IoT overview

IoT basics

- Internet of things
- Extends Internet connectivity beyond standard devices to everyday objects
- Usually uses IPv6 due to the limited number of IPv4 addresses
- Operating systems: Linux or Windows (10) IoT

Top-level components

- **Device**
  - Includes hardware and software that directly interact with the world.
  - They connect to a network to communicate with each other, or to centralized applications
- **Gateway**
  - Enables services to reach cloud services.
  - Infrastructure component providing security and protocol translations
  - Also used as a service that process data on behalf of group or cluster devices.
  - Often a device e.g. smart home hub.
  - Usually from the same vendor
- **Cloud**
  - See cloud computing
- **Sensors**
  - Detects, measures or indicates any specific physical quantity
  - E.g. light, heat, motion, moisture, pressure, or similar entities
  - Converts them into any other form which is mostly, electrical pulses.

IoT communication models

Device-To-Device (D2D)

- Direct communication between devices
- Uses a medium such as Bluetooth Low Energy etc.
- Common in home automation systems e.g. light bulbs or wearables e.g. smart watch and heart monitor.
- Simpler security
- E.g. **Vehicle-to-vehicle (V2V)**
  - Uses Vehicle Ad Hoc Network (VANET)
    - Based on MANET i.e. decentralized wireless network (without routers)

Device-To-Cloud (D2C)

- IoT device directly communicating with the cloud server
- Often uses ethernet or WiFi
- Lets the user (and an application) to obtain remote access to a device
- E.g. smart card for dogs, remote monitoring
- Two credentials:
    - the network access credentials (such as the mobile device's SIM card)
    - credentials for cloud access
- E.g. Nest Learning Thermostat

Device-To-Gateway (DTG)

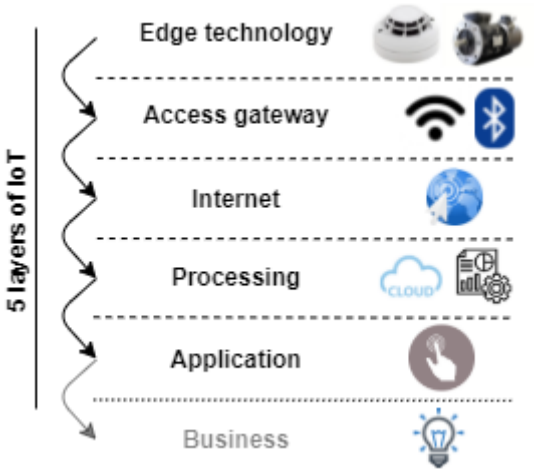
- IoT devices basically connect to an intermediary device to access a cloud service
- Often includes an application software operating on a local gateway device (like a smartphone or a "hub")
- Gateway provides security, protocol translation and usually does aggregation
- E.g. Samsung SmartThing ecosystem

Back-end data-sharing

- Extends device-to-cloud model
- Access are granted to the uploaded data to third-parties
- E.g. Map My Fitness that compiles data from other applications

Layered architecture

- IoT architecture can be categorized into different layers.
- There's no consistency regarding naming of layer.
    - Different methodologies are used but the concepts they represent are very similar.
- 📝 It usually consists of 5 layers:
    1. Edge technology layer the "IoT objects collecting data"
    2. Access gateway layer the "data transporter"
    3. Internet layer the "endpoint connector"
    4. Middleware layer the "data analyzer and processor"
    5. Application layer the "user interface"
- Some sources also name sixth layer:
    6. Business layer the "core logic"
- Each layer is utilized by layer below without knowledge of other layers
-



- Read more: [IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey]

Five-layers of IoT architecture

Edge technology layer

- Also known as **perception layer** or **hardware layer**
- Physical objects (hardware components)
  - Covers IoT capable devices
  - E.g. sensors, actuators, heat sensor, RFID tags, readers, device itself
- Connects devices within network and server
- Gathers environment data
- **Key security components**
  - Encryption and key agreement
  - Sensor data protection
- **Vulnerabilities**
  - Eavesdropping: real time attack to intercept privacy communications.
  - Node Capture: capturing a key node such as gateway to reveal information.
  - Fake Node and Malicious: adding node to input fake data to stop transmitting real information
  - Replay (play back) attack: eavesdrops a communication and reusing it to authenticate.
  - Timing Attack: Extract secrets by observing respond time

Access gateway layer

- Also known as **network layer** or **transport layer**
- Handles data transmission i.e. transferring the data through network
- E.g. Wi-Fi, bluetooth
- Enables communication
  - Connects two endpoints e.g. a clients with a device.
  - Includes the initial data handling.
  - Through e.g. message routing, message identification, and subscriptions.
- **Key security components**
  - Encryption
  - Identity authentication
- **Vulnerabilities**

    - Denial of Service (DoS) Attack with redundant requests
    - Main-in-The-Middle (MiTM) Attack: to intercept and manipulate data in real-time
    - Storage Attack: Changing data stored in device or cloud
    - Exploit attack: Exploits vulnerabilities in an application, system or hardware

Internet layer

- Responsible for end-points connectivity.
- Carries out communication between two endpoints.
    - E.g. device-to-device, device-to-cloud, device-to-gateway and back-end data-sharing.

Middleware layer

- Also known as **processing layer**
- Responsible for device and information management.
- Handles data analytics
    - I.e. storing, processing and analysis of data.
    - E.g. data analysis, data aggregation, data filtering, device information discovery, and access control.
- Behaves as interface for two-way communication between
    - Application layer (the user interface).
    - Edge technology layer (the hardware).
- **Key security components**
    - Key security layer, secure cloud computing, antivirus
- **Vulnerabilities**
    - Exhaustion: Can disturb memory, battery e.g. after effect of a DoS
    - Malware

Application layer

- The user interface for
    - Graphic data representation
    - Controlling, managing and commanding IoT devices.
- Responsible for delivering *service* and *data* to users.
    - A service is application-specific e.g. industrial, manufacturing, automobile, security, healthcare...
- **Key security components**
    - Authentication
    - Key agreement
- **Vulnerabilities**
    - Cross site scripting: injecting code through e.g. JavaScript
    - Malicious code attack: can activate itself or require user attention to perform an action.
    - Dealing with Mass Data
        - Caused by massive amount of data transmission
        - Can lead to data loss and network disturbance

Other IoT layers

Business layer

- Includes business models
- System management
- **Key security components**
  - Privacy protection
- **Vulnerabilities**
  - Business logic attack: exploits a programming flaw
  - Zero-day attack: exploits security hole unknown to the vendor

IoT connectivity

Wireless IoT connectivity

Approx. range up to	Connectivity	Speed
10 cm	NFC	424 kbit/s
1 m	RFID	300 tags per second
10 m	Li-Fi	100 gbit/s
60 m	Bluetooth low energi (BLE)	1 or 2 mbit/s
100 m	WiFi	1300 mbit/s
1 km	Wi-Fi HaLow	78 mbit/s
2 km	5G	20 gbit/s
30 km	LTE-Advanced	300 mbit/s
70 km	Celullar	- (depends on 4g etc.)
1000 km	LPWAN	200 kbit/s
World-wide	VSAT	16 mbit/s

Short-range wireless communication

- **Bluetooth Low Energy (BLE)**
  - Newer versions of bluetooth (after 4.0)
  - Optimized for battery usage.
- **Wi-Fi**
  - Wireless network protocol using radio waves.
  - Wi-Fi 6 specification standard (2020) is the latest standard (x6 faster).
- **Radio-Frequency Identification (RFID)**
  - Data storage tag that can be attached to an item for tracking
  - Passive tag has range up to 1m while active tags can go up to 100m.
  - Used in e.g. passports, credit cards.
- **Li-Fi (Light-Fidelity)**

    - Similar to Wi-Fi, but using visible light for communication
- **Near-Field Communication (NFC)**
    - Based on a radio frequency (RF)
    - Used e.g. in phones, payment cards
    - Must either either physically touch or be in a few centimeters of each other.

Medium-Range Wireless Communication

- **LTE-Advanced**: Formally submitted as a candidate 4G, often being described as 3.9G.
- **Wi-Fi HaLow**: low power, long-range, also known as "WiFi for Internet of Things"
- **5G**: Introduced in 2019, highest with minimum of 10 Gbps

Long Range Wireless Communication

- **Low-Power Wild-Area Network (LPWAN)**
    - Long range communication (up to 10 km) at a low bit rate
- **(VSAT) Very Small Aperture Terminal**
    - World-wide satellite communication technology uses small dish antennas
- **Cellular** using e.g. radio towers to spread e.g. 4G, 5G..

Wired IoT connectivity

- **Ethernet** (cat 6 up to 10 Gbps speed)
- **Power-Line Communication (PLC)** : using electrical wiring to carry power and data, around 200 Mbit/s.

IoT security

IoT threats

- **Lack of security**
  - Speed at which IoT is advancing makes it harder to keep up with evolving security requirements.
  - Being short on processing power and memory leads to lack of security solutions and encryption protocols.
- **Vulnerable interfaces**
  - For both device interfaces and other interfaces (e.g. cloud) it interacts with
  - E.g. lack of authentication/authorization, lacking or weak encryption, and a lack of input and output filtering.
- **Physical security risk**
  - Cannot secure them as traditional devices by e.g. the storage of routers in secure cabinets
- **Lack of vendor support**
  - The support of a certain device may get discontinued
- **Difficult to update firmware and OS**
  - Some require manual intervention to be upgraded, some cannot be upgraded at all
  - Being compliant makes harder to do changes to e.g. medical devices.
- **Interoperability issues**
  - Interoperability: "the ability to make systems and organizations work together" | Wikipedia
  - Each solution provides its own IoT infrastructure, devices, APIs, and data formats
  - Caused by competitive nature of IoT e.g. vendor lock-in

OWASP Top 10 IoT (2018)

- OWASP Internet of Things Top Ten was introduced in 2004 and updated in 2018

1. **Weak, guessable, or hardcoded passwords**
   - Use of easily brute forced, publicly available, or unchangeable credentials
   - Including backdoors in firmware or client software that grants unauthorized access to deployed systems
2. **Insecure network services**
   - Unneeded or insecure network services running on the device itself
   - Bigger threat for those that are expose to the internet
   - Allows compromise confidentiality, integrity/authenticity, or availability of information or allow unauthorized remote control...
3. **Insecure ecosystem interfaces**
   - Includes web, backend API, cloud, or mobile interfaces outside of the device
   - Allows compromise of the device or its related components.
   - E.g. lack of authentication/authorization, lacking or weak encryption, a lack of input and output filtering.
4. **Lack of secure update mechanism**
   - Lack of firmware validation on device

    - Lack of secure delivery (un-encrypted in transit)
    - Lack of anti-rollback mechanisms
    - Lack of notifications of security changes due to updates.
5. **Use of insecure or outdated components**
    - Use of deprecated or insecure software components/libraries
    - Insecure customization of operating system platforms
    - Use of third-party software or hardware components from a compromised supply chain
6. **Insufficient privacy protection**
    - Use of users personal information insecurely, improperly, or without permission.
7. **Insecure data transfer and storage**
    - Lack of encryption or access control of sensitive data
    - Can be anywhere within the ecosystem e.g. at rest, in transit, or during processing.
8. **Lack of device management**
    - Lack of security support on devices deployed in production
    - Capabilities include e.g. asset management, update management, secure decommissioning, systems monitoring, and response.
9. **Insecure default settings**
    - Can be shipped with insecure settings or without ability to make restrictions.
10. **Lack of physical hardening**
    - Easily accessible physically

IoT attacks

IoT attack surface areas

- **Device memory**: Credentials
- **Ecosystem access control**: Implicit trust between components
- **Device physical interfaces**: Privilege escalation, CLI
- **Device web interface**: SQL injection, XSS
- **Device firmware**: Sensitive data exposure, hardcoded credentials
- **Device network services**: Unencrypted/poorly encrypted services.
- **Administrative interface**: SQL Injection, XSS
- **Local data storage**: Data encrypted with discovered keys, lack of integrity checks.

IoT attack types

- **Access control**
    - E.g. remote access control or gaining access to administration panels
- **BlueBorn Attack**
    - Amalgamation of techniques and attacks against known, already existing Bluetooth vulnerabilities
- **Jamming Attack**
    - Also known as **signal jamming attack**
    - Jamming the signal to prevent the communication of devices
- **Man-in-the-middle attack**
    - E.g. by sniffing through Foren6
        - Passive sniffer
        - Reconstruct a visual and textual representation of network information to support real-world Internet of Thingl

- **HVAC attack**
  - Takes place when one hacks IoT devices in order to shut down air conditioning services.
  - Can allow access to a corporate systems.
- Backdoor (not just IoT related)
- Exploit kits
  - Malicious scripts used to exploit poorly patched devices.
- Replay attack
  - Attackers send intercepted messages to target device to perform DoS.
  - See also SDR-based attacks
- Ransomware attack
  - Type of malware that uses encryption to block user's access to his/her device.
- Privilege escalation
- Side channel attack
  - Attackers extract info about encryption keys by observing the emission signals (side channels) from IoT devices.
- Web application attacks, web server attacks
- Cloud computing attacks
- Mobile application threats
- DoS / DDoS
  - Can be done by converting devices into an army of botnet.
- **Forged malicious devices**
  - Attackers replace authentic IoT devices with malicious device.
- Resetting to an insecure state
- Removal of storage media
- Firmware attack
- Network service attacks
- Unencrypted local data storage
- Confidentiality and integrity issues
- Malicious updates
- Insecure APIs
- Eavesdropping
- **Sybil attack**
  - Attacker uses multiple forged identities to create strong illusion of traffic congestion.

Rolling code attack

- Also known as **hopping code** attack.
- Used in keyless entry systems such as garage door openers and keyless car entry systems.
- Attacker capture signal from transmitter device, simultaneously blocking the receiver to receive the signal
- Attacker uses the signal to gain unauthorized access
- E.g. stealing car with captured signal
  - Attacker jams and sniffs the signal to obtain the code transferred to vehicle's receiver

- Tools include HackRF One hardware tool.

SDR-based Attacks

- Attackers use Software Defined Radio (SDR) to examine the communication signals in the IoT network and sends spam content or texts to the interconnected devices.
- Can also change the transmission and reception of signals between the devices.
- Includes
    - **Replay attack**
        - The attacker obtains frequency used for data sharing between devices and captures data.
    - **Cryptanalysis Attack**
        - Attacker uses same procedure as replay attack and also reverse engineering of the protocol to capture the original signal.
    - **Reconnaissance attack**
        - Attacker obtains info about the target device from the device's specification.
        - See also information gathering

Firmware extraction

- Allows looking for data in filesystem or reverse engineering it for vulnerabilities.
- Flow example:
    1. `binwalk` is a common tool for it found on Kali Linux.
    2. `firmwalker` to list vulnerabilities by scanning all files.

Device memory containing credentials

- Can be used for reading/manipulating data
- Allows pushing firmware updates
- Enables usage of devices to other devices in the network

Fault injection attacks

- Also known as **perturbation attacks**
- Occur when a perpetrator injects any faulty or malicious program into the system to compromise the system security.
- **Optical, Electro Magnetic Fault Injection (EMFI), Body Bias Injection (BBI)**
    - Injection using projecting lasers and electromagnetic pulses.
- **Power/clock/reset/glitching**
    - Injections into power supply and clock network of the chip.
- **Frequency/voltage tampering**
    - Tampering with clock frequency of the chip
- **Temperature attacks**
    - Attackers alter the temp for the operating the chip.

DNS rebinding

- Done by compromising browsers as traffic tunnels to exploit private services.
- Done through malicious script in a webpage to manipulate resolution of domain names.
- Can help to gain access over the target's router using a malicious JavaScript code injected on a web page.
    - After that, an attacker can assault any device activated using the default password.

Hacking Methodology

Information gathering

- Also known as **IoT footprinting**
- Includes collecting information regarding target IoT devices
- Information can include e.g. IP address,running protocols, vendor, type of device, hostname, ISP, device location, banner of the target IoT device.
- Can involve using
    - IoT search engines to find manufacturer or device information.
    - Searching for hardware registrations in regulating bodies
        - Can help to find information regarding compliance standards, user Manuals, documentation, wireless operating frequency, and photos
        - E.g.
            - FCC ID search by "United States Federal Communications Commission registry"
            - IC ID Search by "Industry Canada (IC)"
            - KCC identifier search by Korean Communications Commission
            - CMII/CMIIT search by China Ministry of Industry and Information Technology
- See also Footprinting

Vulnerability scanning

- Scanning the network and devices to find vulnerabilities
- Search for weak password
- Software and firmware vulnerabilities
- Tools
    - `nmap`
    - `hping`
    - Firmalyzer
        - Security assessments with risk analysis in IoT networks
        - Proprietary platform

Attack

- Exploiting vulnerabilities
- E.g. running rolling code attack

Gain access

- Gain unauthorized access
- Privilege escalation
- Install backdoor

Maintain attack

- Logging out
- Clearing logs
- Covering tracks

IoT attack countermeasures

- **Encrypt**
  - Use encrypted communication (SSL/TLS)
  - Implement end-to-end encryption
  - Use VPN architecture
  - Encrypt drives
- **Password policies**
  - Use strong password
  - Ensure secure password recovery
- **Update devices**
  - Patch vulnerabilities
  - Firmware update
- **Restrict access**
  - Prevent the devices against physical tampering
  - Allow only trusted IP's to access device from internet
  - Implement strong authentication mechanisms.
    - E.g. two-Factor Authentication
  - Use Lockout feature to disable multiple login attempts
- **Monitor**
  - Implement IPS/IDS in the network
  - Periodic assessment of devices
- **Disable unused or unnecessary ports and services**
  - Disable UPnP port on routers
  - Monitor traffic on port 48101 for infected traffic
  - Disable telnet as it's insecure protocol
  - Disable Guest or Demo user accounts