Application Layer Security for the Internet of Things

CASTOR Software Days 15-10-2019

Francesca Palombini, Ericsson Research

I E T F®

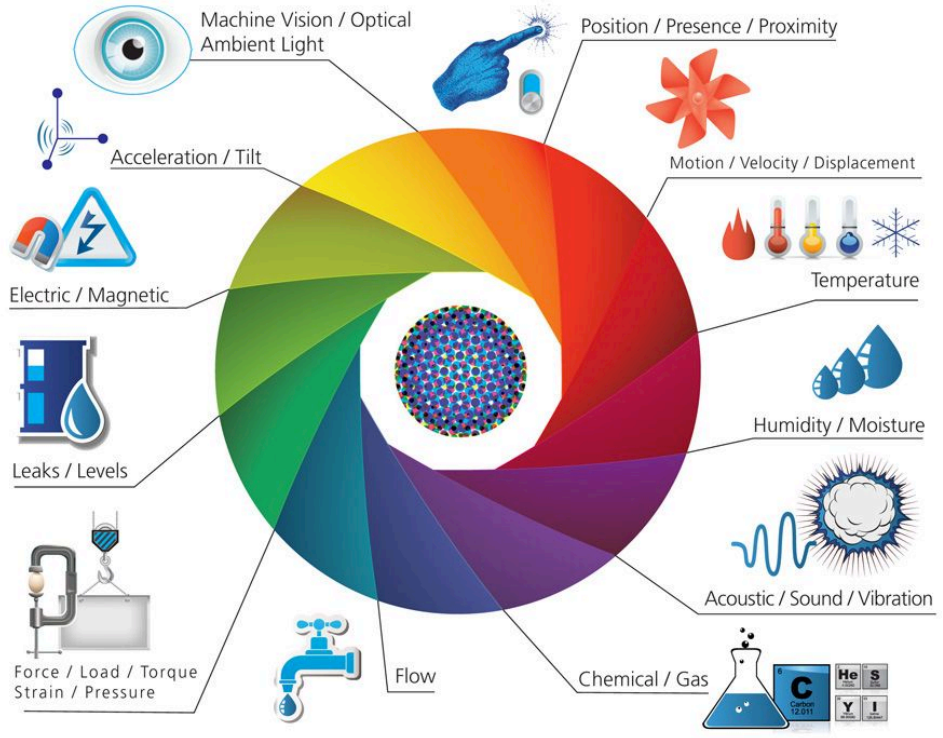
HTTP TCP QUIC

TLS IP

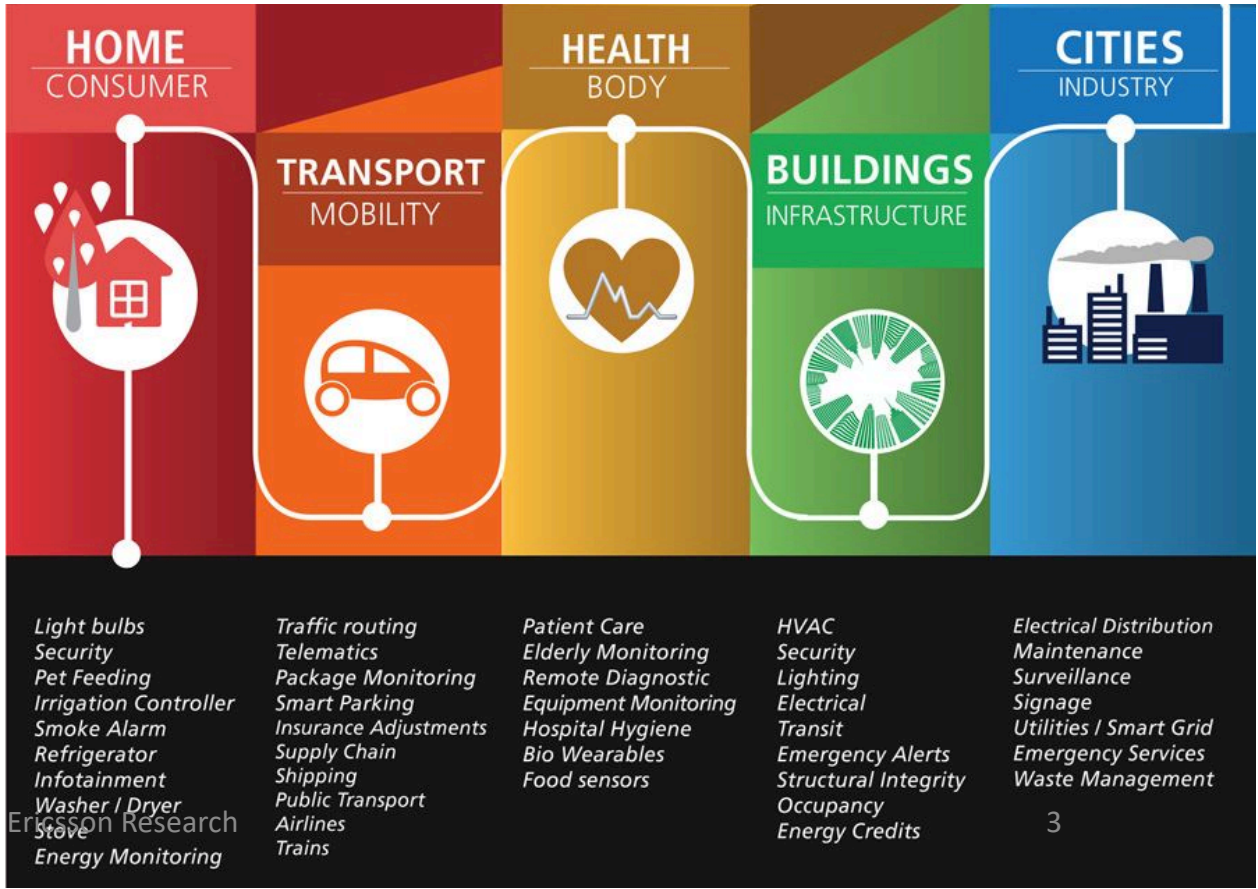
… and many more

The IoT is happening

Gartner Says 5.8 Billion Enterprise and Automotive IoT Endpoints Will Be in Use in 2020
https://www.gartner.com/en/newsroom/press-releases/2019-08-29-gartner-says-5-8-billion-enterprise-and-automotive-io



Pictures credit of Postscapes / Harbour Research



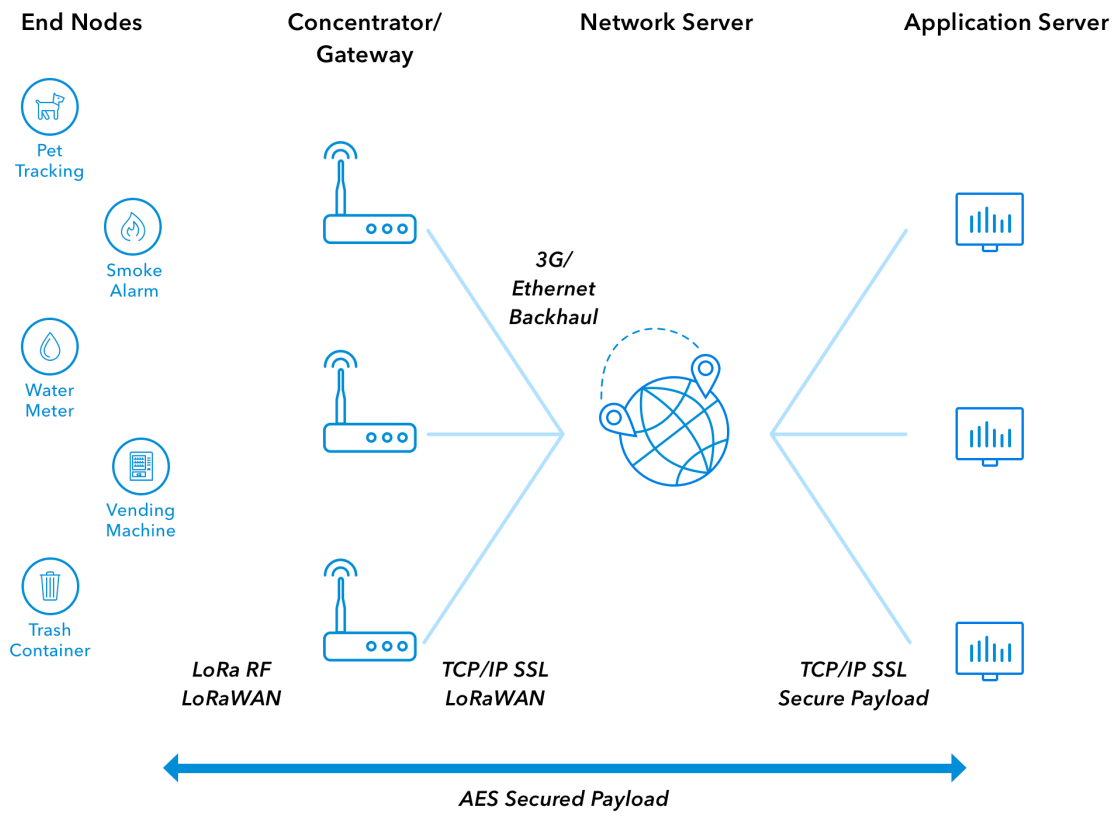
The “T” in “IoT”

- maximum code complexity (ROM/Flash)
- size of state and buffers (RAM),
- amount of computation feasible in a period of time ("processing power"),
- available power
- user interface and accessibility in deployment (ability to set keys, update software, etc.).

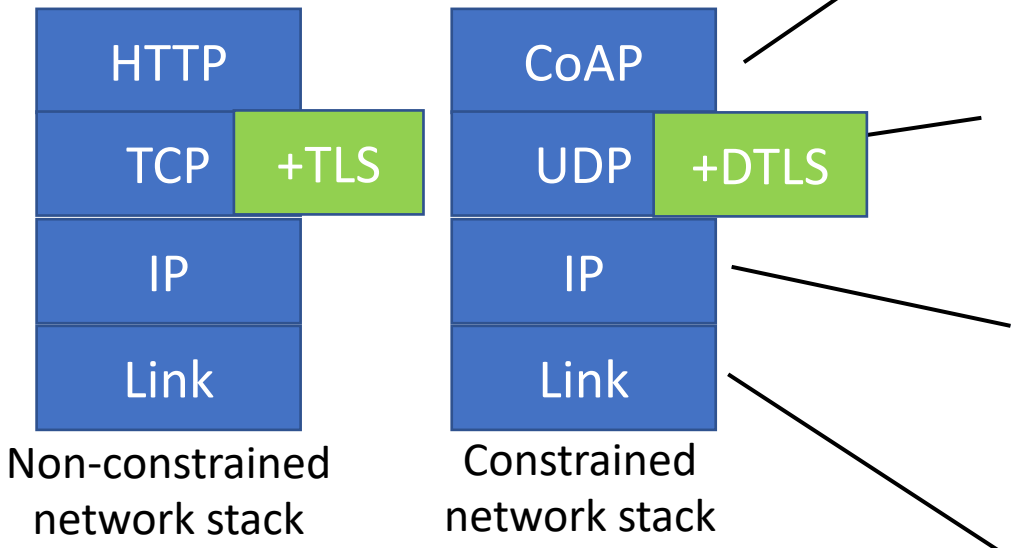
Name	data size (e.g., RAM)	code size (e.g., Flash)
Class 0, C0	<< 10 KiB	<< 100 KiB
Class 1, C1	~ 10 KiB	~ 100 KiB
Class 2, C2	~ 50 KiB	~ 250 KiB

Terminology for Constrained-Node Networks, RFC7228

Example: LoRaWAN architecture



The “I” in “IoT” – Internet Protocol Stack



Constrained Application Protocol – RFC7252
Maps to HTTP, optimized for constrained

TCP is too expensive for constrained node networks, UDP is used instead. DTLS is developed to run over UDP.

Including an adaptation layer protocol such as 6LoWPAN, 6TiSCH, IPv6 over BLE

LoRaWAN, Zigbee (over IEEE802.15.4), Bluetooth LE, NB-IoT …

The “s” in IoT stands for “security”

Strong coupling to physical assets
→ Intrinsic safety and privacy risks

The number of things
→ Distributed Denial-of-Service

The hype and low security incentives
→ Common insecure deployments

Power constrained and sleepy devices
→ Security overhead challenges

Gateways and proxies
→ End-to-end security challenges

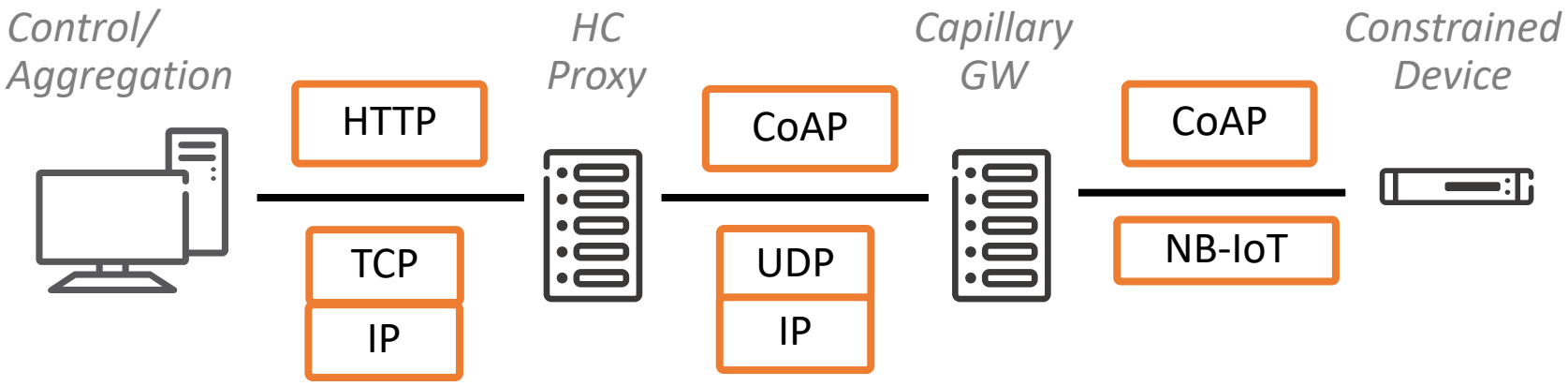
Need for end-to-end security in constrained environments

Example scenario

End-to-end aspects
- Endpoints
- Transport layer protocols
- Application layer protocols
- Intermediaries

Constrainedness
- Message overhead
- Round-trips
- Public-key operations

Security aspects
- Encryption
- Integrity and replay protection
- Authentication
- Authorization

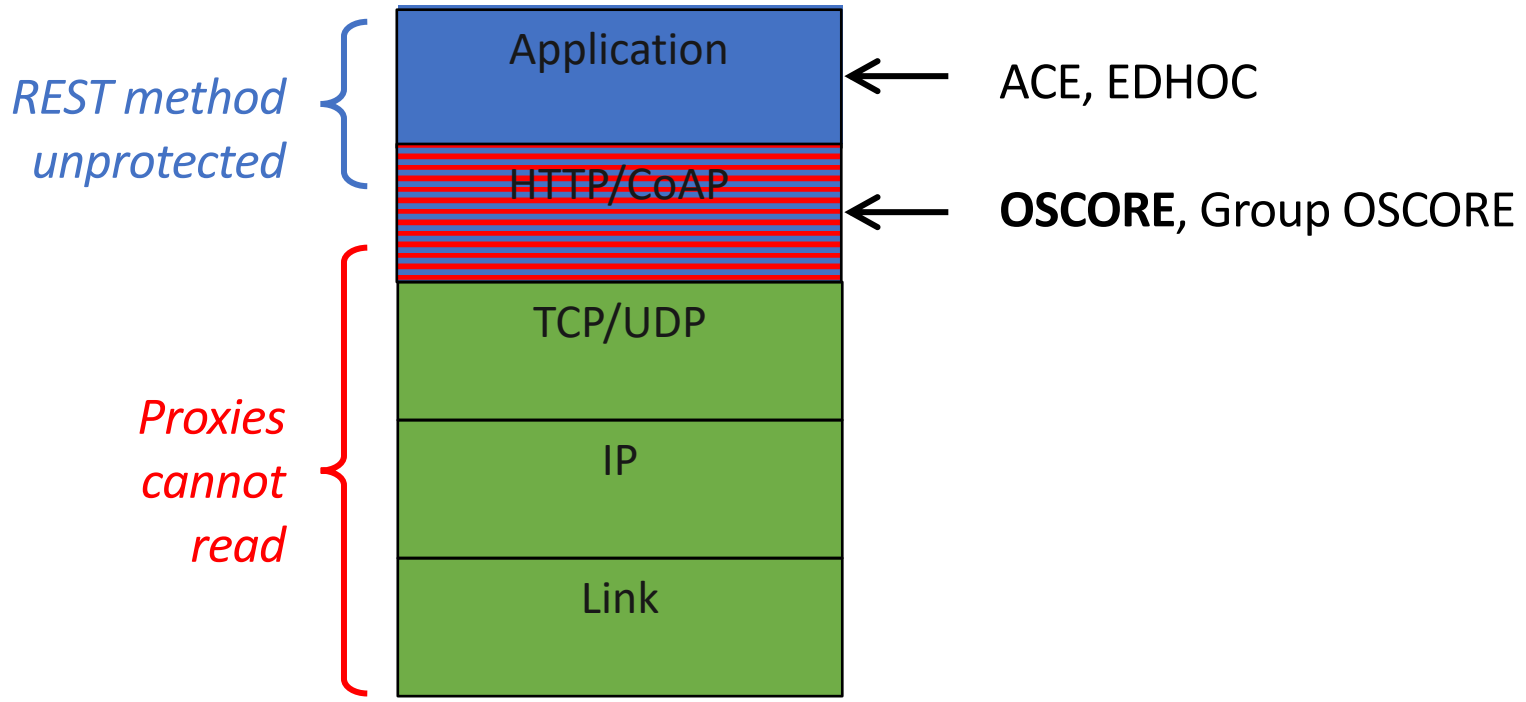


Application Layer Security for the IoT

Requirements

- Independence of transport layer
- Support for proxy operations
- Protection of REST operations (HTTP/ CoAP)
- Optimized for constrained devices
- Standardized
- Wide applicability

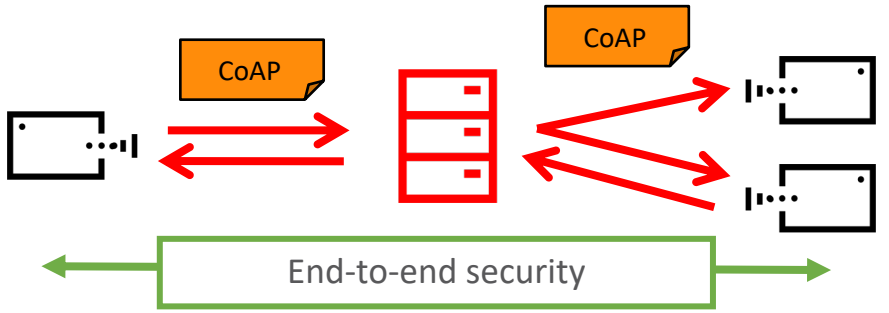
Security layer?

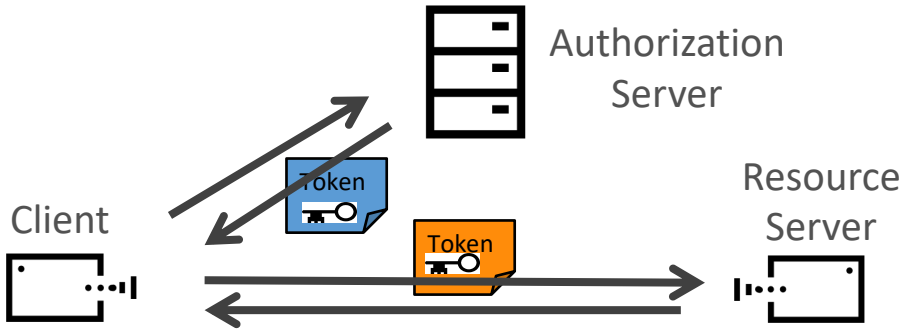


New IoT security protocols

New IoT Security Protocols

- COSE (CBOR Object Signing and Encryption): Secure message format based on binary data format CBOR (small)
- OSCORE: Lightweight communication security protocol (once keys are in place)
- Group OSCORE: OSCORE in groups, adds signature
- EDHOC: Lightweight Diffie-Hellman key exchange
  To securely develop shared secrets to derive keys for OSCORE
- ACE: Lightweight authorization and access control
  A delegation protocol to convey authorization, enables a client to obtain scoped access to a resource





Object Security for Constrained Restful Environments - OSCORE

Application layer communication security protocol

- Object Security for Constrained RESTful Environments
- Protecting CoAP, HTTP, LwM2M
- End-to-end encryption, integrity and replay protection

Designed for constrained IoT deployments

— Low overhead (minimum 10-15 bytes)

— Low footprint in addition to CoAP

— Independent of transport layer

— Supports multicast and group communication

Developed by Ericsson Research in collaboration with RISE SICS

— Standardized in the IETF

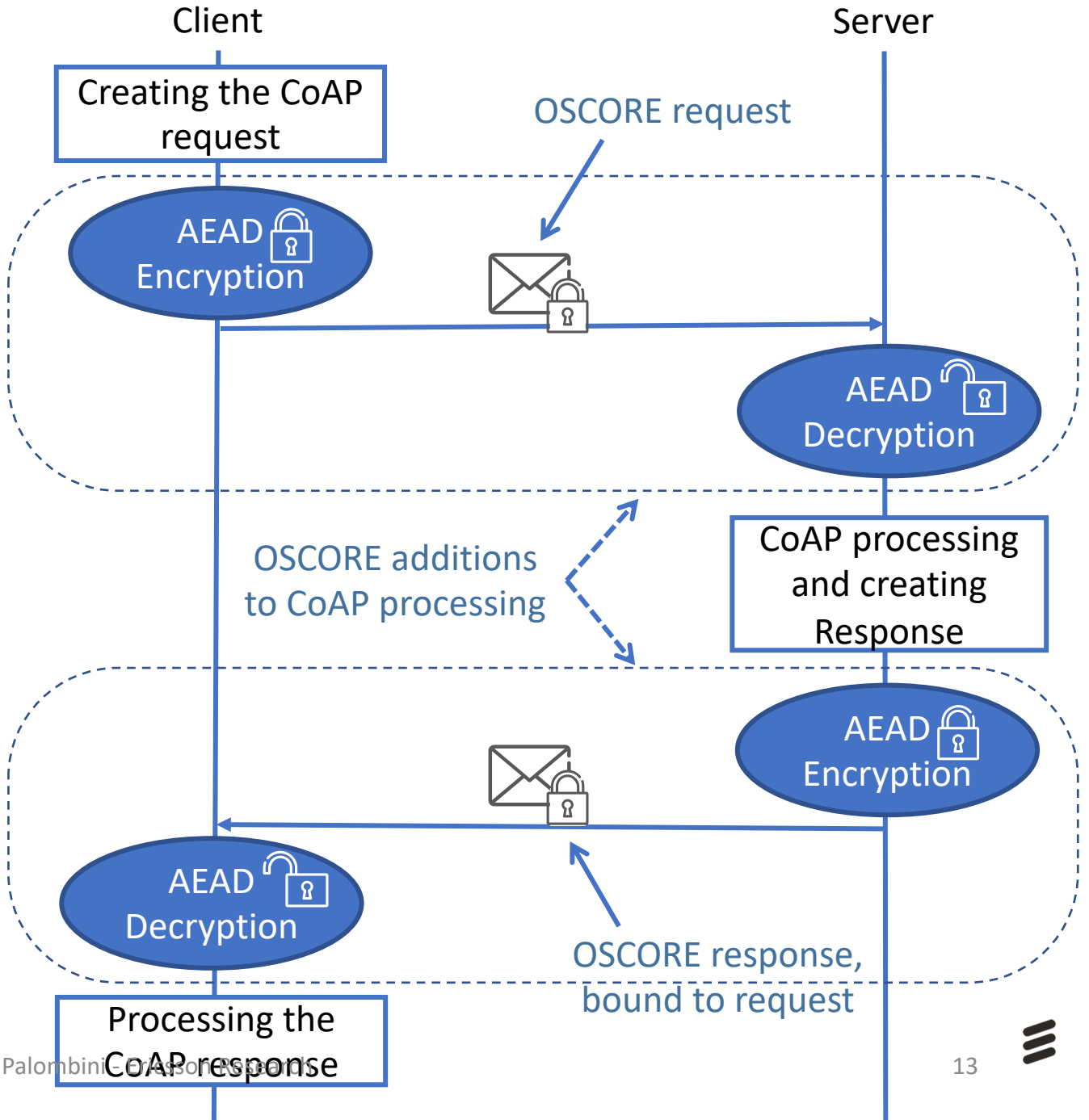
— Adopted by LwM2M, OCF, Fairhair Alliance

— Open source Eclipse implementation in progress

— To be included in Ericsson IoT Accelerator

OSCORE Processing

- Addition to CoAP
- Uses Authenticated Encryption with Additional Data (AEAD)
- AES-128-CCM-8 mandatory to implement
- Protection of CoAP messages using the COSE format
- Replay protection
- Handling partial loss of security context



OSCORE Message Overhead

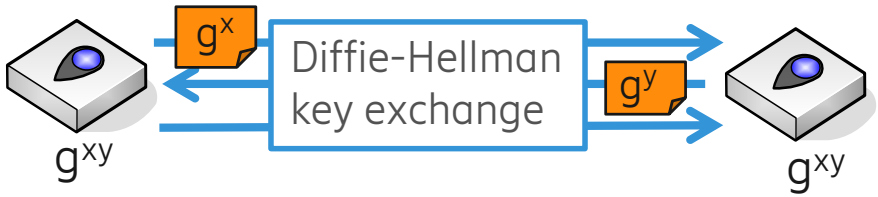
Protocol	*Overhead (B) for Sequence Number = '05'*	*Overhead (B) for Sequence Number = '1005'*	*Overhead (B) for Sequence Number = '100005'*
DTLS 1.2	29	29	29
DTLS 1.3(work in progress)	11	12	12
TLS 1.2	21	21	21
TLS 1.3	14	14	14
DTLS 1.2 (GHC)	16	16	16
DTLS 1.3 (GHC) (wip)	12	13	13
TLS 1.2 (GHC)	17	18	19
TLS 1.3 (GHC) (wip)	15	16	17
OSCORE Request	13	14	15
OSCORE Response	11	11	11

https://tools.ietf.org/html/draft-ietf-lwig-security-protocol-comparison

EDHOC
Lightweight Key Exchange on Application Layer

Status

- Formal review by Univ. of Copenhagen
- Constrained implementation by Univ. of Murcia
- Significant reduction of overhead
- Mature specification
- Good support in the IoT community (6tisch, NB-IoT, LoRaWAN)



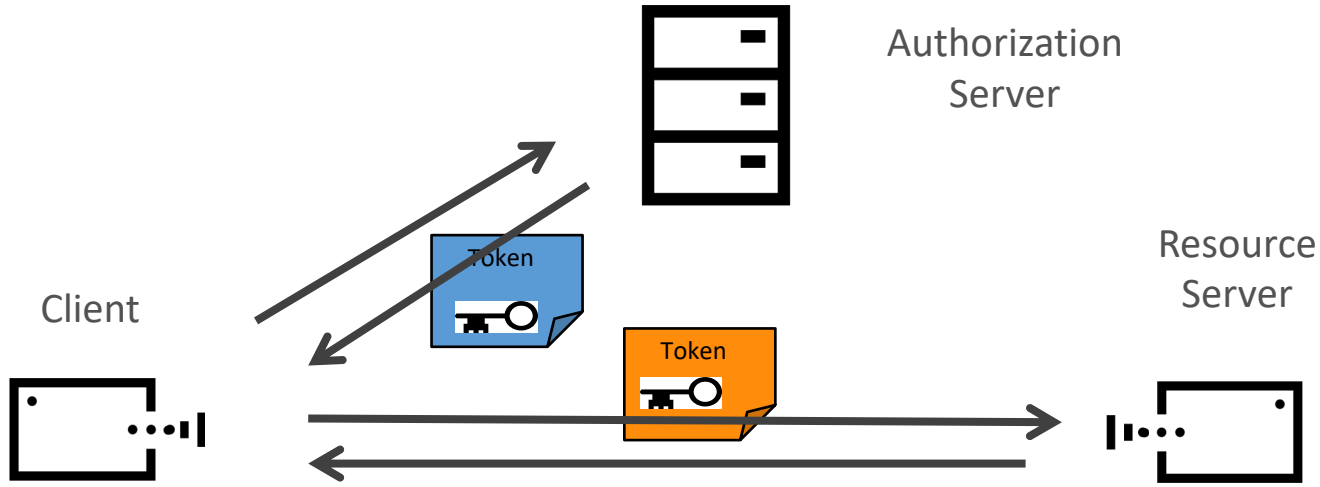
Flight	#1	#2	#3	Total
DTLS 1.3 RPK + ECDHE	149	373	213	735
DTLS 1.3 PSK + ECDHE	186	190	57	433
DTLS 1.3 PSK	136	150	57	343
EDHOC RPK + ECDHE	38	121	86	245
EDHOC PSK + ECDHE	43	47	12	102

3x

4x

Authentication and Authorization for Constrained Environments (ACE)

ACE: Lightweight authorization and access control
A profile of OAuth 2.0 using CBOR and COSE secure objects, runs over CoAP



1.Client acquires Access Token from Authorization Server

1.Client presents Access Token to Resource Server to get access

Standardization and Implementation Status

- OSCORE is an IETF standard: RFC8613
- Several implementations exist, for several CoAP libraries: C, C#, Java, Python
- Interoperability tests have been run

- OSCORE group communication is in progress, and implementations are being developed

- EDHOC is a work in progress at IETF
- Partial implementations exist, formal verification has been done

- ACE is soon to be published as RFC
- A couple of implementations exist and have run interoperability tests

Key takeaways

IoT is happening

Security is important

Former security solutions are not optimized

We are working on it!