IOT SECURITY

WHAT IS IOT ?

IoT

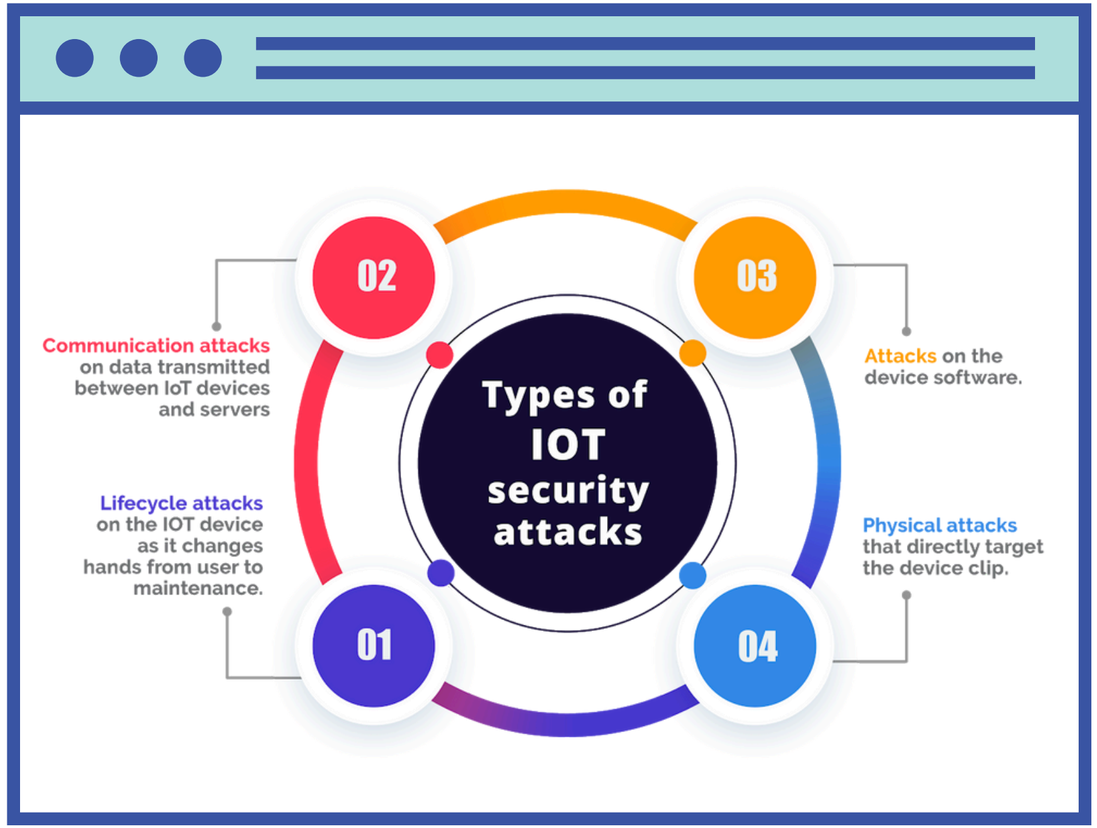
- The Internet of Things (IoT) is a network of physical devices that can transfer data to one another without human intervention.
- The term was first coined by computer scientist Kevin Ashton in 1999.

A Good Example of Iot is Your Smart Home Devices

WHAT IS IOT SECURITY ?

IoT

- IoT security is based on a cybersecurity strategy to protect IoT devices and the vulnerable networks they connect to from cyber attacks.
- IoT devices have no built-in security.
- In next Slides we will discuss IoT compromise in different Ecosystems

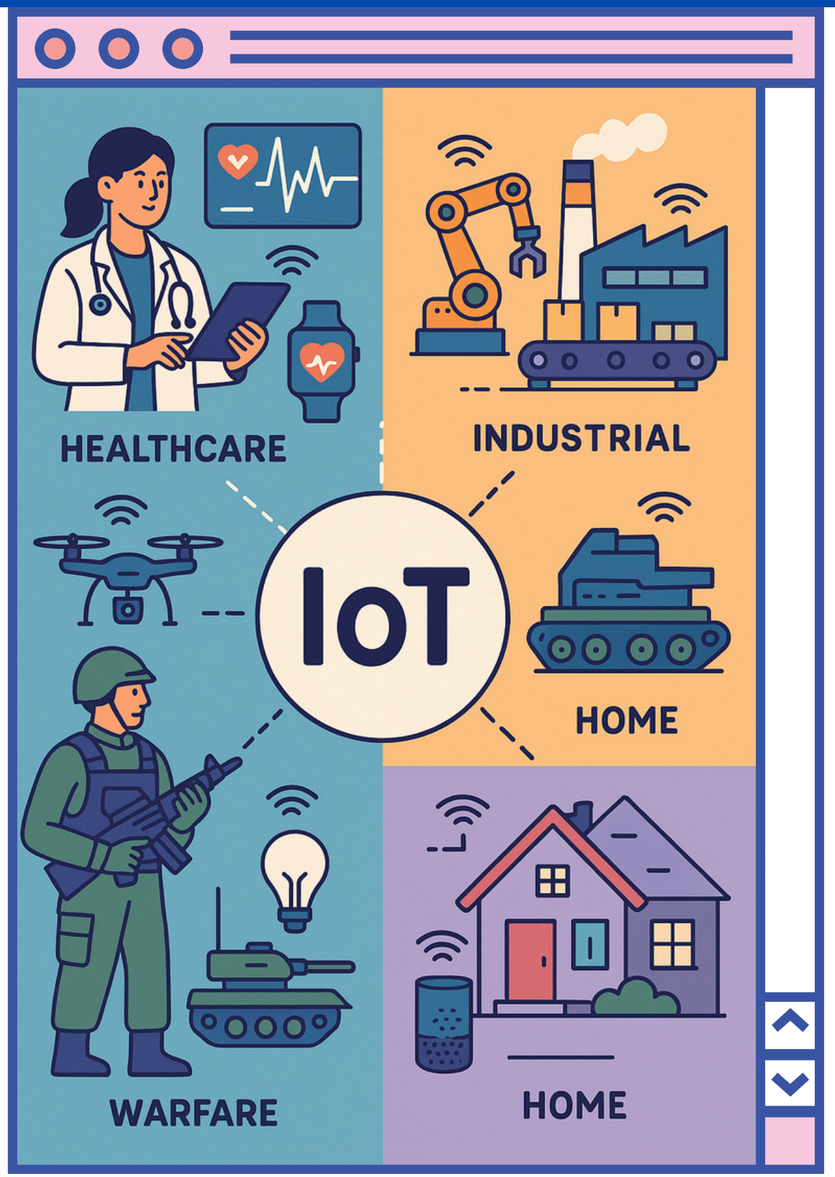


Types of Iot Security Attacks

IOT USED IN DIFFERENT SECTORS AND THEIR SECURITY ASPECTS

IOT USED IN VARIOUS SECTORS

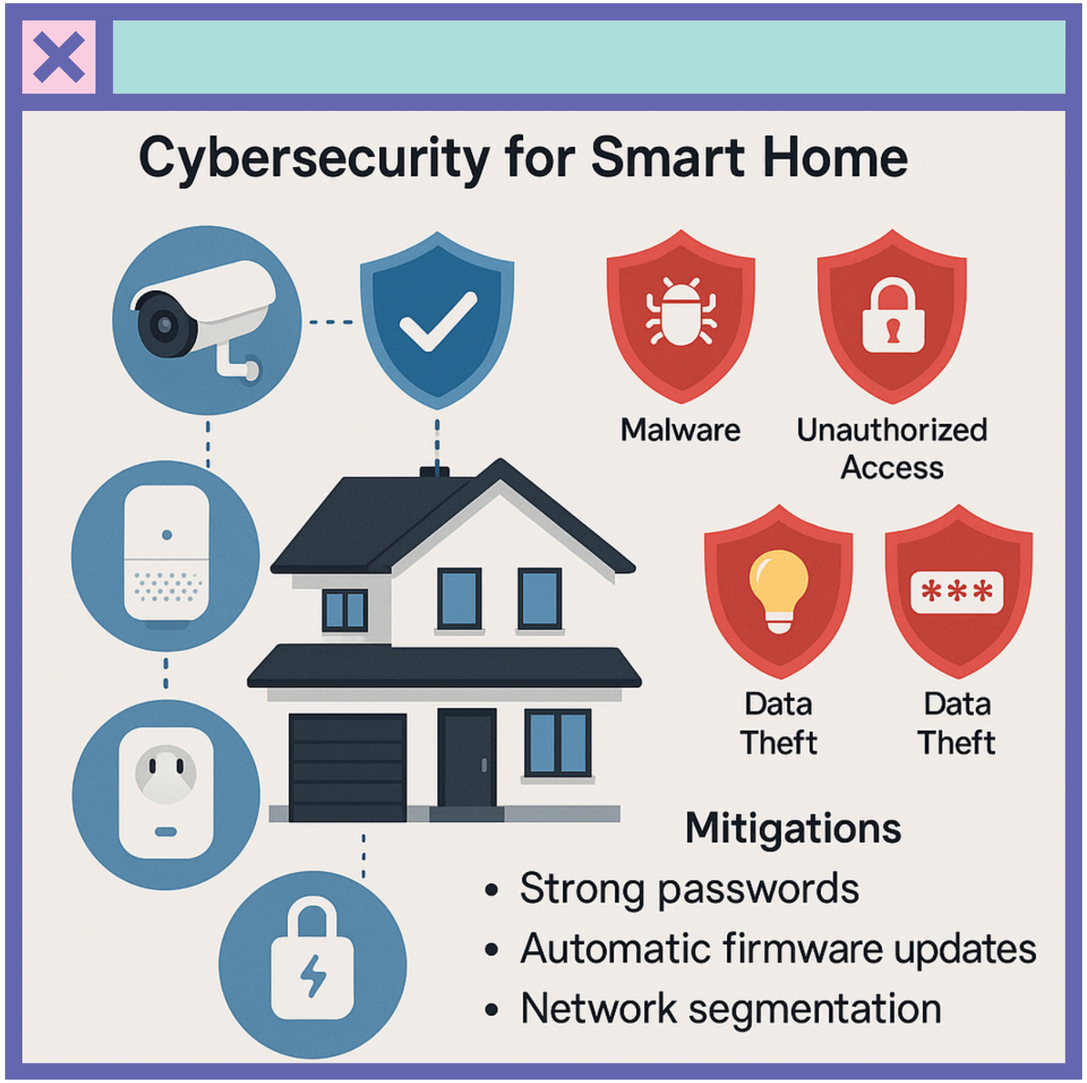
- The Internet of Things (IoT) is used in homes for smart devices that enhance convenience and energy efficiency
- In industries for automation and monitoring equipment
- In healthcare for remote patient monitoring and medical devices
- In warfare for advanced surveillance and communication systems.



IOT USED IN HOME

Scenario

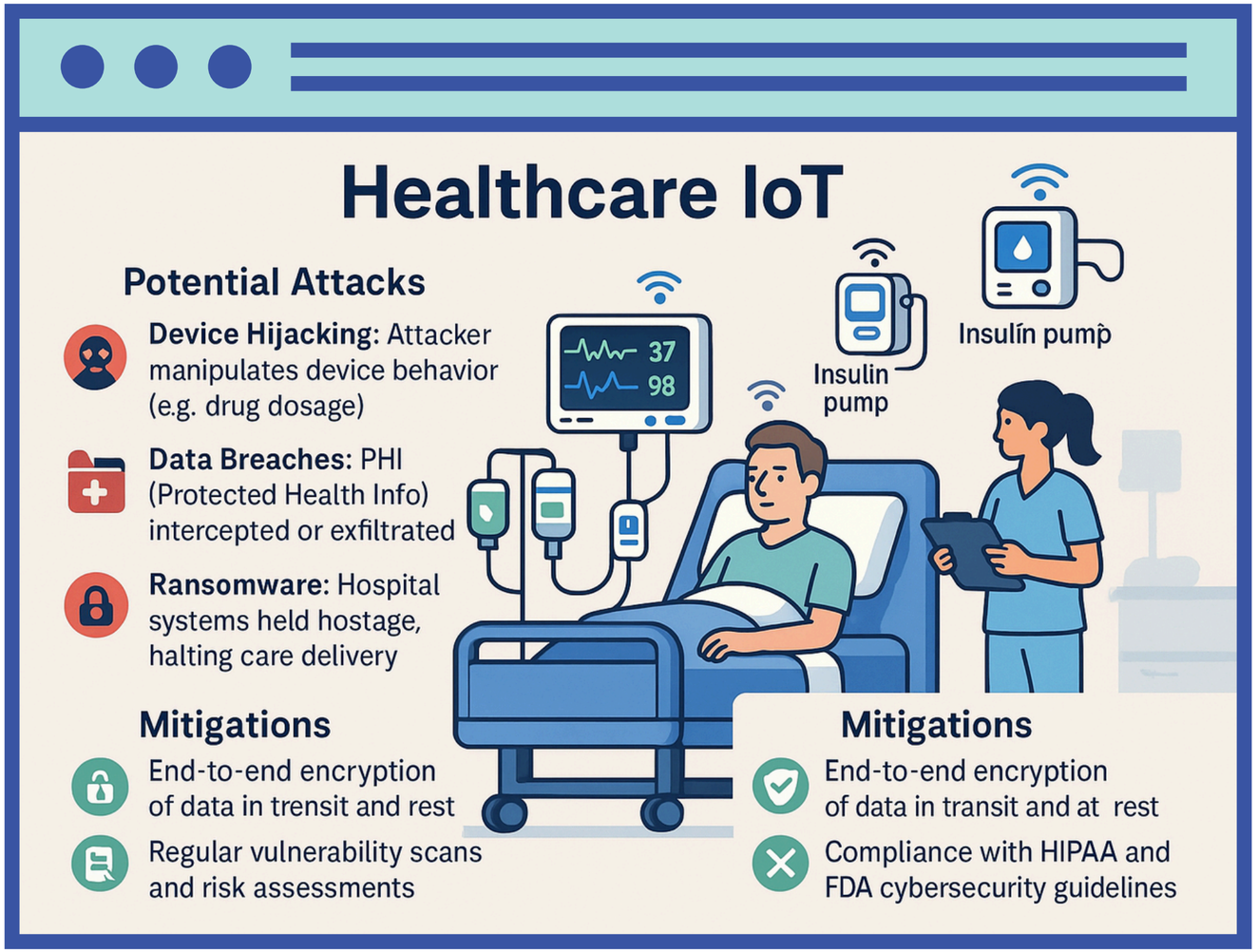
A smart home ecosystem includes IoT-enabled thermostats, smart lights, smart locks, voice assistants (e.g., Alexa), and CCTV systems that communicate over Wi-Fi and are managed via a mobile app.



IOT USED IN HEALTHCARE

Scenario:

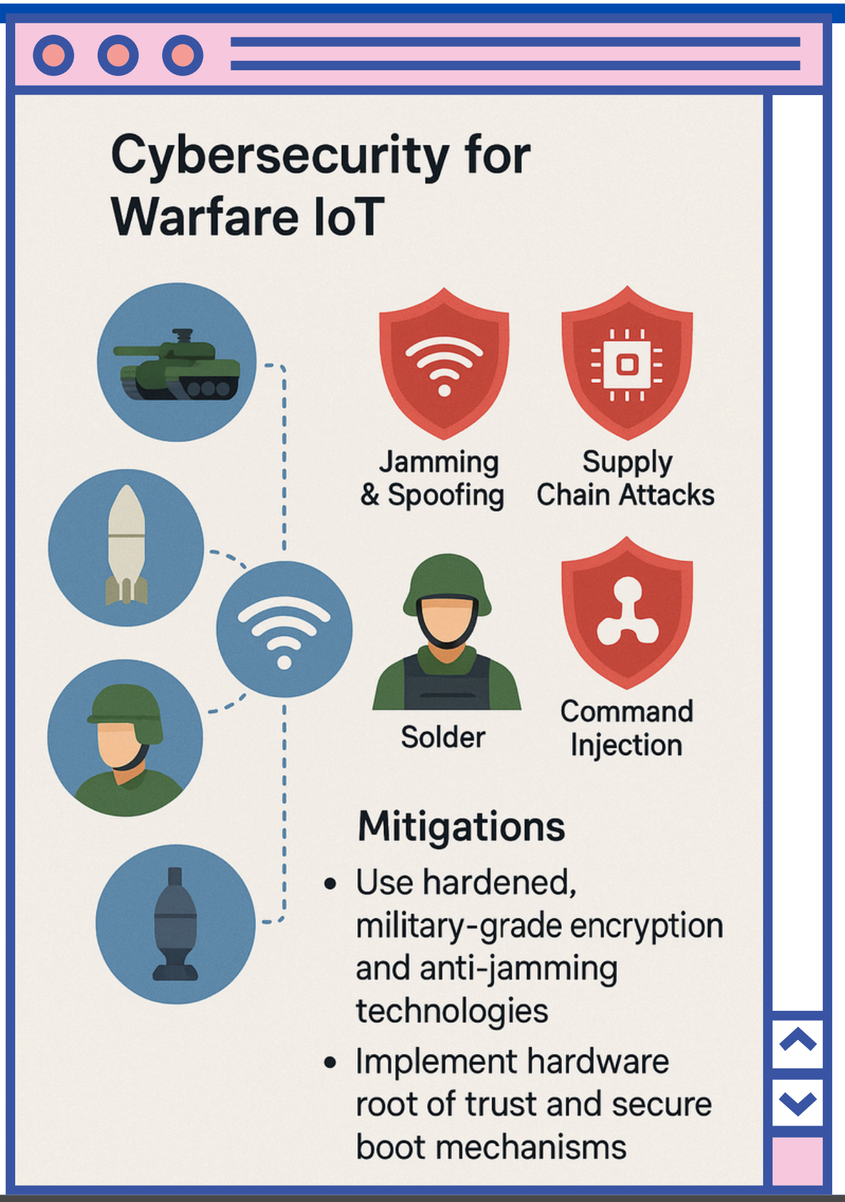
Hospitals use IoT for patient monitoring (heart rate, oxygen levels), insulin pumps, connected imaging machines, and smart infusion pumps.



IOT USED IN WARFARE

Scenario:

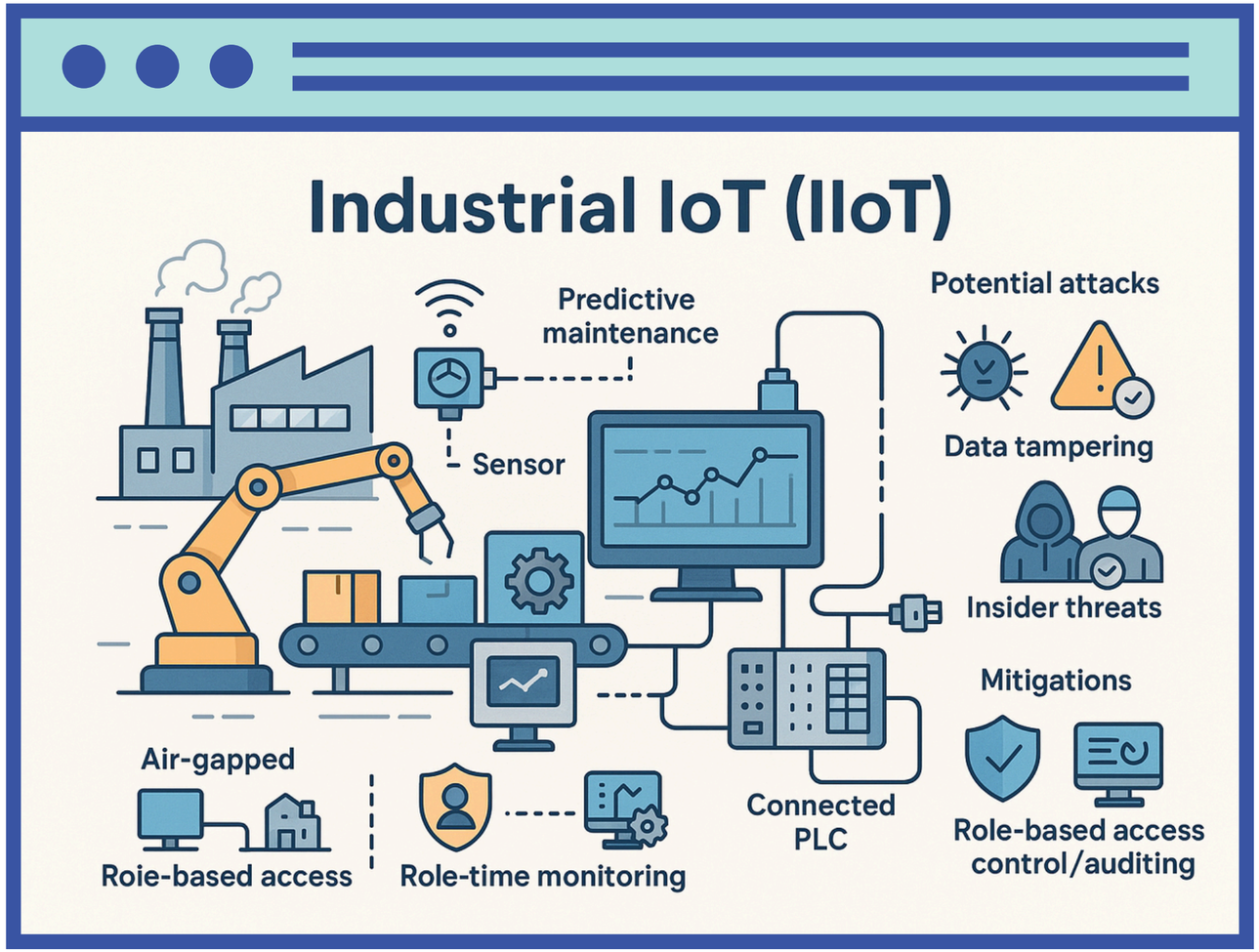
Modern battlefields deploy IoT-enabled drones, smart ammunition, autonomous tanks, soldier wearables, and remote surveillance systems.



IOT USED IN INDUSTRIY

Scenario:

A factory uses IoT for robotic automation, predictive maintenance sensors, SCADA systems, and connected PLCs (Programmable Logic Controllers).



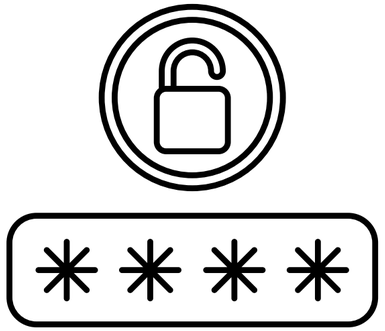
OWASP TOP 10 (IOT SECURITY)

OWASP TOP 10 (IOT)

A project designed to help manufacturers, developers, and consumers better understand the security issues associated with IoT and to enable users to make better security decisions when building, deploying, or assessing IoT technologies. *~ OWASP*

OWASP TOP 10 (IOT)

1. **Weak, Guessable, or Hardcoded Passwords**
   - Default or hardcoded passwords allow easy access.
   - Weak and Guessable Passwords are easy to brute.



2. **Insecure Network Services**
   - Exposed services can be remotely exploited.
   - Insecure endpoint and API are a threat.

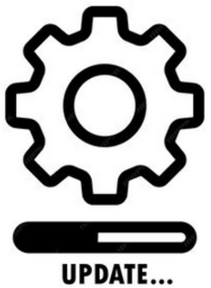
OWASP TOP 10 (IOT)

3. Insecure Ecosystem Interfaces

- Weak API/mobile/cloud interfaces can expose devices.
- Poor authentication mechanisms can be dangerous.

4. Lack of Secure Update Mechanism

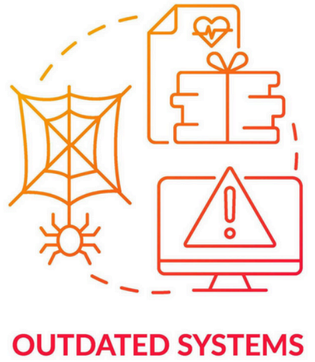
- Insecure or missing firmware updates or patches.
- Updates transmitted over insecure protocols (HTTP)



OWASP TOP 10 (IOT)

5. Use of Insecure or Outdated Components

- Outdated libraries or components can be exploited.
- Unpatched dependencies increase security risks.



6. Insufficient Privacy Protection

- Personal or Sensitive data exposure or misuse.
- Weak data handling may lead to privacy violations.

OWASP TOP 10 (IOT)

7. Insecure Data Transfer and Storage

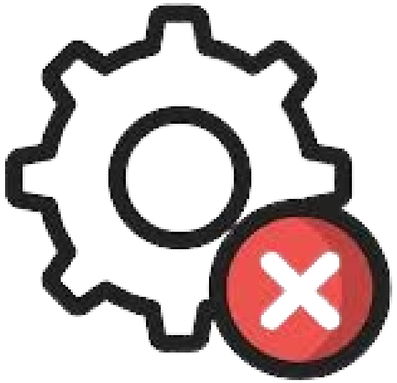
- Data not encrypted in transit or at rest.
- Weak protocols can expose sensitive information.

8. Lack of Device Management

- Poor and dis-organized asset or update management.
- No centralized control over devices or policies.

OWASP TOP 10 (IOT)

9. Insecure Default Settings



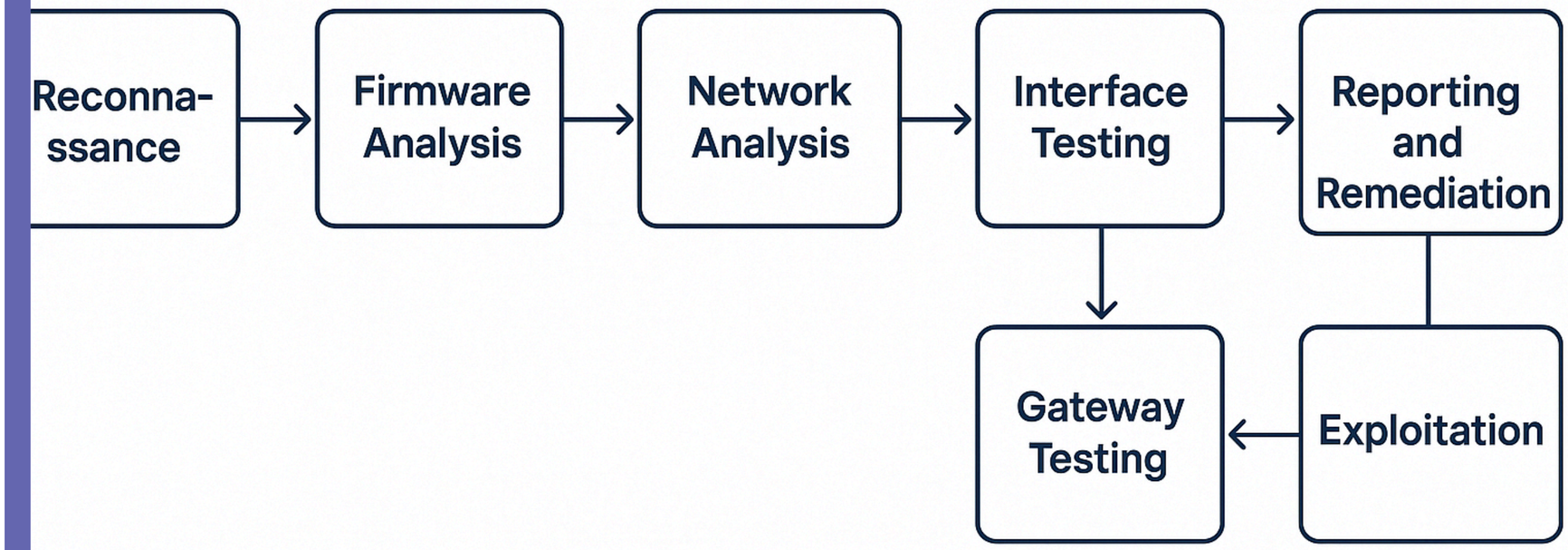
- Devices shipped with weak and insecure defaults.
- Default credentials/settings often go unchanged.

10. Lack of Physical Hardening

- Easy to tamper with the hardware/device
- No protection against physical access or theft.

IOT PENTESTING METHODOLOGY

IoT Pentesting Methodology



RECONNAISSANCE

- Gather information about the IoT system, including device types, communication protocols (e.g., MQTT, CoAP, or HTTP), and associated interfaces like mobile apps or APIs.
- Tools like Wireshark can be used to analyze network traffic, and techniques such as scanning for open ports (e.g., with Nmap) help map the system’s architecture.

FIRMWARE ANALYSIS

- Extract and analyze the firmware from IoT devices.
- Static Analysis: Use tools like Binwalk or Ghidra to inspect the firmware for hardcoded credentials, insecure configurations, or cryptographic flaws.
- Dynamic Analysis: Emulate the firmware using tools like QEMU to observe runtime behavior, identify vulnerabilities, or detect malware.

NETWORK ANALYSIS

- Examine communication between devices, gateways, and cloud systems.
- Identify weaknesses like unencrypted data transmission, weak protocols (e.g., HTTP instead of HTTPS), or default credentials in use.
- Simulate attacks like Man-in-the-Middle (MITM) or DoS to test the resilience of the system’s communication layer.

INTERFACE TESTING

- Evaluate user-facing components like web dashboards, mobile apps, or APIs for vulnerabilities.
- Test for weak authentication mechanisms, session management flaws, or injection vulnerabilities (e.g., SQL injection, XSS).
- Use tools like Burp Suite to test APIs and web interfaces for exploitable issues.

GATEWAY TESTING

- Analyze gateways for protocol exploits, weak encryption, or improper access control.
- Test edge computing processes for vulnerabilities that attackers could exploit to compromise local processing.
- Simulate attacks on protocol translations (e.g., MQTT or CoAP) to check for unauthorized access or data leakage.

POSTMAN

EXPLOITATION

- Attempt to exploit identified vulnerabilities in devices, communication, interfaces, or cloud services.
- Examples include gaining shell access on a device, bypassing authentication mechanisms, or injecting malicious payloads into APIs.
- Tools like Metasploit or custom scripts are often used in this step.

REPORTING AND REMEDIATION

- Document all findings, including identified vulnerabilities, exploitation methods, and their potential impact on the system.
- Provide recommendations for mitigating risks, such as applying patches, enforcing encryption, or improving authentication practices. Ensure actionable steps for developers and system administrators to fix security gaps.

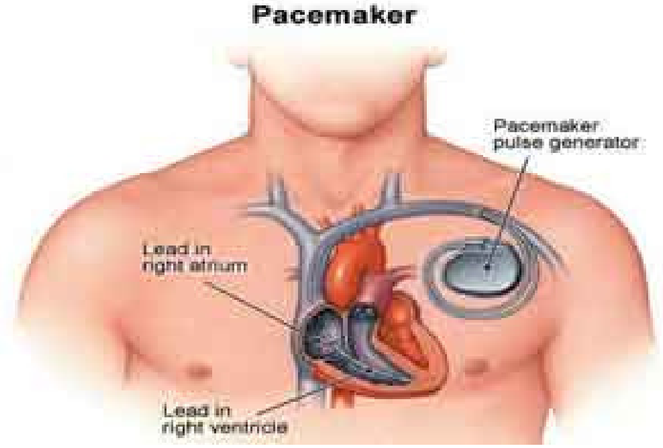
SOME POPULAR IOT ATTACKS

MIRAI BOTNET ATTACK (DYN ATTACK)

- Distributed Denial-of-Service (DDoS) attack that occurred on October 21, 2016.
- Mirai malware scanned the internet for vulnerable IoT devices
- Once it found an insecure device, it logged in and infected it, turning it into a bot
- The attacker remotely commanded the botnet to flood Dyn’s DNS servers with massive amounts of junk traffic
- By targeting **Dyn (a DNS provider)**, the attack had a cascading effect, taking down multiple major websites at once

THE HACKABLE CARDIAC DEVICES

- St. Jude’s devices used a home transmitter to remotely send patients data to doctor.
- This transmitter communicated via RF signals and Wi-Fi, but had no encryption or authentication.
- Hackers could **intercept wireless signals** between the implant and transmitter.
- Battery drain attacks, Malicious Shock Command, Data Theft.



THE OWLET WIFI BABY HEART MONITOR VULNS

- Popular wearable device that tracks infants' heart rate and oxygen levels
- The monitor transmitted sensitive baby health data (heart rate, oxygen levels) in plaintext over WiFi
- If the sensor sock came off (e.g., baby kicked it loose), the monitor did not automatically restart
- The device lacked a secure update mechanism, meaning security flaws could not be patched easily.
- MITM attacks, DOS attacks, WIFI jamming.

TOOLS USED IN IOT SECURITY

ATTIFY OS

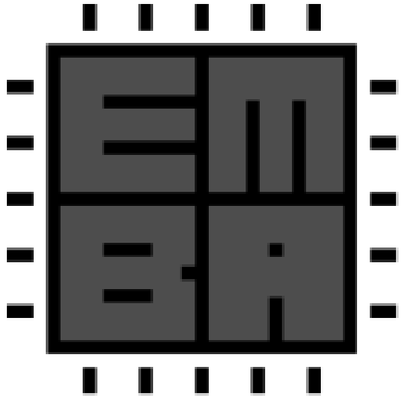
- Opensoure Linux-based penetration testing distribution specifically designed for IoT and embedded device security assessments.
- Contains pre-installed iot security tools
- Support for Hardware Interface like JTAG, UART etc
- Contains automated firmware extraction scripts.

FIRMADYNE

- Firmadyne is an open-source framework designed for automated emulation and vulnerability analysis of IoT firmware
- Emulate firmware in a virtualized custom based QEMU environment.
- Detect vulnerabilities (e.g., backdoors, hardcoded credentials).
- Test exploits without needing physical hardware.
- **D-Link firmware backdoor** (CVE-2019-16920) was discovered using Firmadyne.
- Found hardcoded admin credentials.

EMBEDDED ANALYZER

- **EMBA** is an **open-source firmware security analyzer** specifically designed for **IoT and embedded devices**.
- It automates firmware extraction, static vulnerability scanning, dynamic emulation-based analysis and compliance checking
- Developed as a successor to tools like **Firmadyne**



THANK YOU