FH JOANNEUM
Elektronik und
Computer Engineering

Security in the World of IoT

FH JOANNEUM
Institute of Electronic Engineering
Egon Teiniker
2021-12-15

Security in the World of IoT

Outline

- Knowledge for IoT Security

- IoT Devices
  - Identifying Buses and Interfaces
  - Firmware Reverse Engineering
  - Countermeasures Against Static Analysis

- IoT Services
  - API Security Basics
  - Edge Security
  - Service to Service Communication
  - Message Queuing Telemetry Transport (MQTT)

FH JOANNEUM
Elektronik und
Computer Engineering

Knowledge for IoT Security

OWASP IoT Top 10 - 2018
I01:2018 - Weak Guessable, or Hardcoded Passwords
I02:2018 - Insecure Network Services
I03:2018 - Insecure Ecosystem Interfaces
I04:2018 - Lack of Secure Update Mechanism
I05:2018 - Use of Insecure or Outdated Components
I06:2018 - Insufficient Privacy Protection
I07:2018 - Insecure Data Transfer and Storage
I08:2018 - Lack of Device Management
I09:2018 - Insecure Default Settings
I10:2018 - Lack of Physical Hardening

Knowledge for IoT Security

OWASP IoT Top 10 – 2018

- **I01:2018 - 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.

- **I02:2018 - Insecure Network Services**
  Unneeded or insecure network services running on the device itself, especially those exposed to the internet, that compromise the confidentiality, integrity/authenticity, or availability of information or allow unauthorized remote control.

Knowledge for IoT Security

OWASP IoT Top 10 – 2018

- **I03:2018 - Insecure Ecosystem Interfaces**
  Insecure web, backend API, cloud, or mobile interfaces in the ecosystem outside of the device that allows compromise of the device or its related components. Common issues include a lack of authentication/authorization, lacking or weak encryption, and a lack of input and output filtering.

- **I04:2018 - Lack of Secure Update Mechanism**
  Lack of ability to securely update the device. This includes lack of firmware validation on device, lack of secure delivery (un-encrypted in transit), lack of anti-rollback mechanisms, and lack of notifications of security changes due to updates.

Knowledge for IoT Security

OWASP IoT Top 10 – 2018

- **I05:2018 - Use of Insecure or Outdated Components**
  Use of deprecated or insecure software components/libraries that could allow the device to be compromised. This includes insecure customization of operating system platforms, and the use of third-party software or hardware components from a compromised supply chain.

- **I06:2018 - Insufficient Privacy Protection**
  User's personal information stored on the device or in the ecosystem that is used insecurely, improperly, or without permission.

Knowledge for IoT Security

OWASP IoT Top 10 – 2018

- **I07:2018 - Insecure Data Transfer and Storage**
  Lack of encryption or access control of sensitive data anywhere within the ecosystem, including at rest, in transit, or during processing.

- **I08:2018 - Lack of Device Management**
  Lack of security support on devices deployed in production, including asset management, update management, secure decommissioning, systems monitoring, and response capabilities.

Knowledge for IoT Security

OWASP IoT Top 10 – 2018

- **I09:2018 - Insecure Default Settings**
  Devices or systems shipped with insecure default settings or lack the ability to make the system more secure by restricting operators from modifying configurations.

- **I10:2018 Lack of Physical Hardening**
  Lack of physical hardening measures, allowing potential attackers to gain sensitive information that can help in a future remote attack or take local control of the device.

IoT Devices

Identifying Buses and Interfaces

- **Universal Asynchronous Receiver Transmitter (UART)**
  Debug logs from device bootup, terminal
- **Serial Peripheral Interface (SPI) and Inter-Integrated Circuit (I2C)**
  Data communications between different components (EEPROM, RTCs) in an embedded device circuit.
- **Joint Test Action Group (JTAG)**
  Read/write data, debug running processes, modify program execution flow.

IoT Devices

Firmware Reverse Engineering

Firmware is a piece of code residing on the nonvolatile section of the device. It consists of various components such as **kernel**, **bootloader**, **file system**, and additional resources.

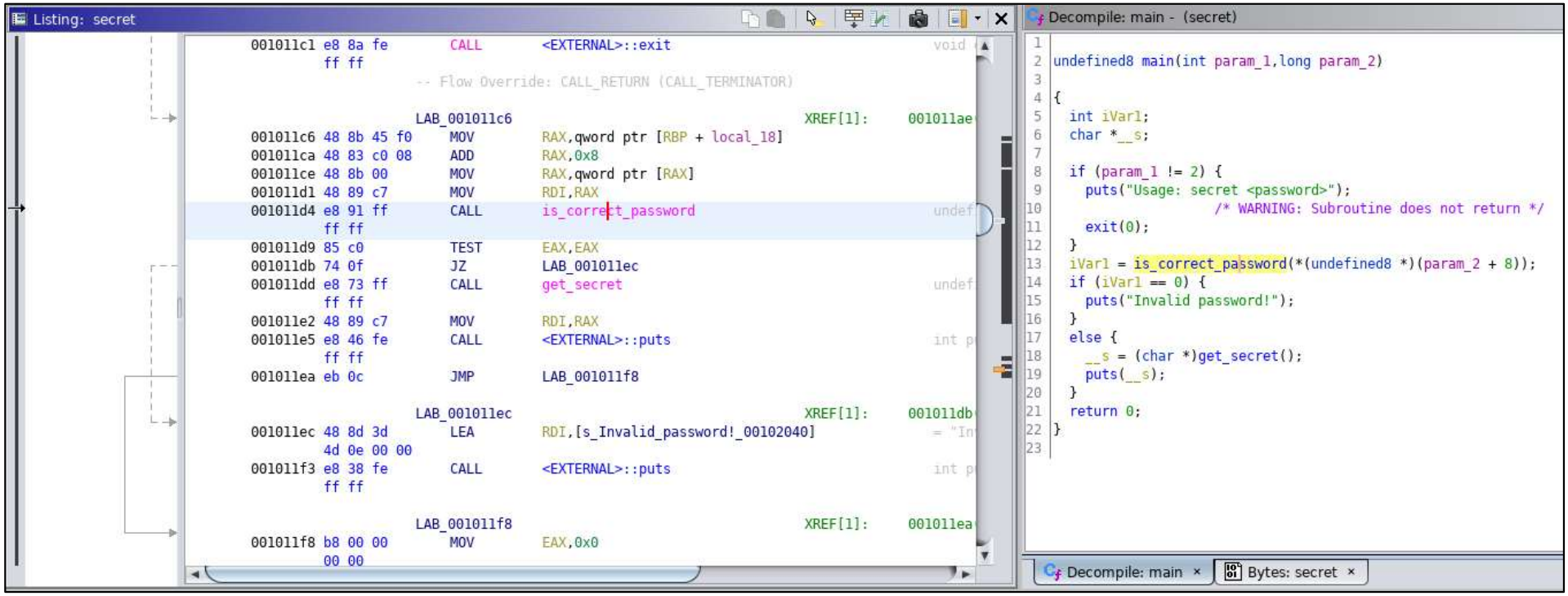
```
$ binwalk -e IoTGoat-x86.img

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
262144        0x40000         Linux EXT filesystem, blocks count: 4096,
                              image size: 4194304, rev 2.0, ext2 filesystem data,
                              UUID=57f8f4bc-abf4-655f-bf67-946fc0f9c0f9
5325930       0x51446A        xz compressed data
17301504      0x1080000       Squashfs filesystem, little endian, version 4.0,
                              compression:xz, size: 3481630 bytes, 1352 inodes,
                              blocksize: 262144 bytes, created: 2019-01-30 12:21:02
```

https://www.softscheck.com/en/reverse-engineering-tp-link-hs110/

IoT Devices

Firmware Reverse Engineering



IoT Devices

Countermeasures Against Static Analysis

As developers, we can do a lot to complicate reverse engineering:

- **Do not store secrets** (passwords or cryptographic keys) in the code.
- **Do not deliver debug information** (without compiler flag -g) with your binary file.
- **Use compiler optimizations** (compiler flag **–O2**) to make generated code more compact and difficult to read.
- Use the tool **strip** to also remove symbolic constants.

IoT Services

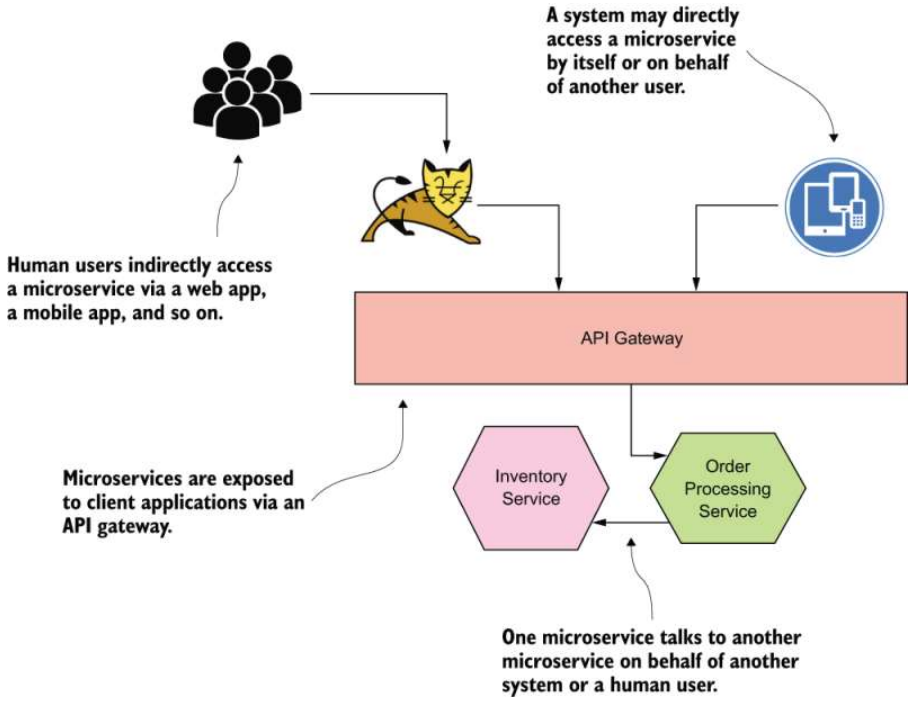
API Security Basics

- **Transport Layer Security (TLS)**
- **Authentication**
  Verifying who someone is.
- **Authorization**:
  Verifying what specific applications, files, and data a user has access to.
  Service-, Function-, and Object-Level.
- **API Specifications**
  Open API (versioning, operations, responses, data constraints)

IoT Services

Edge Security

- **API Gateway**
  Routing, Request Filtering
- **Access Control**
  OAuth 2.0
- **Throttling, Monitoring**

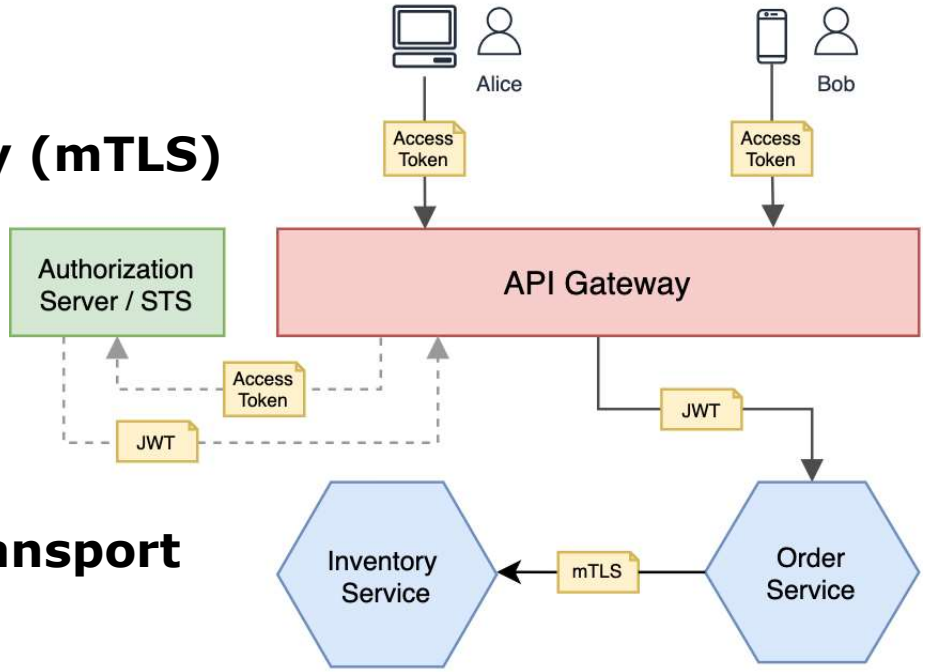


(Siriwardena, 2020)

IoT Services

Service to Service Communication

- **Mutual Transport Layer Security (mTLS)**
  Certificate management
- **JSON Web Token (JWT)**
- **Message Queuing Telemetry Transport (MQTT) over TLS**
  (publish/subscribe, client/broker)



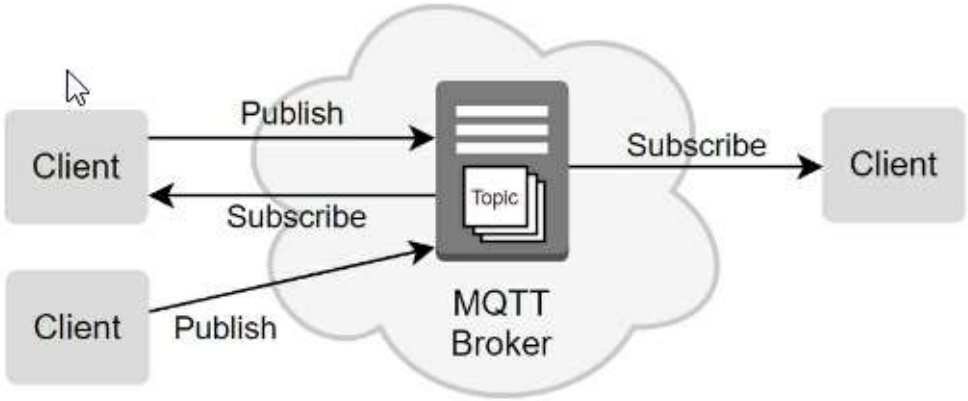
(Siriwardena, 2020)

IoT Services

Message Queuing Telemetry Transport (MQTT)

MQTT is a broker-based **publishing and subscription messaging protocol**.

MQTT uses **Transport Layer Security (TLS)** encryption with username, password protected connections.



(Smart, 2020)

FH | JOANNEUM
Elektronik und
Computer Engineering

References

- **OWASP Internet of Things**
  https://owasp.org/www-project-internet-of-things/

- *Aditya Gupta*
  The IoT Hacker's Handbook
  Apress 2019

- *Prabath Siriwardena, Nuwan Dias*
  Microservices Security in Action
  MANNING 2020

- *Gary Smart*
  Practical Python Programming for IoT
  Packt Publishing, 2020