Security in the Internet of Things (IoT)

Mobile Systems Security 2016

Optional lecture

March 22nd, 2016

Aalto University

UNIVERSITY OF HELSINKI

What is the Internet of Things ?

Latest evolution of the Internet: machine to machine (or device to device) communications

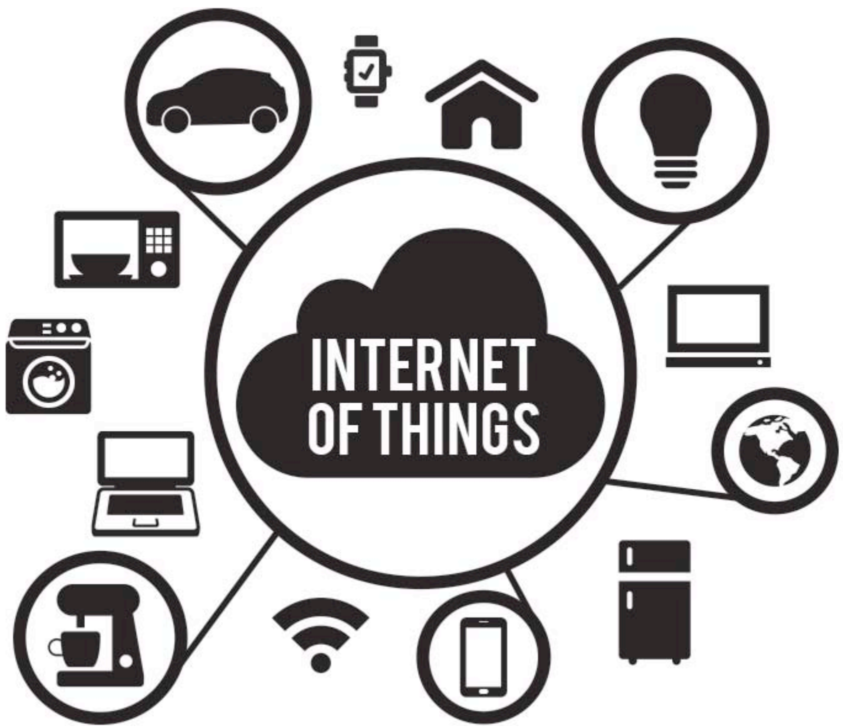
Contents (WWW)

⬇

Services (WEB 2.0)

⬇

People (Social Media)

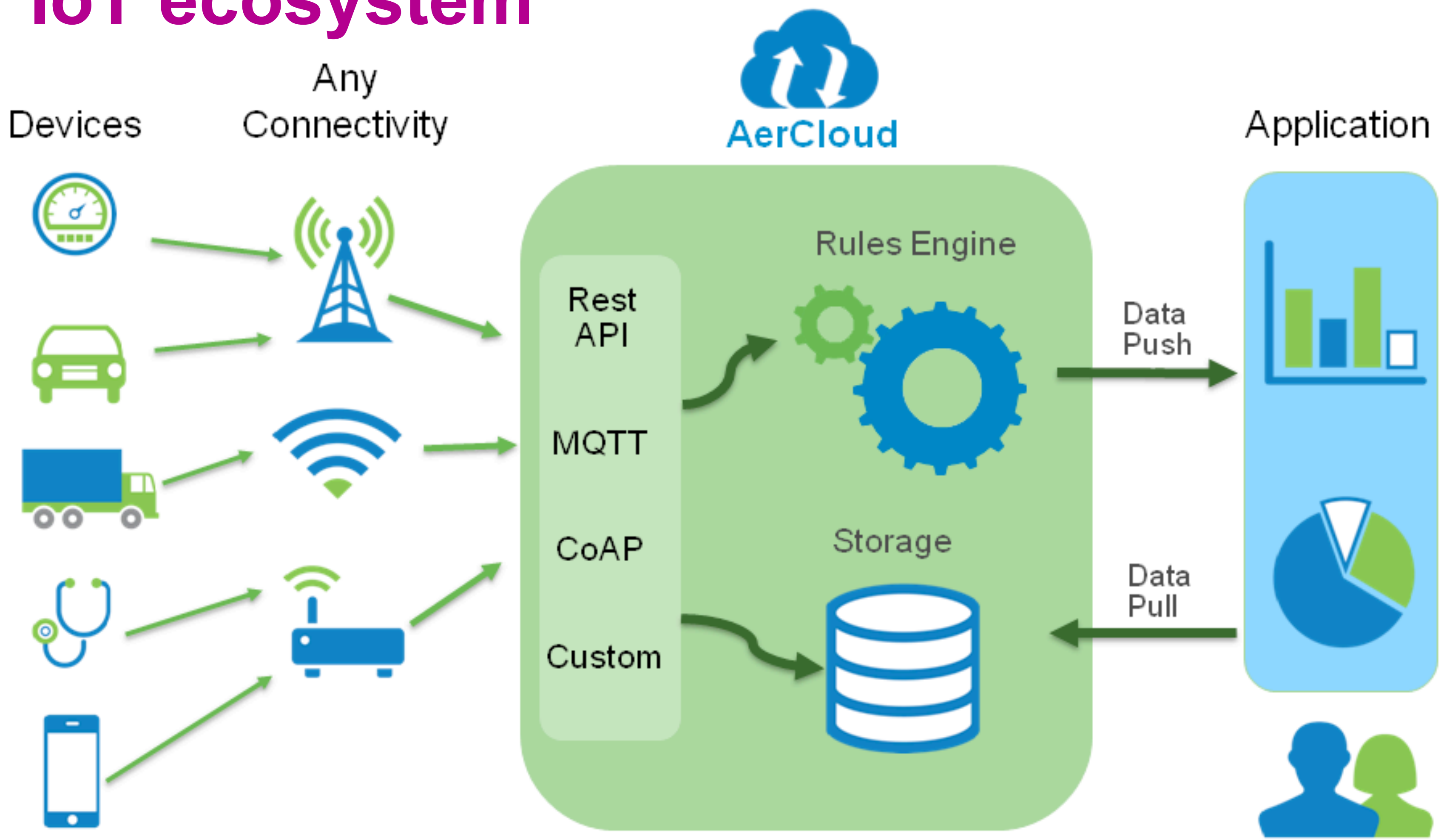


Definition

- **Wikipedia:** “Network of physical objects, devices, vehicles, buildings and other items embedded with electronics, software, sensors, and network connectivity that enables these objects to collect and exchange data.”

- **Gartner:** “Network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment.”

IoT ecosystem



[1] http://www.aeris.com/technology/aercloud/

Aalto University

UNIVERSITY OF HELSINKI

IoT characteristics

- Entity heterogeneity (3 tiers)[1]:
  - High-end devices (laptop, smartphone, tablets)
  - Low-end devices (sensors, actuators)
  - Passive entities (barcode, QR-code, RFID)
- Communication heterogeneity:
  - Wired communications (ethernet)
  - WiFi / 3G / 4G
  - Bluetooth (LE) / Zigbee / 6LoWPAN
- Highly personal data
- Device manufacturers are not security expert

[1] Corvington and Carskadden "Threat Implications of the Internet of Things", 2013

Node Constraints [1]

- Maximum code complexity (ROM/Flash)
- Size of state buffers (RAM)
- Amount of computation feasible in a period of time (processing capabilities)
- Available power
- User interface and accessibility in deployment (set keys, update software)

[1] RFC7228 “Terminology for Constrained-Node Networks” (https://tools.ietf.org/html/rfc7228)

Network Constraints [1]

- Low achievable throughput
- High packet loss
- Asymmetric link characteristics
- Penalties for using large packets (e.g. high packet loss due to link layer fragmentation)
- Reachability over time (wake-up and sleeping time of devices)
- Lack of advance services (e.g. IP multicast)

[1] RFC7228 “Terminology for Constrained-Node Networks” (https://tools.ietf.org/html/rfc7228)

Securing the IoT

- System (access control, authentication)
- Application
- Mobile
- Cloud
- Network (communications)

Approaches

1. Threat analysis (e.g. RFC 3552)

2. Follow security recommendation (e.g. NIST, IETF, etc.)

3. Learn from attacks

4. Follow Design Patterns

1. Threat Analysis

- **Assumption:** attacker has nearly complete control of the communications channel
- **Scenarios:**
  - Active vs. passive attacker
  - On-path vs. off-path
- Risk Analysis ➡ Security requirements
- Fulfill requirements:
  - Authentication
  - Authorization
  - Traffic Security (confidentiality, data-origin, integrity, availability)
  - Non-repudiation (optional)

Threat Analysis: Limitations

- Gives theoretic security requirements to meet
- **But:** leaves room for interpretation in implementation
  - Which layer to apply security protection ?
  - Which existing security frameworks to use ?
- Complex to perform
  - Consideration of vulnerable devices used as attack vector

2. Security recommendations (e.g. NIST, IETF)

- Key management requirements [1]
- Key length recommendations [2]
- Randomness requirements [3]
- Avoid possibility of pervasive monitoring
- Protocol or domain specific recommendation (crypto algorithm, WLAN security, use of TLS / DTLS, etc.)

[1] RFC 4107 “Guidelines for Cryptographic Key Management” (https://tools.ietf.org/html/rfc4107)
[2] RFC 4492 “Elliptic Curve Cryptography Cipher Suites for TLS” (https://tools.ietf.org/html/rfc4492)
[3] RFC 4086 “Randomness Requirements for Security” (https://tools.ietf.org/html/rfc4086)

3. Learn form Attacks

Selected attacks to illustrate common problems:

- Inadequate software update mechanism
- Missing Key Management
- Insecure configuration files and default passwords
- Missing communication security
- Physical attacks

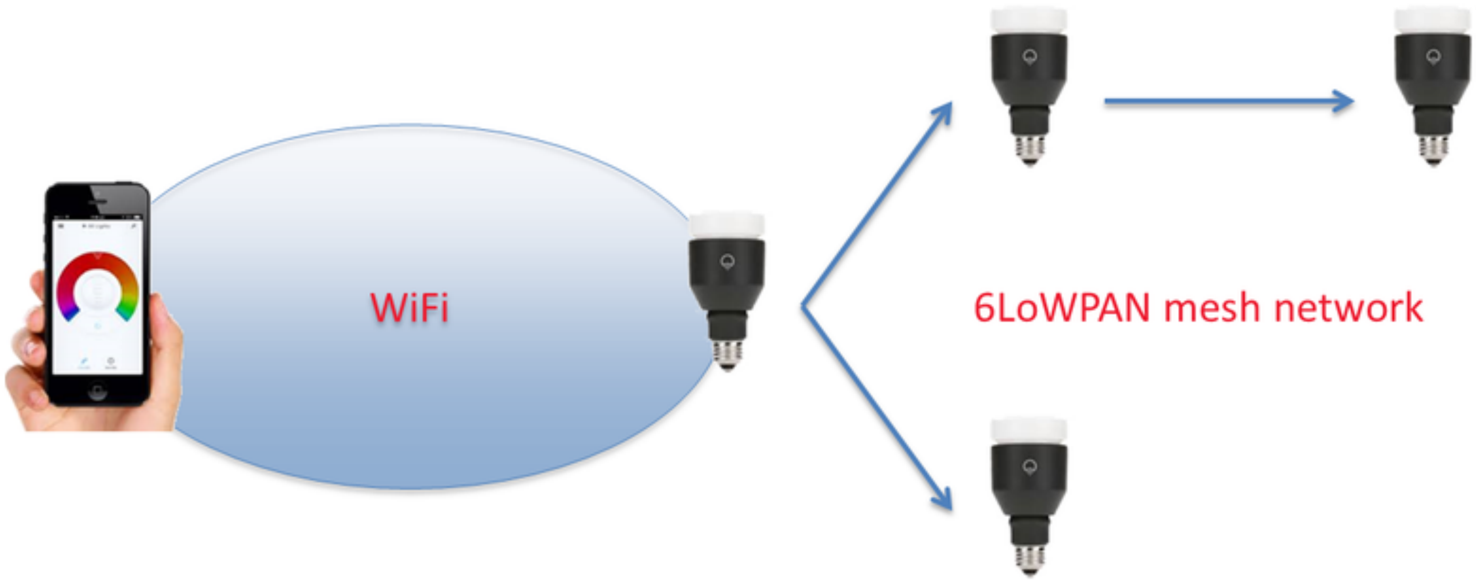
Inadequate software update mechanism

- **Example:** Huawei Home gateway
- Embedded web server (released 2002) with buffer overflow vulnerability
- Fix released in 2005 by web server company
- Vulnerability still exploited [1] (2015)

[1] http://www.computerworld.com/article/2860843/vulnerability-in-embedded-web-server-exposes-millions-of-routers-to-hacking.html

Missing Key Management Problem

- **Example:** LIFX [1] - Internet connected light bulb



- AES key shared among all devices to simplify key management

[1] http://www.contextis.com/resources/blog/hacking-internet-connected-light-bulbs/

Insecure Configuration Files and Default Passwords

- **Example:** Foscam, Linksys, Panasonic surveillance / baby monitoring cameras

- Default passwords or insecure default settings
- Similar problems on LED bulbs, etc.

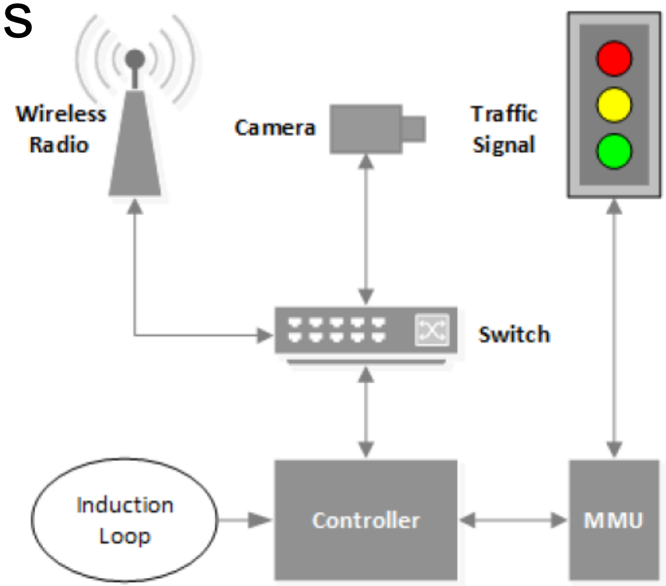
[1] http://www.networkworld.com/article/2844283/microsoft-subnet/peeping-into-73-000-unsecured-security-cameras-thanks-to-default-passwords.html

Aalto University

UNIVERSITY OF HELSINKI

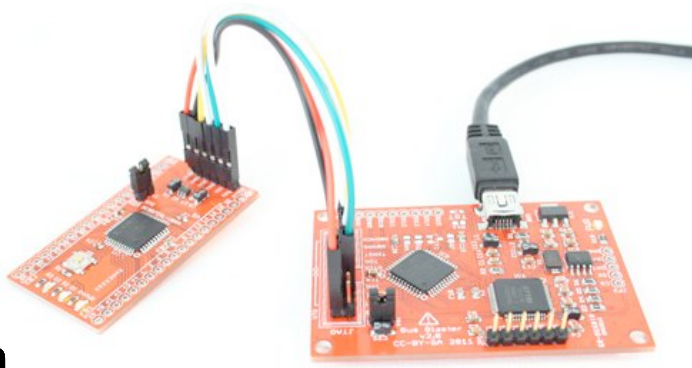
Missing Communication Security

- **Example:** Traffic infrastructure [1]
- Unencrypted wireless communications
- Default Username and Passwords (published online by manufacturer)
- Controller settings can be configured remotely
- FTP connection to write configuration files
- Physical attacks



[1] Ghena,et al. Green "Lights Forever: Analyzing the Security of Traffic Infrastructure"

Physical Attacks



- **Example:** extract keys, configuration data, firmware images

- Use of debug / test interfaces & sniffing on inter-bus communication interfaces like Serial Peripheral Interface (SPI) or Inter-Integrated Circuit (I^2C).

- Key extraction within a trusted execution environment using power analysis or fault injection (glitching) attacks.

Intermediate Summary (3 methods)

- 90% of the threats are common among all Internet protocols.

- Most of the (exploited) security vulnerabilities are rather basic .

- Many exploits of IoT systems today (particularly in the consumer space) are hoaxs.

4. Communications Design Patterns

- Device-to-Device
- Device via Smart Phone to Cloud
- Device via Gateway to Cloud
- P2P Communication in Local Network
- Device-to-Cloud

Device-to-Device Communication

- Characteristics:
  - Device talks directly to other device
  - Communication relies on link layer protocol mechanism (often no IP)

- Security:
  - Usually based on direct relationship between devices: **pairing**
  - Channel security provided mostly at the link layer

- Standardization:
  - RFID, 6LowPAN, ZigBee
  - Bluetooth Low Energy (LE) [1]

[1] https://www.bluetooth.com/what-is-bluetooth-technology/bluetooth-technology-basics/low-energy

Device via Smart Phone to Cloud

- Characteristics:
  - Extension of the device-to-device communication
  - Device interacts with smart phone and cloud services
- Security:
  - Classical smart phone app / Web development
  - Rarely end-to-end security
- Standardization:
  - Bluetooth LE, NFC

Aalto University

UNIVERSITY OF HELSINKI

Device via Gateway to Cloud

- Characteristics:
  - Devices communicate with cloud services via a network gateway
  - Apps/websites allow user-friendly, remote access/monitoring
- Security:
  - Network access authentication need
  - Example: EAP, PANA, AAA, etc.
- Standardization:
  - IEEE 802.15.4, WiFi, Bluetooth LE

P2P Communication in Local Network

- Characteristics:
  - Variant of “device via gateway to cloud” with local-only operation.
  - Discovery of nodes to communicate with
- Security:
  - Communication assumed to be local in the network
  - Authentication of nodes
- Standardization:
  - Universal Plug and Play (UPnP) + UPnP-UP
  - DNS Service Discovery
  - Bonjour (Apple)

Device-to-Cloud

- Characteristics:
  - Devices communicate with cloud service directly
  - Pre-configured to work with specific cloud service only
  - Always-on reachability required
  - Radio technology and IP-connectivity to local network for Internet access
- Security:
  - Network access authentication
  - End-to-end security for cloud access
- Standardization:
  - WiFi

Good Practices (recommendation)

- Always encrypt → avoid pervasive monitoring
- Follow key length recommendation (112-bit symmetric key equivalent)
- Support automatic key management
- Automatic software update mechanism
- Communication channel security (DTLS/TLS)
- Authentication and authorization solution
- Reduce physical attack surface:
  - Crypto implementations that consider side channel attacks
  - Disabled debug facilities before launching product
  - Hardware-based crypto support
  - Memory protection unit (MPU) integration

Current research project in IoT security

- Encryption Key Provisioning and Device Pairing
- Vulnerable & Unpatched IoT Devices (software update issues)
- Authentication of Passive IoT Entities

Encryption Key Provisioning and Device Pairing

- Pairing and provisioning → ok for 2 devices
- What about tens of devices ?
- **Solution:** Ambient audio signature [1]
  - Monitor the ambient sound perceived by several devices
  - Conclude if they are in a same room
  - Pair them and generate context specific encryption key

[1] Miettinen et al. “Context-Based Zero-Interaction Pairing and Key Evolution for Advanced Personal Devices”

Vulnerable & Unpatched IoT Devices (software update issues)

- Co-existence of vulnerable and secure devices in same network
- Vulnerable devices ➡ secure devices
- **Solution:** Identification and isolation of vulnerable devices [1]
  - Fingerprinting of connected devices
  - Inference of vulnerabilty / infection
  - Isolation in different VLAN

[1] https://wiki.aalto.fi/display/sesy/2015#id-2015-AutomatedIdentificationofIo(InternetofThings)Devices

Authentication of Passive IoT Entities

- Beacons / RFID tags / bar code send information to other devices
- Attackers can broadcast malicious content using this means
- **Solution:** Assess the legitimacy of broadcast information
  - Fingerprinting techniques for bluetooth beacons
  - Map physical characteristics of devices to legitimacy of the information (central database / collaborative system)

Credits

- Hannes Tschofenig - “How to secure the Internet of Things?” - 2014