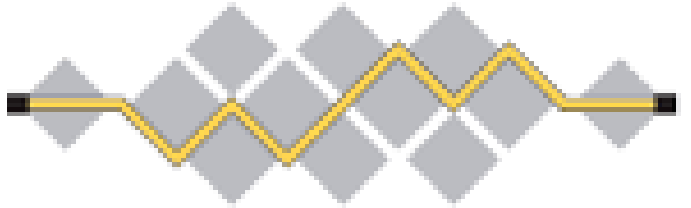
Lightweight Authenticated Key Exchange for Internet-of-Things Use Cases

Mališa Vučinić
Research Scientist, Inria

Outline

1. Internet of Things
2. Why standardize?
3. How to standardize?
   - Internet Engineering Task Force (IETF)
4. Lightweight Authenticated Key Exchange (LAKE)
   - A Primer on EDHOC
   - Security
   - Performance
5. Next Steps

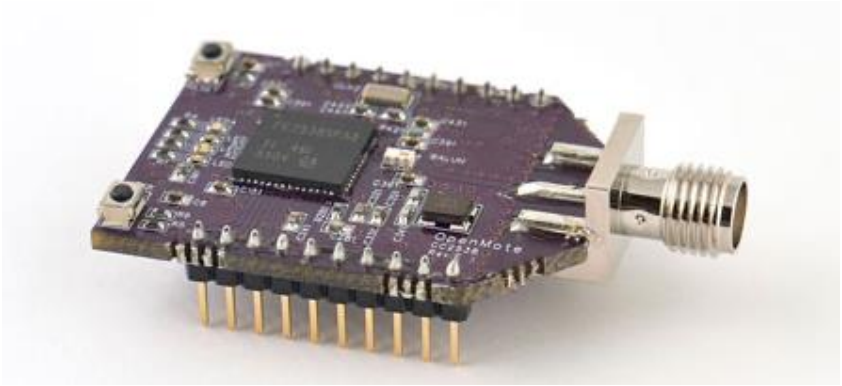


IETF®

Inria

Internet of Things Devices

Telos B (2005)
8 MHz
10 kB RAM
48 kB flash



OpenMote CC2538 (2014)
32 MHz
32 kB RAM
512 kB flash

nRF52840 dongle (2018)
64 MHz
256 kB RAM
1 MB flash

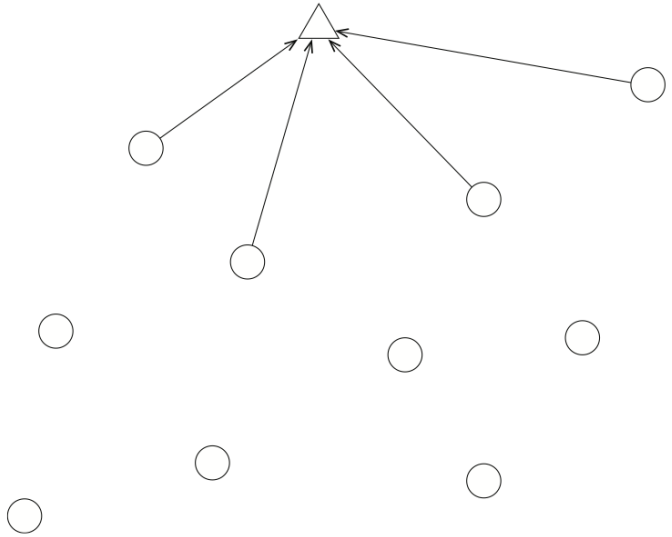
Internet of Things Radio Technologies

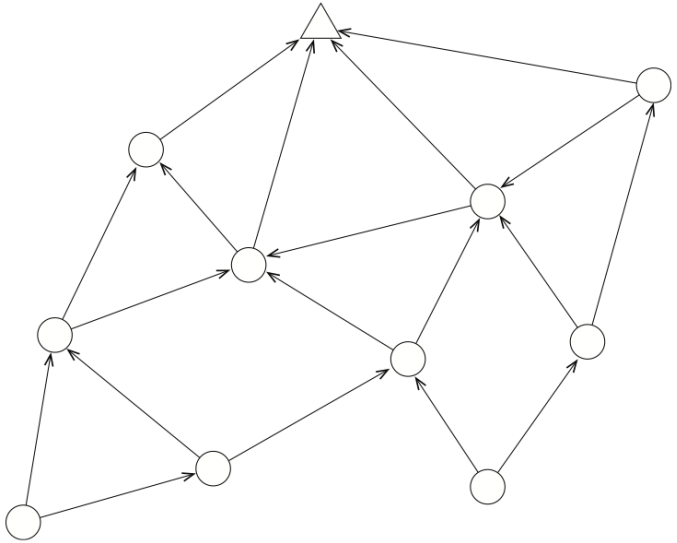
LoRaWAN®

NB-IoT™

6TiSCH

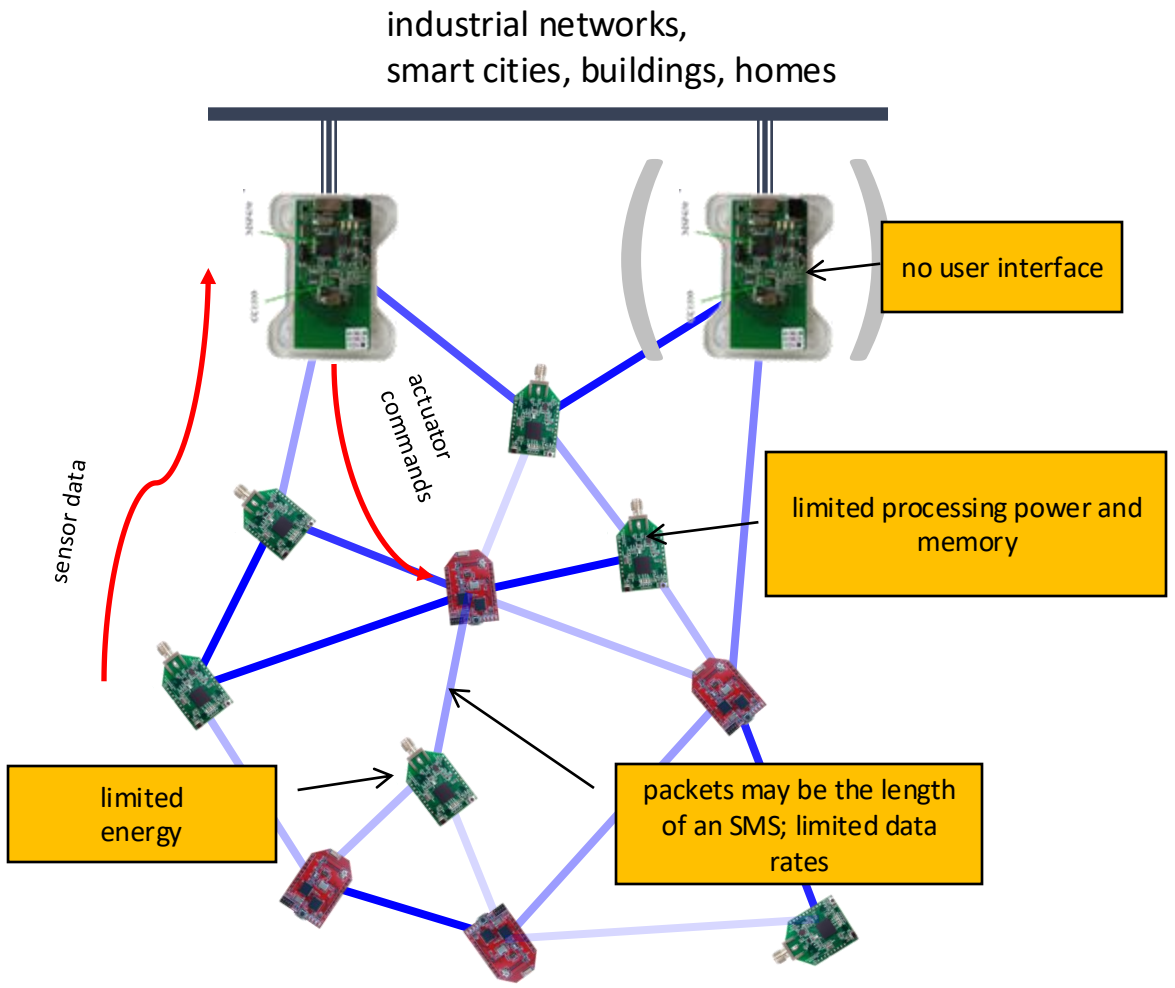
Bluetooth™ Low Energy



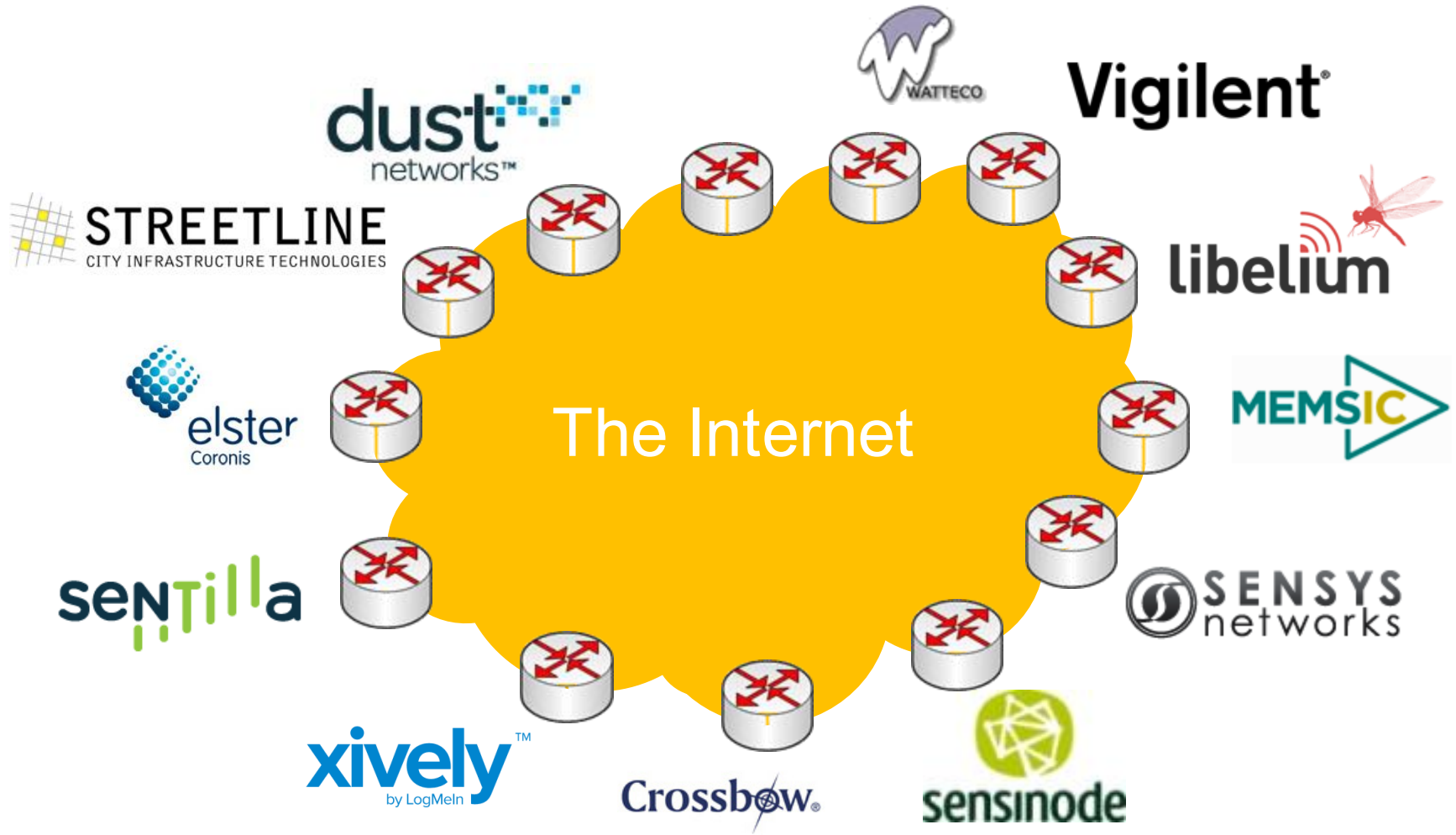


Internet of Things Constraints

- Standardized in different organizations
- Common device and network constraints
- Common security concerns



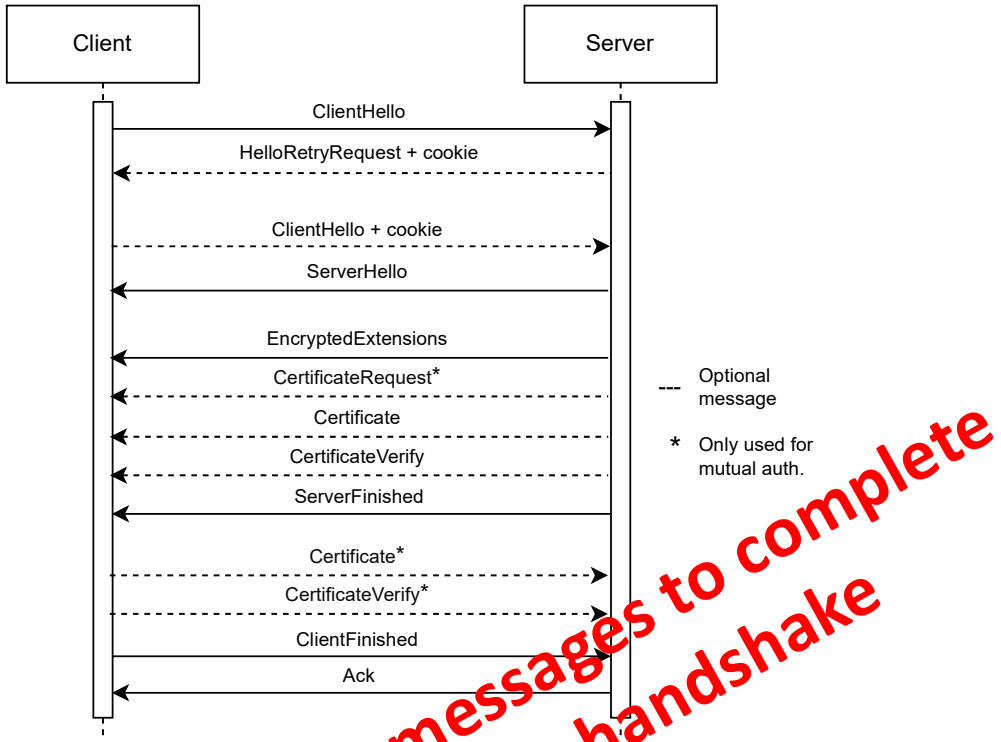
Why Standardize?



Slide credit: Thomas Watteyne

Existing Internet Security Technologies are Heavy

Transport Layer Security 1.3



>10 messages to complete the handshake

X.509 Certificates

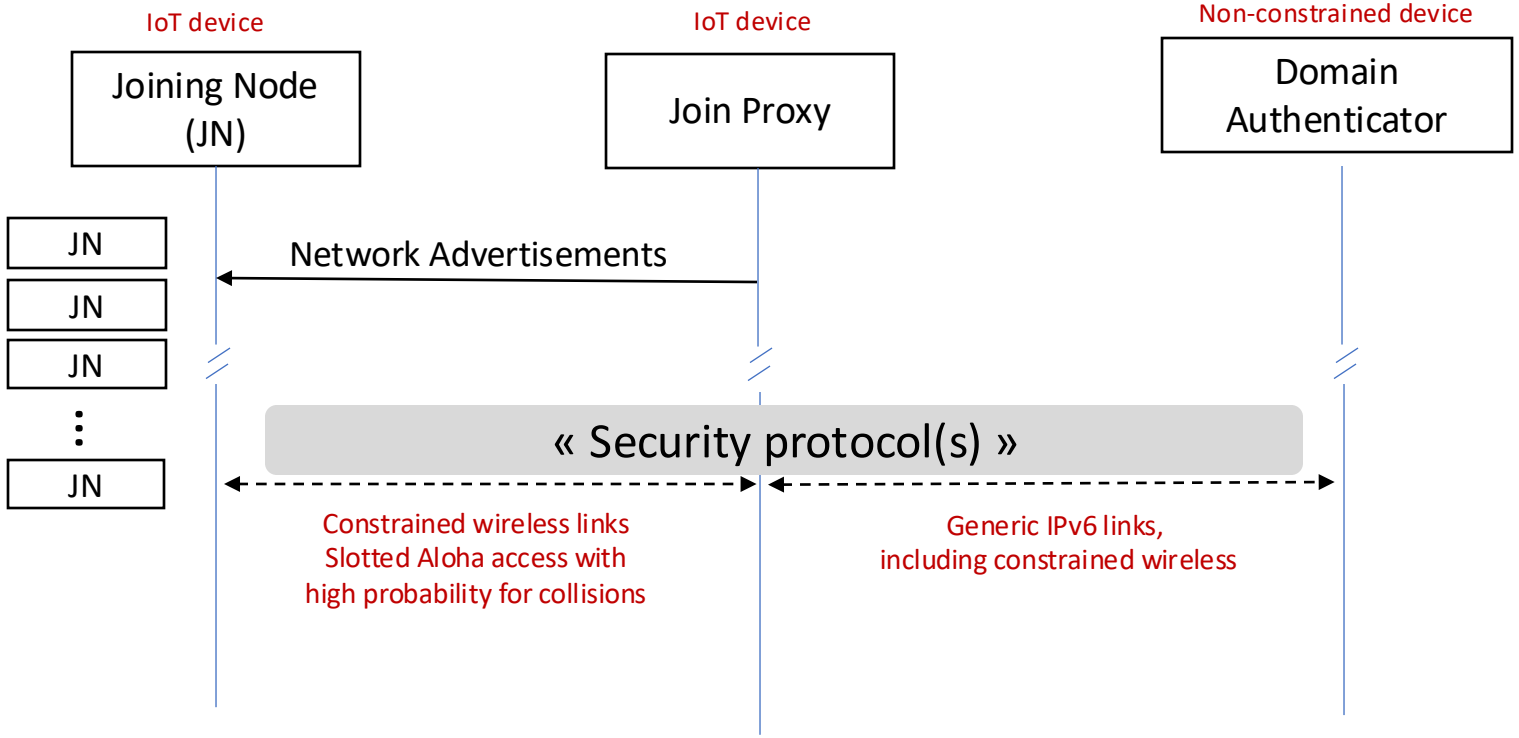
```
▼ Certificates (3065 bytes)
    Certificate Length: 1939
  ▼ Certificate: 3082078f30820677a003020102020842e179905e13d18c30...
    ▼ signedCertificate
        version: v3 (2)
        serialNumber: 4819266737223750028
      ▼ signature (sha256WithRSAEncryption)
            Algorithm Id: 1.2.840.113549.1.1.11 (sha256WithRSAEncryption)
      ▶ issuer: rdnSequence (0)
      ▶ validity
      ▶ subject: rdnSequence (0)
      ▶ subjectPublicKeyInfo
      ▶ extensions: 9 items
    ▼ algorithmIdentifier (sha256WithRSAEncryption)
        Algorithm Id: 1.2.840.113549.1.1.11 (sha256WithRSAEncryption)
      Padding: 0
      encrypted: 0cc439cc3f5c59686111082b8c5d76f7f4d52492fc0c0169...
    Certificate Length: 1120
  ▼ Certificate: 3082045c30820344a003020102020d01e3a9301cfc720638...
    ▼ signedCertificate
        version: v3 (2)
        serialNumber: 0x01e3a9301cfc7206383f9a531d
      ▼ signature (sha256WithRSAEncryption)
            Algorithm Id: 1.2.840.113549.1.1.11 (sha256WithRSAEncryption)
      ▶ issuer: rdnSequence (0)
      ▶ validity
      ▶ subject: rdnSequence (0)
      ▼ subjectPublicKeyInfo
```

>3kB of data!!!

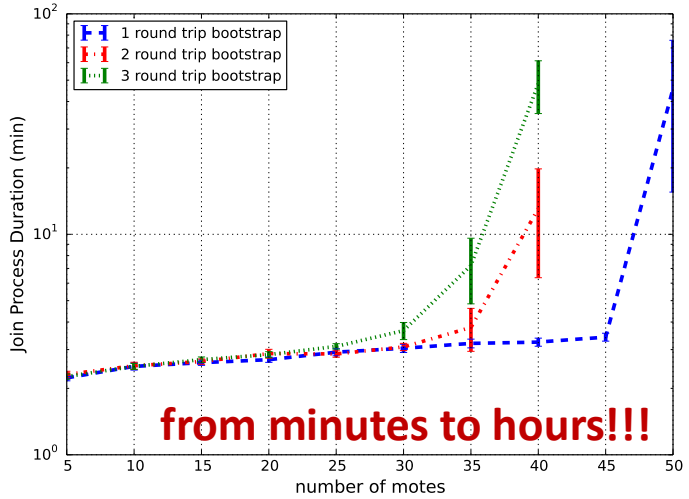
Existing Internet Security Technologies are Heavy

Network Formation Phase

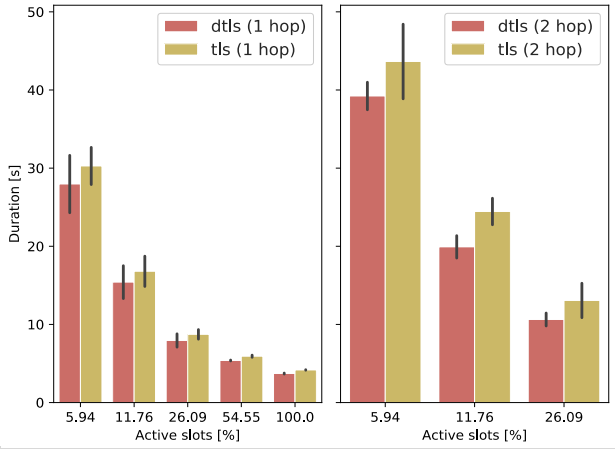
- ~100 B/s of shared bandwidth available
- Number of nodes joining
- Number of L2 **frames** exchanged for network access authentication



Time installers need to spend on site [1]



TLS performance in 6TiSCH [2]



[1] Vučinić, Mališa, et al. Broadcasting strategies in 6TiSCH networks. Internet Technology Letters, 2018.
[2] Claeys, Timothy et al. Performance of the Transport Layer Security Handshake over 6TiSCH. MDPI Sensors, 2021.

IETF is Working on a Lightweight Security Stack

- Internet Engineering Task Force
- Behind e.g. TCP/IP suite
- Open process, open standards (RFCs)
- 100+ working groups
- 7 areas
  - Applications and Real-time Area
  - General Area
  - Internet Area
  - Operations and Management Area
  - Routing Area
  - Security Area
  - Transport Area

INTERNET STANDARD
Errata Exist

Internet Engineering Task Force (IETF) — S. Deering
Request for Comments: 8200 — Retired
STD: 86 — R. Hinden
Obsoletes: 2460 — Check Point Software
Category: Standards Track — July 2017
ISSN: 2070-1721

Internet Protocol, Version 6 (IPv6) Specification

Abstract

This document specifies version 6 of the Internet Protocol (IPv6). It obsoletes RFC 2460.

Status of This Memo

This is an Internet Standards Track document.

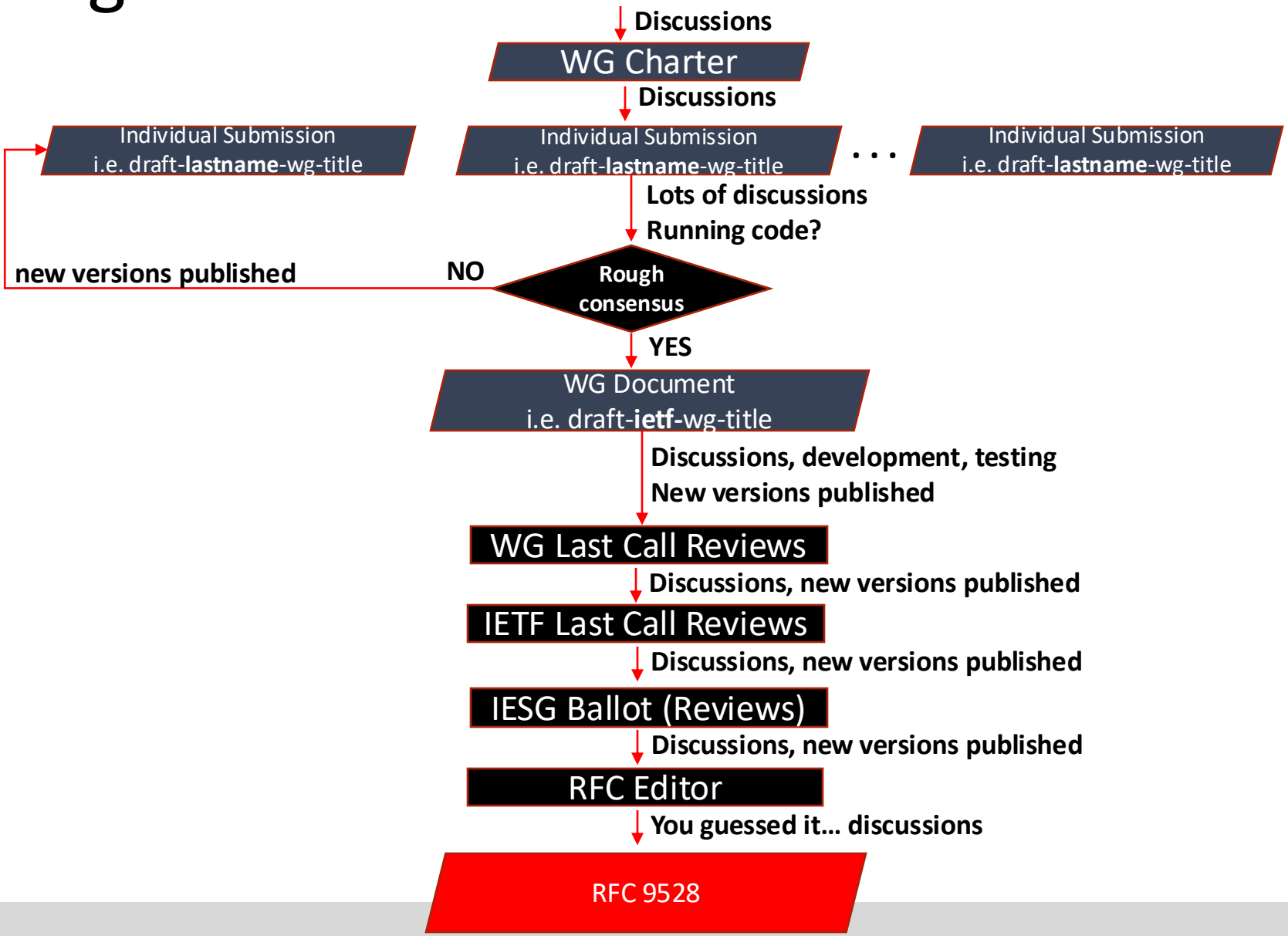
This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc8200.

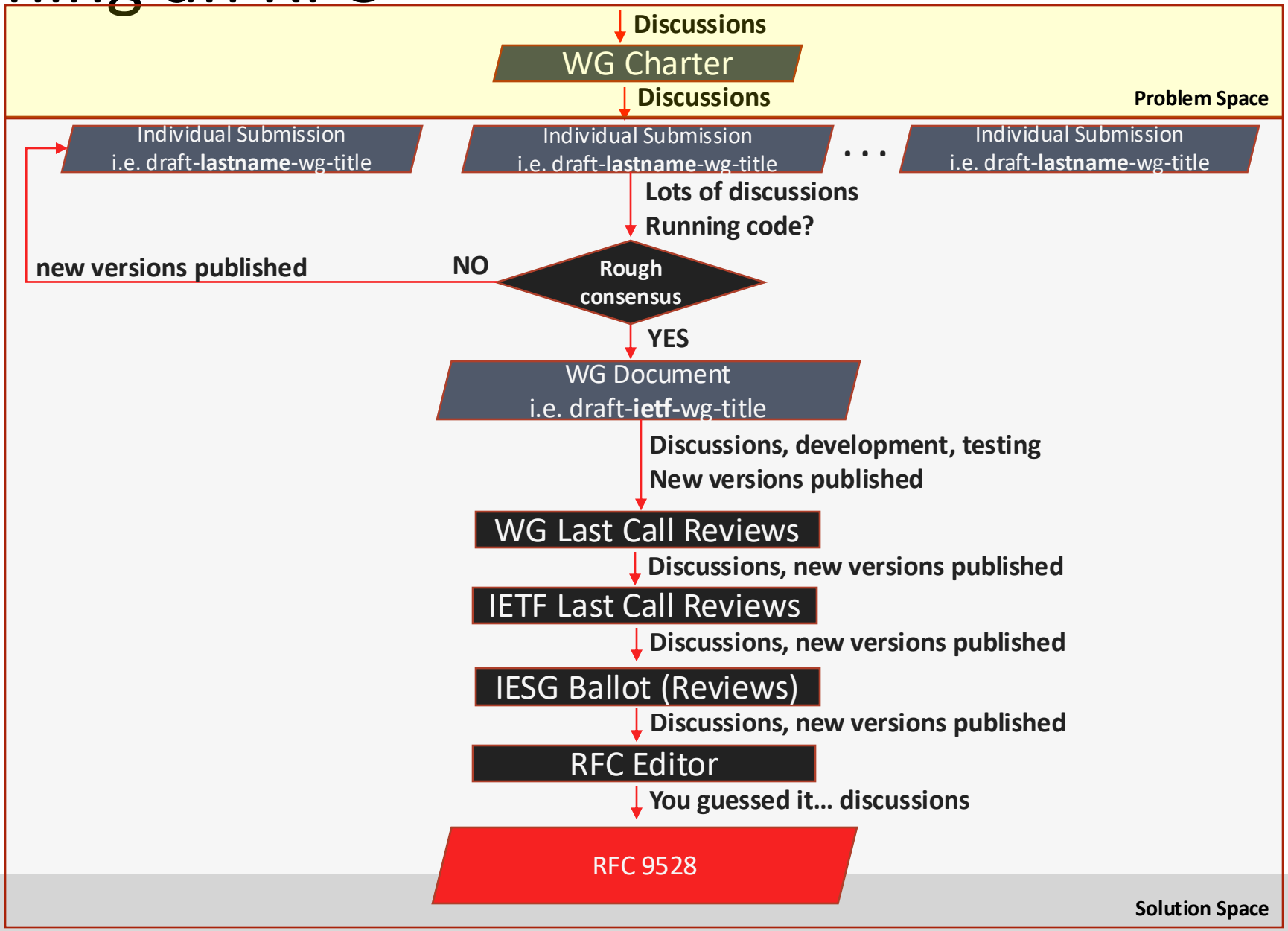
"We reject kings, presidents and voting. We believe in rough consensus and running code."

Dave Clark

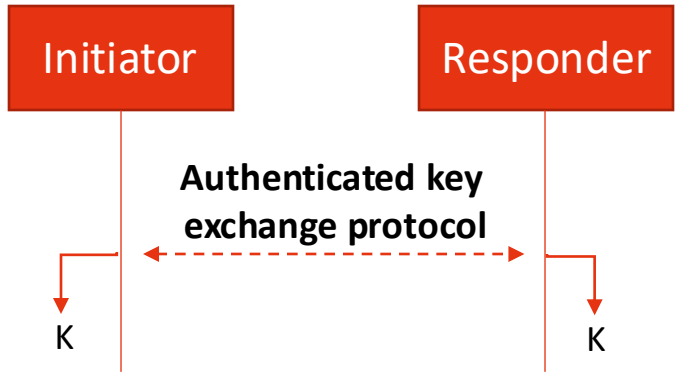
Publishing an RFC



Publishing an RFC



IETF LAKE



- IETF working group [1]
  - created November 2019
  - Co-chaired by Inria
  - lakewg.org
- LAKE: **L**ightweight **A**uthenticated **K**ey **E**xchange
- Authenticated key exchange for constrained environments
- Solution protocol is called **EDHOC** [2]

Overhead comparison [3]

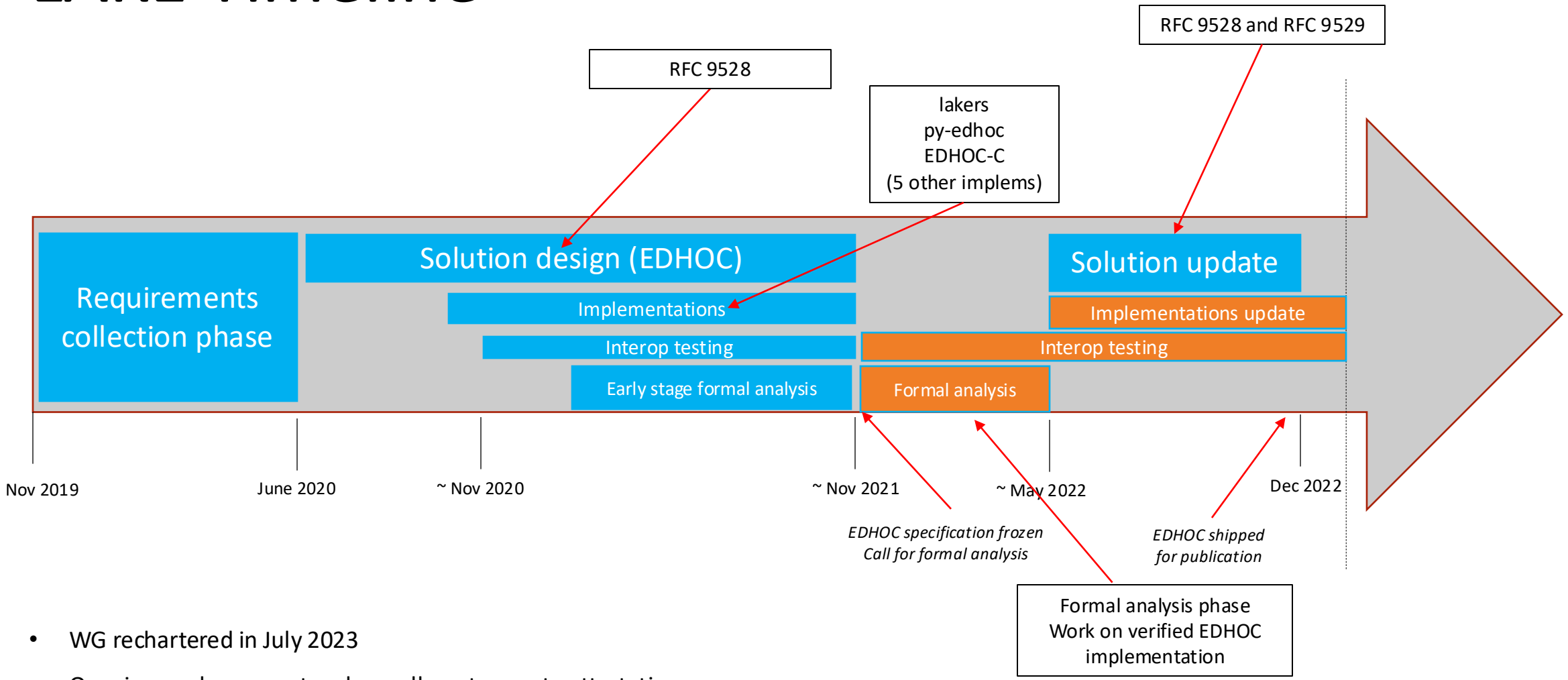
Flight	#1	#2	#3	Total
DTLS 1.3 – RPKs, ECDHE	185	454	255	894
DTLS 1.3 – Compressed RPKs, ECDHE	185	422	223	830
DTLS 1.3 – Cached RPK, PRK, ECDHE	224	402	255	881
DTLS 1.3 – Cached X.509, RPK, ECDHE	218	396	255	869
DTLS 1.3 – PSK, ECDHE	219	226	56	501
DTLS 1.3 – PSK	136	153	56	345
EDHOC – X.509s, Signature, x5t, ECDHE	37	115	90	242
EDHOC – RPKs, Signature, kid, ECDHE	37	102	77	216
EDHOC – X.509s, Static DH, x5t, ECDHE	37	58	33	128
EDHOC – RPKs, Static DH, kid, ECDHE	37	45	19	101

What does "lightweight" mean?

Metric	Current status
Number of round trips to complete	3 messages, 2 round trips
Bytes on the wire	101 bytes: 37 + 45 + 19
Wall-clock time to complete	Impacted by radio technology
The amount of new code	9-10 kB of flash, 2 kB of RAM

[1] https://datatracker.ietf.org/wg/lake/about/
[2] https://datatracker.ietf.org/doc/draft-ietf-lake-edhoc/
[3] https://datatracker.ietf.org/doc/draft-ietf-lwig-security-protocol-comparison

LAKE Timeline



- WG rechartered in July 2023
- Ongoing work on zero-touch enrollment, remote attestation, ….

LAKE Security Goals

Goal	Description
Mutual authentication	Agree on fresh session ID, roles and credentials of each peer
Confidentiality	Derived key known only to the two peers; forward secrecy
Downgrade protection	Agree on crypto algs proposed and those chosen
Security level	>= 127 bits: strength of authentication, established keys and downgrade protection
Identity protection	Protect identity of one peer against active attacks, the other identity against passive
Protection of External Data	External data protected to the same level as the message it is carried within

LAKE Solution: **E**phemeral **D**iffie-**H**ellman **o**ver **C**OSE (1/2)

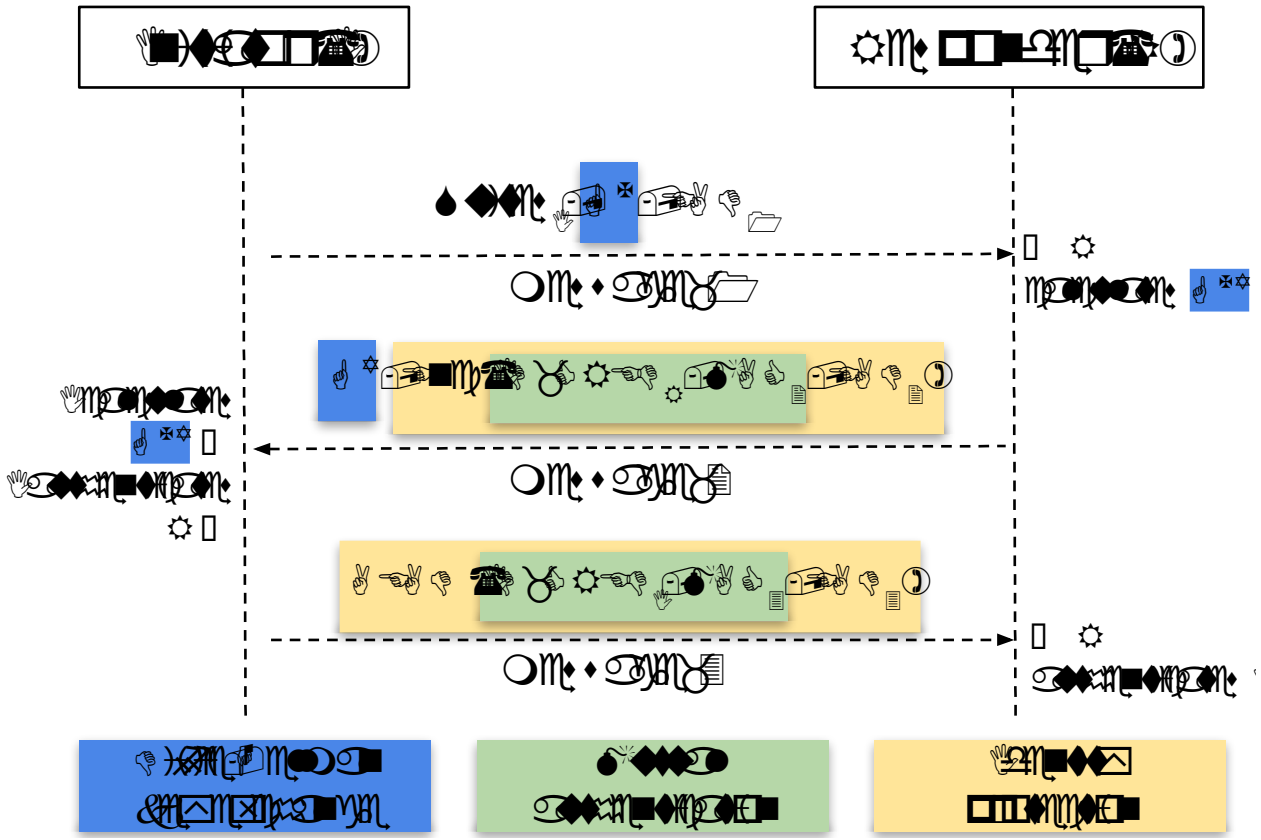
- Authentication credentials
  - Conventional signature keys
  - Support for static Diffie-Hellman keys
  - Transport of certificates by “reference”
- Crypto agility
  - Suites based on secp256r1, X25519, …
- Forward secrecy
  - Exchange of ephemeral Diffie-Hellman keys
- Protocol design based on SIGMA-I
- Compact encoding with CBOR

EDHOC Authentication Modes

ID	Initiator authentication key	Responder authentication key
0	Signature	Signature
1	Signature	Static Diffie-Hellman
2	Static Diffie-Hellman	Signature
3	Static Diffie-Hellman	Static Diffie-Hellman
TBD	Pre-shared symmetric key	Pre-shared symmetric key

Ongoing work of Elsa Lopez Perez et al.

EDHOC: **E**phemeral **D**iffie-**H**ellman **o**ver **C**OSE (2/2)



Static-Static authentication mode

Term	Description
$Suites_I$	Cipher suite supported and selected by I
G^X, G^Y	Ephemeral keys of I and R
ID_CRED_R, ID_CRED_I	Identifier or full credential of I and R
MAC_2, MAC_3	Message authentication code in messages 2 and 2
EAD_1, EAD_2, EAD_3	External authorization data

Security Analysis of EDHOC

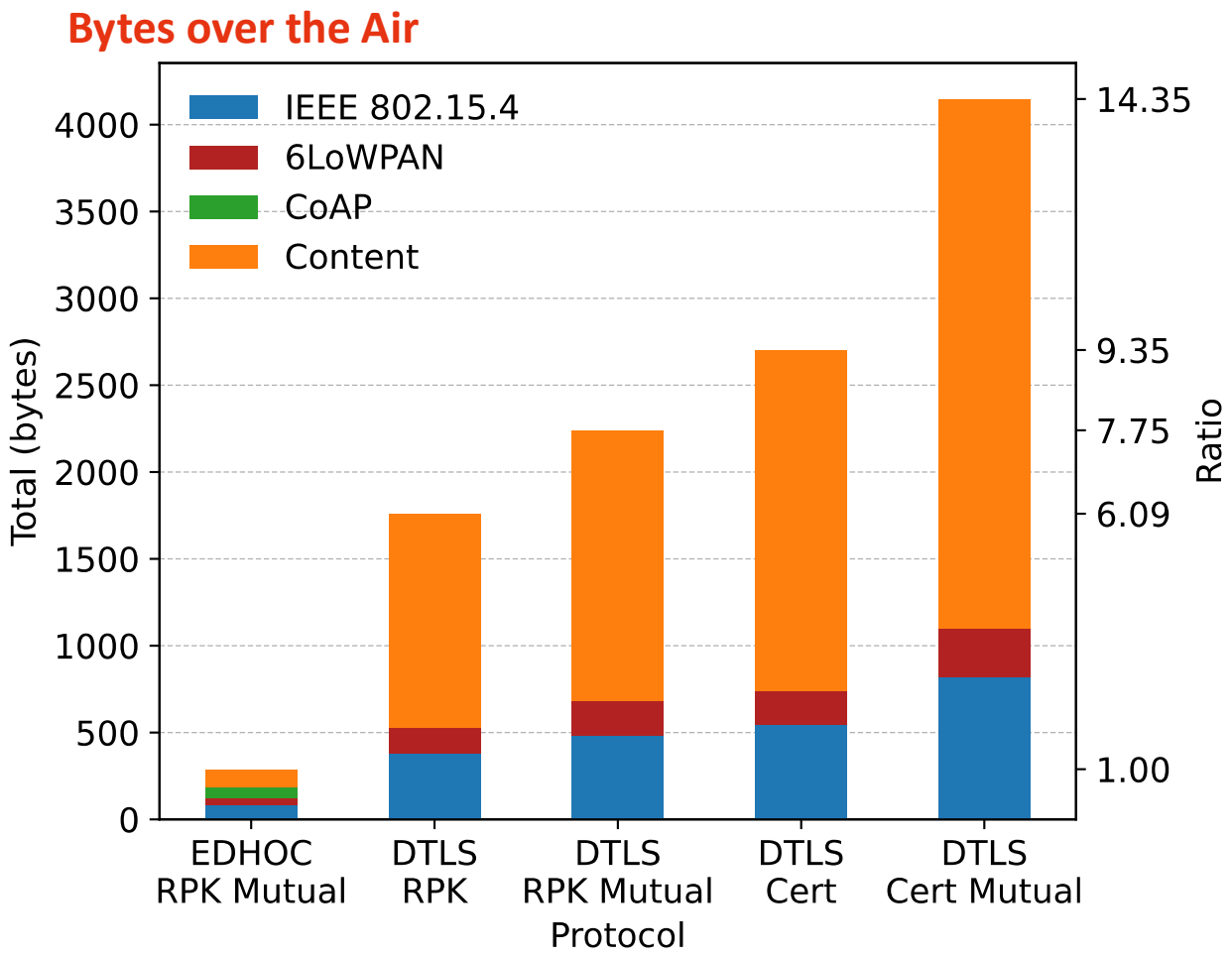
- ”Formal analysis” phase in the development
  - From November ’21 to May ‘22
- Academic community invited to study the protocol [7]
- 4 teams responded
  - ETH Zurich
  - École normale supérieure
  - Inria Nancy
  - Norrman et al.
- Studies in symbolic and computational model
- Improvements incorporated in the protocol design

Security goal	Vulnerability/Attack	Proposal	Initial draft	Improved draft	Method	Proof type	Ref.
Confidentiality	Weak final key	Final key depending on PRK_{4x3m} and TH_4 (PRK_{out})	12	14	0-1-2-3	S	[4]
	Transcript collision attack	Reorder arguments in the hash function	12	14	0-1-2-3	S	[4]
	Duplicate Signature Key Selection (identity misbinding attack)	Include full/unique authentication credentials in the hash function. Build transcript hashes based on plaintext	14	17	0	S	[3]
	Key reuse	Not to reuse keys across HKDF calls of Extract and Expand	17	17	0	C	[3]
	Reuse of the last key-exchange–internal key	Use a dedicated session key (PRK_{out})	14	14	0	C	[3]
	Salt Collision Attack	Use TH_2 as salt in the HKDF Extract function to derive PRK_{2e}	15	?	3	C	[1]
	Weak data authentication		12	14	0-1-2-3	S	[4]
Mutual authentication	KCI	Modify the construction of message 3	15	?	3	C	[1]
	Leaking ephemeral secrets breaks authentication	Entity authentication should only rely on long-term authentication secrets	12	14	0-1-2-3	S	[4]
	Injective agreement	Add a fourth message as an option	1	?	0-1-2-3	S	[6]
Identity protection	Initiator impersonation		12	14	0-1-2-3	S	[4]
Security level	Attacks in 2^{64} operations for the Responder	Add a fourth message	15	?	3	C	[2]
Protection of external data	AEAD Key/IV reuse	Do not allow message recomputation from stored data	12	14	1-2-3-4	S	[4]

References

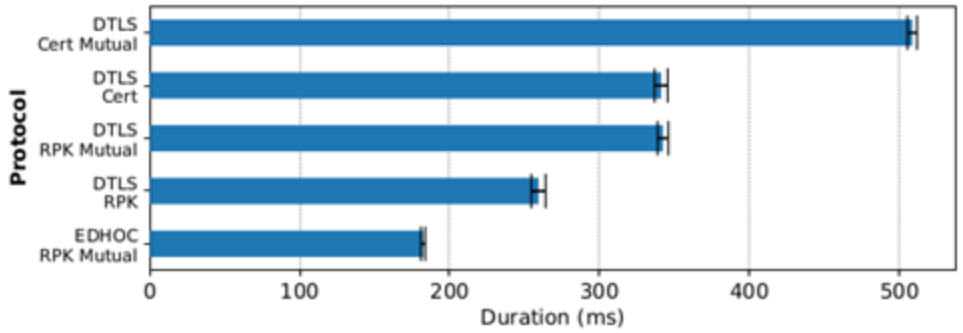
1. Cottier and Pointcheval. Security analysis of the EDHOC protocol. 2022.
2. Cottier and Pointcheval. Security analysis of improved EDHOC protocol. In *FPS 2022*.
3. Günther and Tshibumbu Mukendi. Careful with MAC-then-sign: A computational analysis of the EDHOC lightweight authenticated key exchange protocol. In *Euro S&P 2023*.
4. Jacomme et al. A comprehensive, formal and automated analysis of the EDHOC protocol. In *USENIX Security 2023*.
6. Norrman et al. Formal analysis of EDHOC key establishment for constrained IoT devices. In *International Conference on Security and Cryptography 2021*.
7. Vučinić et al. Lightweight authenticated key exchange with EDHOC. IEEE Computer, 2022.

Performance of EDHOC (1/2)

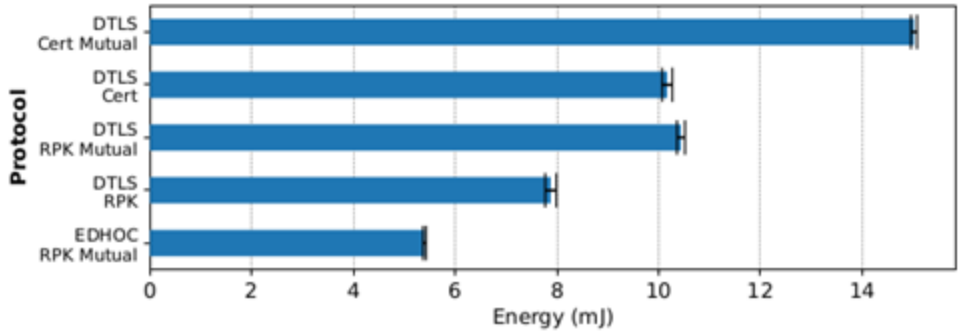


[1] Performance Comparison of EDHOC and DTLS 1.3 in Internet-of-Things Environments. Geovane Fedrecheski, Mališa Vučinić, Thomas Watteyne. Submitted to: IEEE Wireless Communications and Networking Conference (WCNC), 2024.

Performance of EDHOC (2/2)

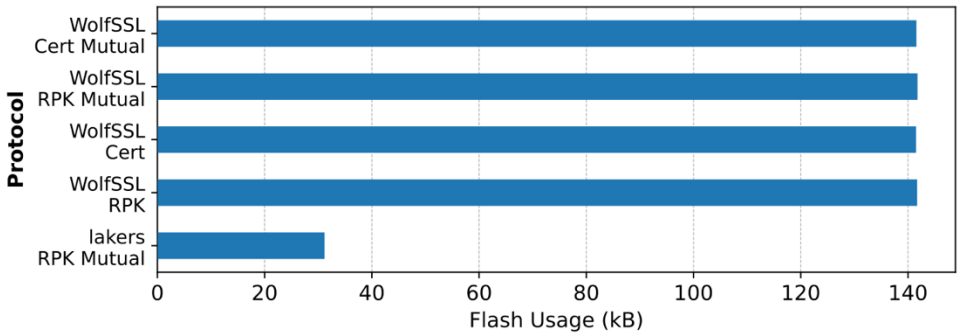


(a) Handshake Duration

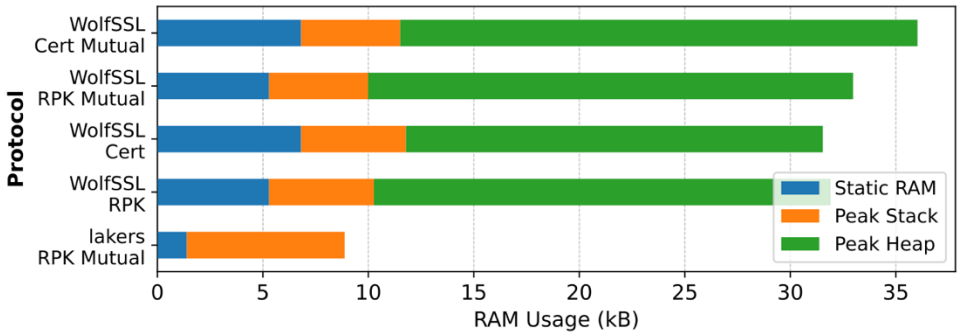


(b) Energy Consumption

Fig. 4: Results of time and energy measurements for a handshake under several configurations.



(a) Flash memory



(b) RAM.

Fig. 6: Flash memory and RAM usage for the `lakers` and `wolfSSL` implementations.

[1] Performance Comparison of EDHOC and DTLS 1.3 in Internet-of-Things Environments. Geovane Fedrecheski, Mališa Vučinić, Thomas Watteyne. Submitted to: IEEE Wireless Communications and Networking Conference (WCNC), 2024.

20

The Ecosystem and Next Steps

Implementations and interop testing

- 7+ implementations
  - C, Python, Rust, Java
  - 3 led by Inria
- 6 interop testing events organized
  - From Feb. 2021 to Dec. 2022
- Products are already on the market!
- More info at lakewg.org

Related work

- OSCORE (RFC8613) for message protection
- Group OSCORE
- Secure zero-touch onboarding with EDHOC [1]
- OSCORE-based certificate enrollment [2]

Next steps

- Authentication based on Pre-Shared Keys (PSKs)
- Remote attestation over EDHOC [3]

[1] https://datatracker.ietf.org/doc/draft-ietf-lake-authz/
[2] https://datatracker.ietf.org/doc/draft-ietf-ace-coap-est-oscore/
[3] https://datatracker.ietf.org/doc/draft-ietf-lake-ra

EDHOC with PSKs: Why?

- Leveraging existing infrastructure
  - Billions of PSK-provisioned devices deployed worldwide
  - Cost-effective: update software rather than replace hardware
  - Preserves the investments in current technology
- Gradual transition
  - EDHOC + PKS supports adoption by older devices
  - Bridges the gap between current and future security protocols
  - Ensures backward compatibility while moving forward

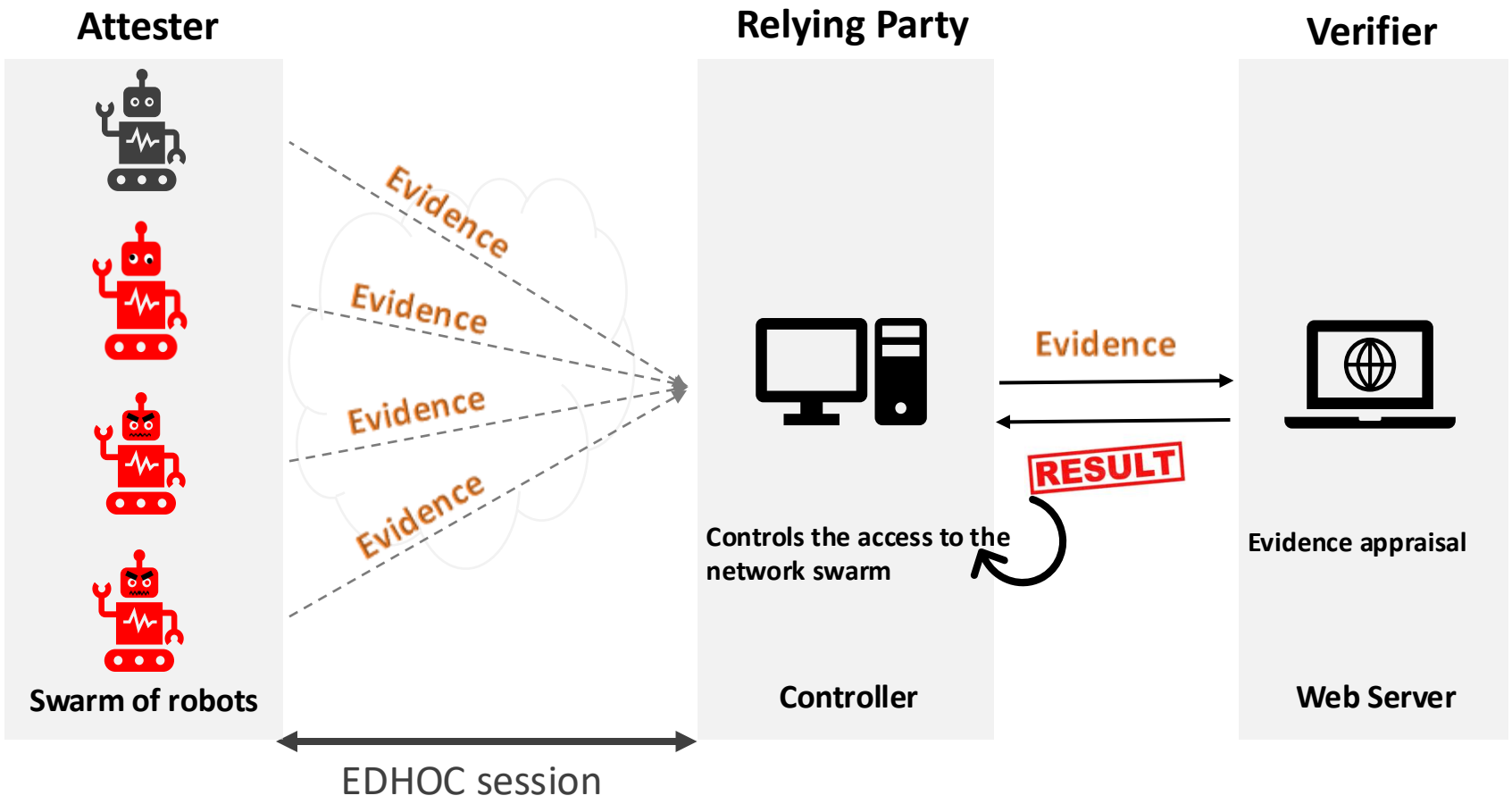
- EDHOC with PSK offers a practical, cost-effective path to enhance IoT security while utilizing existing infrastructure

Remote Attestation

Remote attestation is a security service to verify and confirm the integrity and trustworthiness of a remote device or system.

- Evidence: a set of Claims to demonstrate the integrity and security properties of its software or hardware.
- Attestation result: the output after evaluating the validity of Evidence
- Relying Party: the entity who consumes the Attestation result to reliably apply application-specific actions

EDHOC with Remote Attestation



Conclusion

- EDHOC is RFC 9528 and RFC 9529
- Authenticated key exchange protocol
- Designed for constrained IoT use cases
- Total handshake footprint can be as low as 101 bytes
- Products are available on the market!

Hvala!*

*Thank you!

Inria