims and failure paths covered?
- **Soundness**: Do the arguments logically support the claims?
- **Evidence Quality**: Is the evidence reliable and relevant?
- **Assumptions**: Are assumptions reasonable and documented?
- **Confidence**: Is confidence appropriate for the level of risk?

Reviewing Assurance Cases

When reviewing an assurance case, what questions should we ask?

- Are all important **hazards** and **risks** addressed?
- Do the **arguments** support the claims?
- Is the **evidence** reliable and relevant?
- Are the **assumptions** clearly stated?
- Would the argument remain valid if conditions change?

Common Pitfalls in Assurance Cases

- **Ignoring Security Threats**: Assuming internal systems are safe
- **Underestimating Attack Likelihood**: Assuming attackers are not interested
- **Weak Evidence**: Relying on limited testing or vendor claims
- **Forgotten Assumptions**: Assumptions are undocumented or unsupported
- **Changing Threat Landscape**: New threats invalidate old analyses

Summary

- **Hazards** can arise from both safety failures and security threats
- **FMEA** identifies failure modes and associated risks
- **FTA** traces causes of undesired events using logic gates
- **HAZOP** uses guide words to identify deviations and hazards
- **Assurance cases** use claims, arguments, and evidence to justify safety and security
- **Evidence** should be reliable, relevant, and up to date
- **Assurance** must evolve as systems and threats change

Key References

- IEC 61508. (2010). *Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems*. **[Link]**
- ISO/IEC 31010. (2019). *Risk Management – Risk Assessment Techniques*. **[Link]**
- Leveson, N. (2011). *Engineering a Safer World: Systems Thinking Applied to Safety*. MIT Press. **[Link]**
- Adelard. Goal Structuring Notation (GSN). **[Link]**
- NASA. Fault Tree Handbook with Aerospace Applications. **[Link]**