T2TRG: Thing-to-Thing Research Group

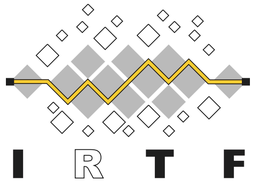
T2TRG Work Meeting, November 3, 2023

Chairs: Carsten Bormann & Ari Keränen

Note Well

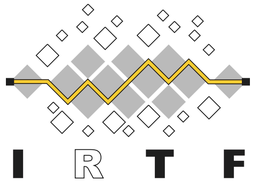
- You may be recorded
- Be nice
- The IPR guidelines of the IETF apply:
  see **http://irtf.org/ipr** for details.

Note Well – Privacy & Code of Conduct



- As a participant in, or attendee to, any IRTF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public
- Personal information that you provide to IRTF will be handled in accordance with the Privacy Policy at https://www.ietf.org/privacy-policy/
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this
- See RFC 7154 (Code of Conduct) and RFC 7776 (Anti-Harassment Procedures), which also apply to IRTF

Note Well – Intellectual Property



- **The IRTF follows the IETF Intellectual Property Rights (IPR) disclosure rules**
- By participating in the IRTF, you agree to follow IRTF processes and policies:
  - If you are aware that any IRTF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion
  - The IRTF expects that you file such IPR disclosures in a timely manner – in a period measured in days or weeks, not months
  - The IRTF prefers that the most liberal licensing terms possible are made available for IRTF Stream documents – see RFC 5743
  - Definitive information is in RFC 5378 (Copyright) and RFC 8179 (Patents, Participation), substituting IRTF for IETF, and at https://irtf.org/policies/ipr

Goals of the IRTF

- The Internet Research Task Force (IRTF) focuses on longer term research issues related to the Internet while the parallel organisation, the IETF, focuses on shorter term issues of engineering and standards making

- **The IRTF conducts research; it is not a standards development organisation**

- While the IRTF can publish informational or experimental documents in the RFC series, its primary goal is to promote development of research collaboration and teamwork in exploring research issues related to Internet protocols, applications, architecture, and technology
- See “An IRTF Primer for IETF Participants” – RFC 7418

Administrivia

- (Blue sheets maintained by meetecho)
- Note-takers: **https://notes.ietf.org/notes-ietf-interim-2023-t2trg-02-t2trg**
- Meetecho chat: also via Zulip: **https://zulip.ietf.org/#narrow/stream/t2trg**
- Mailing List: **t2trg@irtf.org** — subscribe at: **https://www.ietf.org/mailman/listinfo/t2trg**
- Repo: **https://github.com/t2trg/2023-11-prague**

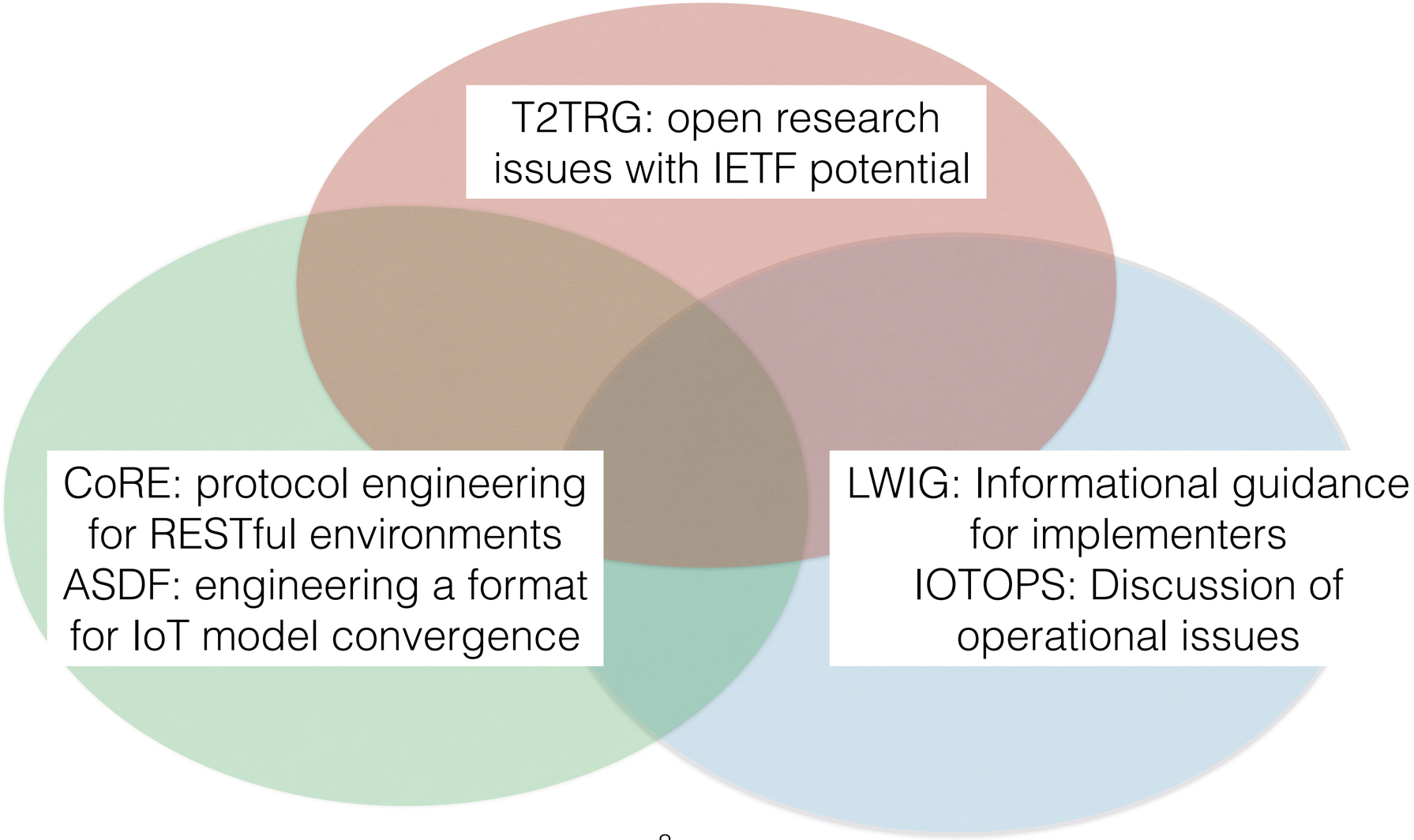
T2TRG scope & goals

- Open research issues in turning a true "Internet of Things" into reality
  - Internet where low-resource nodes ("things", "constrained nodes") can communicate among themselves and with the wider Internet
- Focus on issues with opportunities for IETF standardization
  - Start at the IP adaptation layer
  - End at the application layer with architectures and APIs for communicating and making data and management functions, including security

IRTF and IETF

IRTF
(Research)

IETF
(Engineering)



This meeting

Face-to-Face (8h) “work meeting” on security, protocols, and modeling:

- 3 active RG drafts on security issues:
  - Amplification **attacks** using CoAP
  - Terminology and processes for **initial security setup** of IoT devices
  - A Taxonomy of operational security considerations for manufacturer installed **keys** and Trust Anchors
- 1 potential field of new work:
  - discovery: access control, security, privacy (see 2023-05 work meeting)

Agenda outline

Time	Subjects
1000–1200	Welcome; 10:30: draft-lee-asdf-digital-twin (KST timezone), Security topics: Raytime; active RG documents
1200–1330	(Lunch break)
1330–1530	(Security overflow), Secure Discovery & self-description security considerations; more self-description issues Protocols and Specifications
1530–1600	(Coffee break)
1600–1800	non-IP Cluster: Intro, NIPC, Onion CoAP, GATT…

From classes to instances

Carsten Bormann
T2TRG work meeting 2023-11-03

Class vs. Instance

- A “class” describes a set of (potential or actual) instances, by listing constraints
  - SDF: Description by modeling affordances
- So an instance is a class narrowed down to a singleton?

Close, but no cigar

Instance

- An instance *__is__* (or *__was__* or *__will be__*), not just potentially
- An instance has an **identity** (generalized term of art)
  - Identity **attributes** can change over time, while maintaining an abstract “identity”
  - (Need to consider all identity considerations, e.g., **directed** identity, **linkability**)
- Instances are *__created__* and *__destroyed__* (or garbage collected :-)

Instance Creation

- Instances have creation *__parameters__*
  - May be one-to-one to instance attributes
  - Often require more significant processing
  - May need to model the creation process (lifecycle steps)
- Some parameters/attributes may live in **twin** only

discovery access control, secure **discovery**, privacy-preserving **discovery**

Carsten Bormann

T2TRG work meeting 2023-05-24,
updated for 2023-11-03

Discovery: important for IoT

- Need to **automate** setup processes as much as possible
  - Reducing user involvement:
    Users make mistakes (and just want to “get the work done” anyway)
  - Limited user interfaces in devices
  - Large number of devices (pet ➔ cattle)
  - Large number of **kinds** of devices
- Network Discovery
- Resource Discovery (for Service Discovery)

What do we want to discover?

- (Device ➔ Network:) Device wants to discover the right **network**
  - Often already addressed by L2 (e.g., WiFi)
    May not offer the level of automation and security we want

- (Device ➔ Application:) Device wants to discover the right **community** to announce itself to: discovery service, data hub (e.g., pub/sub broker)

- (Application ➔ Device:) Application wants to discover devices:
  “its” devices, generally available devices (in its **community,**
  possibly established by **proximity**

Discovery vs. Search vs. Name Lookup

- Discovery often implies a search with little available **context** (“bootstrapping”) for at least one of the parties
- Search is a discovery with defined **criteria**
- Name Lookup is a **constrained** search, often with 0 or 1 result

- All are subject to **authorization**, which can be conceptualized as **views**
- Authorization may require **authentication**,
  which may need to be woven into the discovery process
  - Authentication often establishes **proximity**

Discovery: (Security) Objectives

- Roughly: Device can find infrastructure; infrastructure can find device
- “Find”: Learn enough to do something useful then

- Security Objectives: **CIA**
  - **Confidentiality**: No undesirable Disclosure
    Adversary does not learn what it shouldn’t
  - **Integrity**: Discovered information is intact
    Adversary cannot inject/spoof information
    (for DoS or to gain further access)
  - **Availability**: Adversary cannot disrupt functioning

Discovery: Confidentiality

- Confidentiality: Adversary does not learn what it shouldn't
  - Privacy requirements
    (Is there an insulin pump in this room? Any expensive gadgets worth to steal?)
    - Even stable identifiers create linkability issues (see MADINAS)
  - Do not disclose what would increase attack surface

- Realization:
  - Often falling back to (sub-)network security
  - Resource discovery: "View" approach (authorized users see more)
    Issue: Who is authorized to what? How did the authentication work?

Discovery: Integrity

- Integrity: Adversary cannot inject/spoof information
  - Abuse cases: DoS on actual discovery, confusion to gain further access

- Realization:
  - Often falling back to (sub-)network security
  - Integrity requires authentication —
    Issue: Who is authorized to what? How did the authentication work?

Discovery: Availability

- Availability: Adversary cannot disrupt functioning
  - “Finding”: Hard to ensure completeness of enumeration
  - Abuse of discovery capability for DoS (e.g., battery draining)

- Resource use should require authorization —
  Issue: Who is authorized to what? How did the authentication work?

Discovery Security as an Authorization Problem

- What authentication methods and authorization models are available for use in a discovery phase?
  - e.g., choose the right network
    melds into setup phase, e.g., accept “purpose in life”
- What partial authentication/authorization states can be used to authorize disclosure?
  - “Secret handshakes”: Mutual authentication **before** anything is disclosed to non-authorized parties

Multi-Layer Discovery

- Use Discovery Layer n to find parties to authenticate to,
  which are then authorized to help with further discovery

- Use that authentication (and authorization) for Discovery Layer n+1

Discovery: next steps

- Can we describe **challenges** (scenarios) that cannot be adequately addressed by currently deployed mechanisms?
- Do we have **technology** to address some of the challenges?
- Are there common high-level **models** that are useful?
  Issuer/Holder(/Subject)/Verifier (W3C VC term),
  Audience/Scope (OAuth), Delegation (multiple)

The non-IP world

Carsten Bormann
T2TRG work meeting 2023-11-03

Why non-IP?

- “last mile”: often on some specific technology
- Not all of these function well when used as IP networks (WiFi, Thread do)
  - Some could, but the market for non-IP is so much larger, so their apps run non-IP only
- But non-IP nodes need to be part of the same IoT

The last-mile protocol

- non-IP devices don’t have an IP address — *something* has to have that (“gateway”)
- Application code on non-IP devices: usually tied to L2 protocol(’s 1-2-7 ecosystem)
  - May need to abstract that and make that available over IP
- May need setup protocol to make the non-IP device available over IP (managing the IP address, onboarding/authorization)

CoAP as a last-mile protocol?

- For “cooperative” non-IP devices:
  What could be a good protocol abstraction?
- https://datatracker.ietf.org/doc/draft-amsuess-core-coap-over-gatt/ (BLE)
- https://datatracker.ietf.org/doc/draft-bormann-t2trg-slipmux/ (UART)

But what about MQTT?

- MQTT is great as a plumbing-over-IP protocol (NAT/firewall traversal)
  - Essentially a prototypical non-IP protocol over IP
- MQTT addresses notifications ➔ complex in HTTP
  - Now active: https://datatracker.ietf.org/doc/draft-toomim-httpbis-braid-http/ — BRAID, https://datatracker.ietf.org/doc/draft-gupta-httpbis-per-resource-events/ — PREP

Naming and addressing

- If we don’t have an IP address as the (short-term) identity of a device, then what?
- Need to embrace non-IP addresses
  - MAC addresses in GATT
  - no(!) addresses in Slipmux
- Global general mapping to IP addresses won’t happen — live with contextual pairs (tuples)