

# RIOT and SUIT

Koen Zandberg



**TRiBE**  
inTeRnet BEyond the usual

# Overview

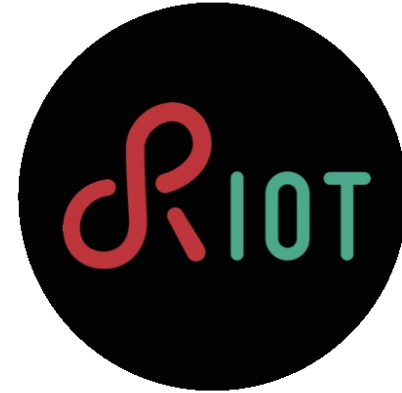
---

- Intro
- Update Architecture Overview
- Device Software Components
- Bootloader
- Demo

# Open source platform: RIOT

---

- 32/16/8-bit MCU
- Open Standards
- Internet of Things protocols stack



<https://github.com/RIOT-OS>

# SUIT Implementation

---

- RIOT SUIT example: <https://git.io/suit-updater>

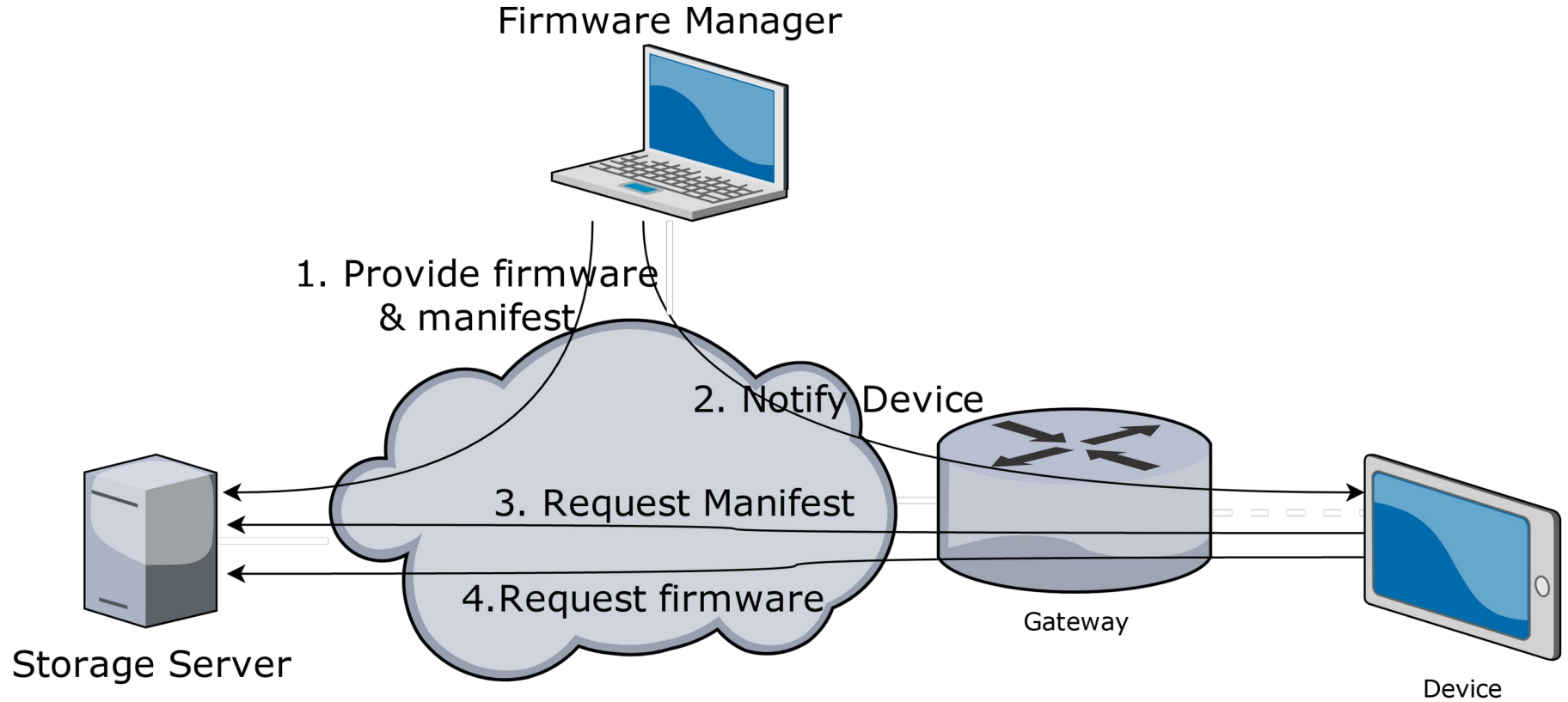
# Overview

---

- Intro
- Update Architecture Overview
- Device Software Components
- Bootloader
- Demo

# Update Architecture Overview

---

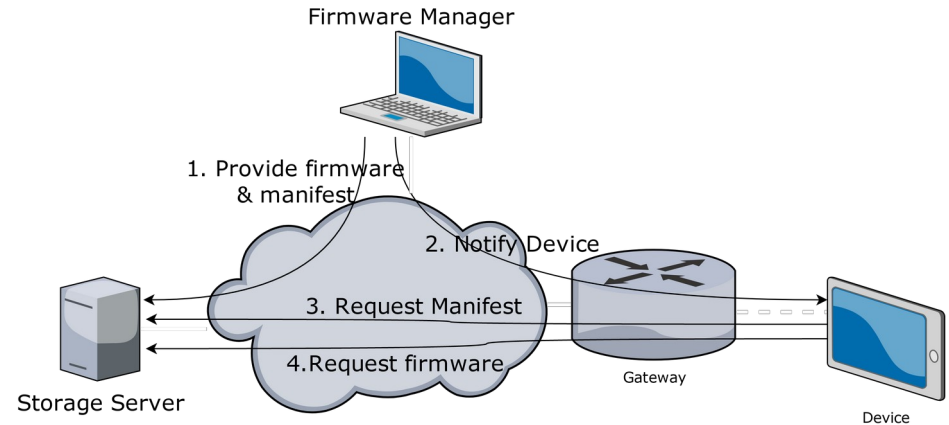


# Update Architecture Overview

---

1. Store firmware and manifest on the server

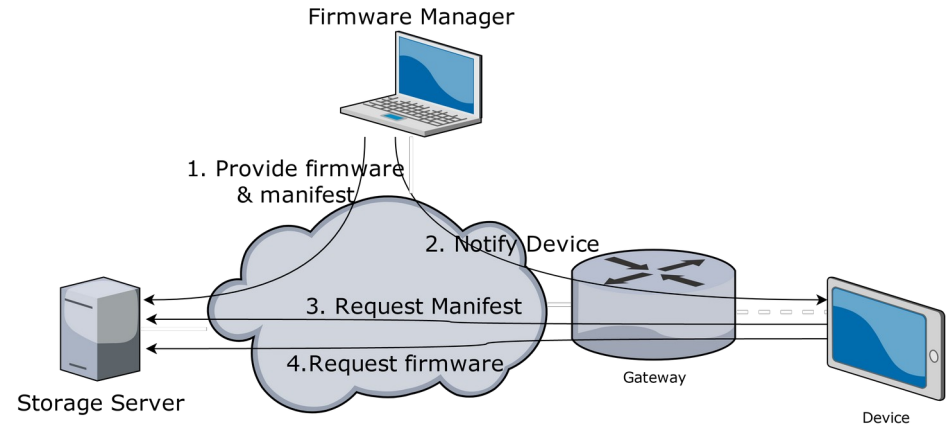
- Within the RIOT demo this is the same machine



# Update Architecture Overview

---

## 2. Notify the IoT device that an update is pending



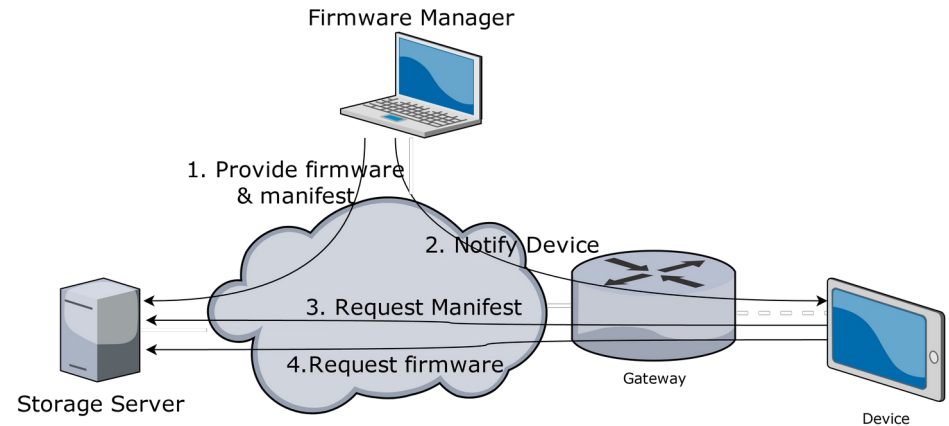


# Update Architecture Overview

---

3. The IoT device requests the manifest from the server.

- Here the SUI manifest parsing starts

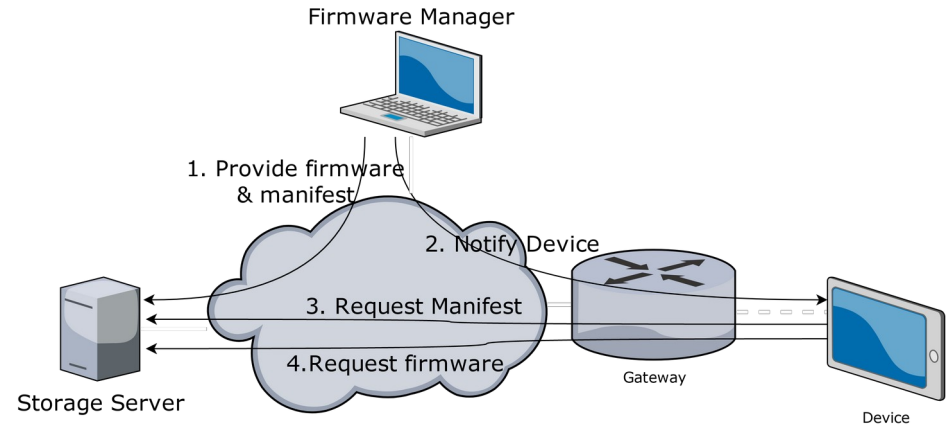


# Update Architecture Overview

---

4. The IoT device downloads the firmware from the server.

- Immediately streamed to the device flash, we skip the install



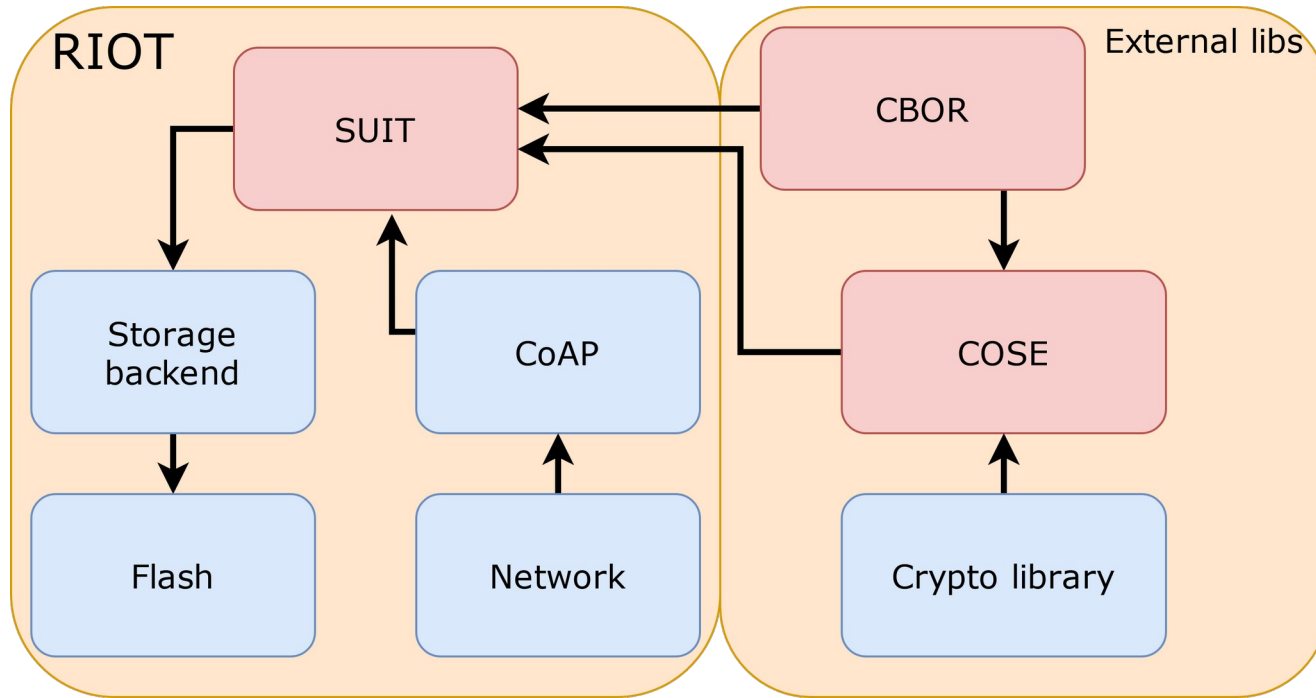
# Overview

---

- Intro
- Update Architecture Overview
- Device Software Components
- Bootloader
- Demo

# Device Software Components

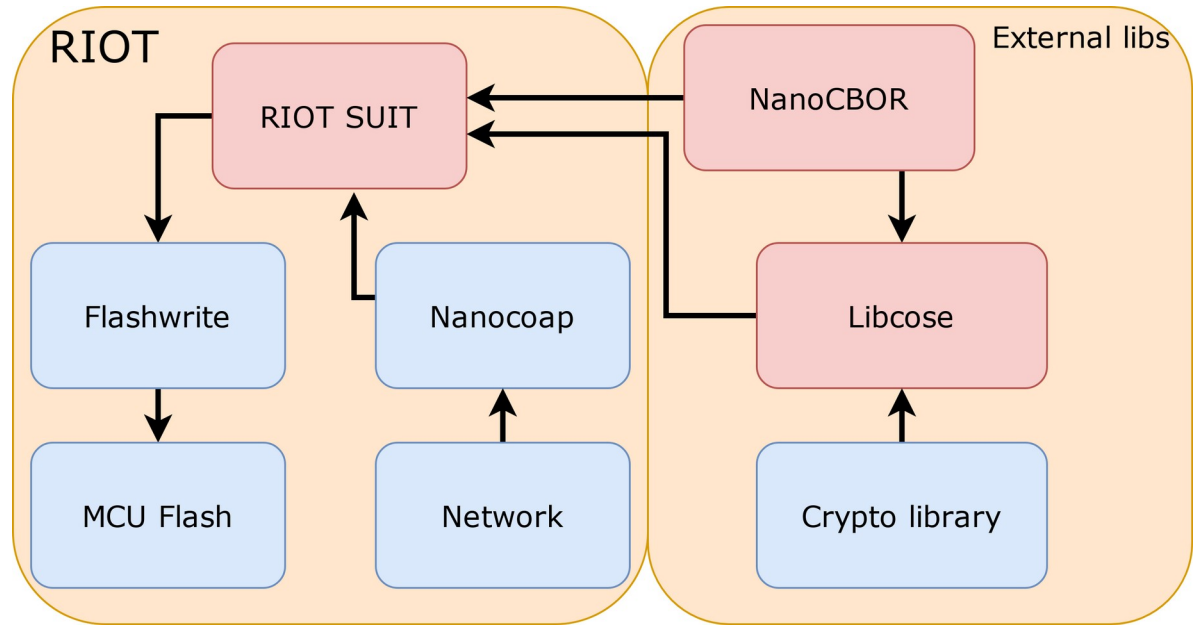
---



# Device Software Components

---

- Main components
  - SUIT Parser
  - NanoCBOR
  - Libcose



# Device Software Components

---

- NanoCBOR
  - Pull style CBOR parser
  - 600B – 800B decoder
  - Optimized for parsing fixed schema structures
  - <https://github.com/bergzand/NanoCBOR>

# Device Software Components

---

- Libcose
  - Embedded COSE library
  - Multiple crypto backend support
  - <https://github.com/bergzand/libcose>

# Device Software Components

---

- SUIT parser
  - Using NanoCBOR and Libcose
  - Iterates over the CBOR maps
  - Jumptable-style design based on the map keys



# Device Software Components

---

- Flash Writer
  - Dual slot A/B architecture
  - Writes the firmware directly to the flash of the device (to the other slot)
  - Version number used to select the slot

# Overview

---

- Intro
- Update Architecture Overview
- Device Software Components
- Bootloader
- Demo

# Bootloader

---

- Just another RIOT firmware
  - Iterate over the slots
  - Determine whether the header is valid
  - Determine which valid firmware has highest seq. number
  - Boot the selected firmware

[https://api.riot-os.org/group\\_\\_bootloader\\_\\_riotboot.html](https://api.riot-os.org/group__bootloader__riotboot.html)

# Demo / Walk-through

---

- Update a payload on RIOT