Networking the IoT with RIOT

Thomas C. Schmidt, Internet Technologies
t.schmidt@haw-hamburg.de
HAW Hamburg, Dept. Informatik
Outline

- The Internet of Things
- The RIOT Operating System
- The RIOT Networking Subsystem
- Federated Authentication for the IoT
- The Case for Information-centric Networking?
- Conclusions & Outlook
The Internet of Things

“A system in which objects in the physical world can be connected to the Internet by sensors and actuators”
(coined 1999 by Kevin Ashton)

50 billion IoT devices expected by 2020!
IoT Networking: Connecting the Physical World to the Internet

- Micro- & Nano Satellites
- Connected Vehicles
- Smart Homes
- eHealth

Industrial Automation

Connected Vehicles

Smart Homes

eHealth

http://inet.haw-hamburg.de/
Use Case: Security in Harsh Industrial Environments
Smart DOM
Hamburg
'Smart' Heating
Evolution Towards an IoT

Distributed local intelligence

- Embedded Controllers

+ Wireless sensor network

- Wireless Networking

+ Internet of Things?

- IPv4 Uplink to the Cloud
This is not yet an Internet of Things!
No Internet without Open Speech and Open Standards
Evolution towards an *Internet of Things* (IoT)

**Distributed local intelligence**
- Embedded Controllers
- Interoperable Information

**Wireless sensor network**
- Wireless Networking
- Distributed Security

**Hype-Internet of Things**
- IPv4 Uplink to the Cloud
- Things loosely joined by IPv6

The Real Internet of Things (C. Bormann)
The many faces of IoT

High-end IoT

Processor: GHz, 32/64 Bit
Memory: M/Gbytes
Energy: Watt
Network access: 5G, WLAN
The many faces of IoT

High-end IoT

Processor: GHz, 32/64 Bit
Memory: M/Gbytes
Energy: Watt
Network access: 5G, WLAN

Low-end (or constrained) IoT

Processor: MHz, 8/16/32 Bit
Memory: kbytes
Energy: MWatt
Network access: 802.15.4, BLE
The many faces of IoT

High-end IoT

Processor: GHz, 32/64 Bit
Memory: M/Gbytes
Energy: Watt
Network access: 5G, WLAN

Low-end (or constrained) IoT

Processor: MHz, 8/16/32 Bit
Memory: kbytes
Energy: MWatt
Network access: 802.15.4, BLE
The Constrained Internet of Things (IoT)

Internet

Memory > 4GB

Memory > 4GB

Memory > 4GB

Memory ~ 2 GB

Memory ~ 1 GB

Memory ~ 500 MB

Memory ~ 16 KB

Memory ~ 8 KB

Memory ~ 100 KB

Constrained + Wireless!
The IoT is Very Heterogeneous

- Various boards
- A zoo of components
- Broad range of radios
- Different Link-layers
- Competing network layers
- Diverging interests and technologies
- A lot of experimentation ...
Low power lossy wireless

Key problem: Energy

The History of Wireless

London 1783: The First Prototype of the Wireless Gallows

We're still working on it

Ohm

Volt

Amp

Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences
RIOT: The Friendly OS for the IoT

Internet

IoT = programmable world
If your IoT device cannot run Linux, then run
RIOT: Facts sheet

- Microkernel architecture (for robustness)
  - The kernel itself uses ~1.5K RAM @ 32-bit
- Efficient hardware abstraction
- Tickless scheduler (for energy efficiency)
- Deterministic O(1) scheduling (for real-time)
- Low latency interrupt handling (for reactivity)
- Modular structure (for adaptivity)
- Preemptive multi-threading & powerful IPC
- Appealing API
RIOT Origins

History

- **2008** – Project roots: The kernel was started as part of the FireWhere project
- **2010** – Towards the IoT: Implementation of 6LoWPAN and RPL was initiated (GLAB)
- **2013** – RIOT goes public: Branding of RIOT as part of the SAFEST project, source code moved to GitHub

Founding institutions

- Freie Universität Berlin
- Hochschule für Angewandte Wissenschaften Hamburg
- INRIA
RIOT Software Components

Application

pkg | sys | network stack

core (kernel) | drivers

Platform-independent

Platform-dependent

drivers/periph

cpu | board

Hardware

sock

netdev

HAL

SW-module

non-OS
RIOT: Built to connect

- Open-access protocols
  - e.g. 6LoWPAN, IPv6, CoAP, ...
- RIOT supports several network stacks
- On many wireless technologies and NICs
RIOT Network Architecture: A Closer Look
API Design (netdev, netapi)

- **Transmission**
  - Asynchronous `send/recv`

- **Configuration and Initialization**
  - Functions `get/set` for options defined in a global key-value store: `netopt`

- **Event Handling**
  - Signaling related to the categories RX, TX, link, or system
  - Provides external event callback to break out of ISR
Supported Components

- **Full Stacks**
  - GNRC
  - LwIP
  - emb6
  - OpenWSN
  - CCN-Lite
  - NDN-RIOT
  - LoRA-WAN
  - (Nimbel BLE)

- **Network Access**
  - 802.15.4 (various radios)
  - 802.15.4 CSMA
  - 802.15.4 TSCH
  - 802.3 Ethernet
  - LoRA
  - (BLE)
Performance of a Fully Layered Stack?

Transmitting a UDP Packet

msp430 16-bit MCU, 16 MHz, 92 KB Flash, 8 KB RAM, on-board 2.4 GHz 802.15.4 radio
Memory

(a) ROM

(b) RAM

API
UDP
IPv6
6LoWPAN
MAC
Auxiliary
Energy

![Bar chart showing energy consumption for different systems. The x-axis represents energy consumption in Joules (J), and the y-axis lists the systems: GNRC (RIOT), lwIP (RIOT), emb6 (RIOT), uIP (Contiki). The chart compares the energy consumption for transmission (TX) and reception (RX). The uIP (Contiki) system has the highest energy consumption, followed by emb6 (RIOT), lwIP (RIOT), and GNRC (RIOT).]
RIOT Network Stack: Packet Processing Rates

![Graph showing packet processing rates for different layers with points indicating 10 Mbps and 1 Mbps speeds.](graph.png)
IoT Stacks in the Wild: RIOT versus RasPi
Processing Time per Layer

The graph shows the processing time in microseconds (μs) for different protocol payloads in bytes. The x-axis represents the protocol payload in bytes, ranging from 0 to 1200 bytes. The y-axis represents the processing time, ranging from 0 to 1400 μs.

Three lines represent different protocols:
- Solid line: UDP + IP + 6Lo
- Dashed line: IP + 6Lo
- Dash-dotted line: 6Lo

The processing time increases as the protocol payload increases.
Outline

- The Internet of Things
- The RIOT Operating System
- The RIOT Networking Subsystem
- Federated Authentication for the IoT
- The Case for Information-centric Networking?
- Conclusions & Outlook
The IoT Authentication Problem

- IoT systems may rely on peerwise verifiable identities
- Current standards (DTLS) leave authentication open
- Huge numbers and heterogeneous deployment hinder central trust hierarchy – TLS analogy is traditional PKI
- Autonomous local configuration and federated trust management preferred
- Constrained environments urge for minimal overhead
- Trust anchors may (should) be outside the loop
IoT End-to-End Authentication

Infrastructure-centric approach for DTLS:

- Identity-based crypto IPs authenticate IoT nodes without individual public keys
- Twisted Edwards ECC enables IBC on constrained nodes
- Federation between subnets - bound to local trust
Local Approach

- Bootstrap: either offline or
  1. TA pre-initializes Gateway with Identity credentials – then can go offline
     Alternative: TA locally protected configurator
  2. Gateway (or TA) performs host configuration by authenticated encryption using symmetric cypher (AEAD)
  3. Gateway distributes TAPK

- Authentication Process:
  - Peerwise end-to-end using IBC signatures

[Diagram showing the flow of communication between TA, Border Gateway, and IoT Device]
Federated Approach

- Problem: Requires local trust anchor
- Solution: Subnet-local Trusted Authorities (TAs) with Cryptographic Subnet IDs
- Federated Public Key Exchange
- Gateway-assisted End-to-end Authentication
- Key Revocation by renumbering
GW-assisted Authentication Process

1. \[H(2001:af:16 + \text{SenderTASp})\]
   \[
   \begin{array}{c}
   \text{SenderID} \\
   \end{array}
   \]
   
2. lookup \text{SenderTASp} for \[2001:af:16:018e:e631:728b:eb82\] prefix

3. \[H(2001:af:16 + \text{SenderTASp})\] 
   \[
   \begin{array}{c}
   018e:e631:728b:eb82 \\
   \end{array}
   \]

4. forward packet + \text{SenderTASp}
Protocol Evaluation

Scenario:
- IoT node: SAM R21 from FIT IoT Lab running RIOT
- Gateway: Rasberry PI
Evaluation: Crypto Load

- Load reduction by Edwards Curve25519

Twisted Edwards Extended vs. Short Weierstrass

<table>
<thead>
<tr>
<th>Operation</th>
<th>Twisted Edwards Extended</th>
<th>Short Weierstrass</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verify</td>
<td>104.74</td>
<td>307.32</td>
</tr>
<tr>
<td>Sign</td>
<td>122.54</td>
<td>389.53</td>
</tr>
<tr>
<td>Extract</td>
<td>100.27</td>
<td>122.51</td>
</tr>
</tbody>
</table>

CPU time [cycles \cdot 10^6]

Contribution to the Relic Library on...
Outline

- The Internet of Things
- The RIOT Operating System
- The RIOT Networking Subsystem
- Federated Authentication for the IoT
- The Case for Information-centric Networking?
- Conclusions & Outlook
Information-centric Networking

Idea:

- Access content – not nodes – in a request/response paradigm
  - Address content directly by name
  - Augment content with (self-)authentication
  - Ubiquitous in-network storage (caching)

Various approaches

- Seminal: TRIAD (Gritter & Cheriton 2001)
- Most popular: NDN (Van Jacobson et al. 2009)
Basics: Content Centric Routing

- Observation 1: In-network states driven by data
- Observation 2: End-users affect backbone states
Opportunities with ICN for IoT

- Resilience
  - Hop-by-hop traversal with in-network caching

- Mobility
  - Seamless support for mobile consumers

- Security
  - DoS difficult: No unwanted data

- Network management
  - Easy deployment and auto-configuration

- Network caches
  - May reduce latency for multiple services
  - Fixed deputies may preserve energy
ICN in RIOT
GNRC versus CCN-Lite

(a) ROM

(b) RAM

Prof. Dr. Thomas Schmidt  http://inet.haw-hamburg.de/
NDN Network Performance

\[ \bar{x} = 1746.64 \text{ ms} \]
\[ \sigma = 3036.55 \text{ ms} \]

Grenoble Site - up to 9 hops
100 % packet delivery
Conclusions & Outlook

- There is demand for a flexible software support with versatile experimentation options.

- The RIOT networking architecture satisfies these demands by:
  - High-level hardware abstraction with clean interfaces
  - Powerful, recursive layering at very low performance penalty

- Federated trust - bootstrapped by crypto-based identifiers

- ICN a promising candidate for IoT deployment.
References


Thanks & Questions ?

Group:  http://inet.haw-hamburg.de/

RIOT:  http://www.riot-os.org/
RIOT Summit
September 13 - 14, 2018
Meet in Amsterdam!

http://summit.riot-os.org