Freie Universität Berlin

HAW HAMBURG

Toward a RESTful Information-Centric Web of Things: A Deeper Look at Data Orientation in CoAP

ACM ICN 2020, Virtual Event, Canada

Cenk Gündoğan[1] Christian Amsüss

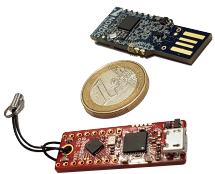
Thomas C. Schmidt[1] Matthias Wählisch[2]

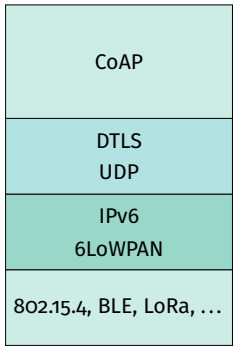
[1]HAW Hamburg [2]Freie Universität Berlin

cenk.guendogan@haw-hamburg.de christian@amsuess.com t.schmidt@haw-hamburg.de m.waehlisch@fu-berlin.de

Common Web of Things Deployments

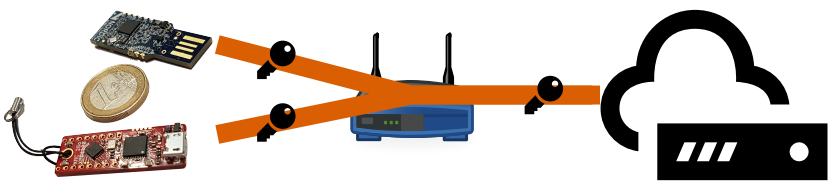
- Constrained IoT devices, gateway, cloud services

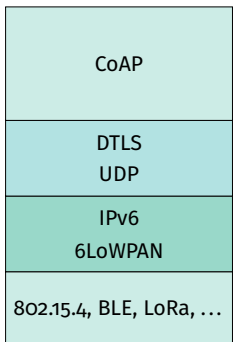




Common Web of Things Deployments

- Constrained IoT devices, gateway, cloud services
- RESTful and secured end-to-end connectivity





Common Web of Things Deployments

- Constrained IoT devices, gateway, cloud services
- RESTful and secured end-to-end connectivity
- Stateful proxying and caching on gateway

Proxy	
CoAP	HTTP
DTLS	TLS
UDP	TCP
IPv6	
6LoWPAN	
802.15.4, BLE, LoRa, …	

Common Web of Things Deployments

- Constrained IoT devices, gateway, cloud services
- RESTful and secured end-to-end connectivity
- Stateful proxying and caching on gateway
- Content object security between endpoints

Proxy	
CoAP	HTTP
OSCORE	
DTLS	TLS
UDP	TCP
IPv6	
6LoWPAN	
802.15.4, BLE, LoRa, …	

Common Web of Things Deployments

- Constrained IoT devices, gateway, cloud services
- RESTful and secured end-to-end connectivity
- **Stateful proxying** and **caching** on gateway
- **Content object security** between endpoints

Proxy	
CoAP	HTTP
OSCORE	
DTLS	TLS
UDP	TCP
IPv6	
6LoWPAN	
802.15.4, BLE, LoRa, …	

Standard CoAP features enable information-centric properties

Can we build a RESTful Web of Things with CoAP and information-centric properties?

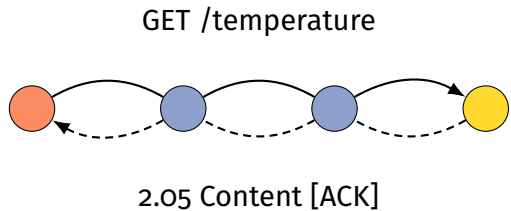
How will it perform?

Outline

Constructing an Information-centric Web of Things

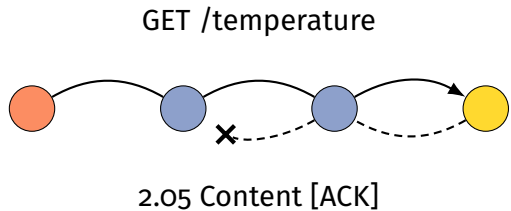
Baseline: Plain CoAP Setup

- Request–Response **GET** method
- Piggybacked acknowledgments



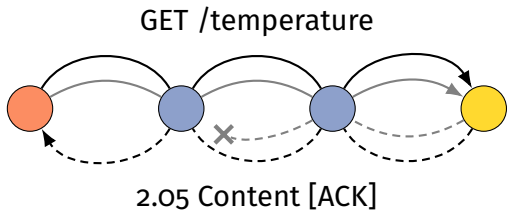
Baseline: Plain CoAP Setup

- Request–Response **GET** method
- Piggybacked acknowledgments
- Message timeouts at endpoints



Baseline: Plain CoAP Setup

- Request–Response **GET** method
- Piggybacked acknowledgments
- Message timeouts at endpoints
- End-to-end retransmissions



Information-centric Properties with Plain CoAP

- ☒ **Stateful forwarding**
- ☒ **Content caching**
- ☒ **Decoupling of data from location**
- ☒ **Content object security**

Extended CoAP Setup

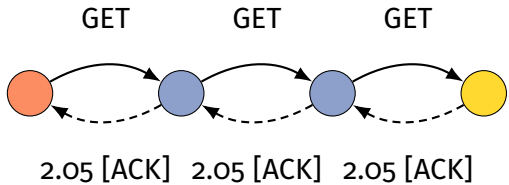
GET /temperature

2.05 Content [ACK]

Extended CoAP Setup

Stateful forwarding

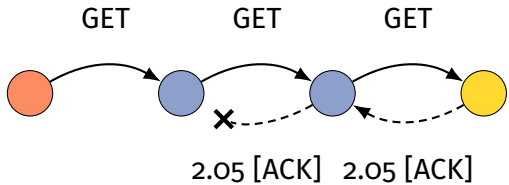
- Proxy on each forwarding node



Extended CoAP Setup

Stateful forwarding

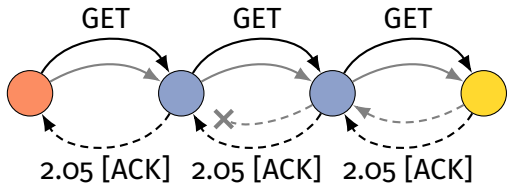
- Proxy on each forwarding node
- Hop-wise message timeout



Extended CoAP Setup

Stateful forwarding

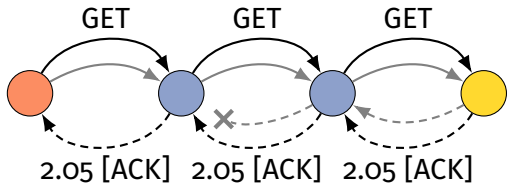
- Proxy on each forwarding node
- Hop-wise message timeout
- Retransmissions on each forwarder



Extended CoAP Setup

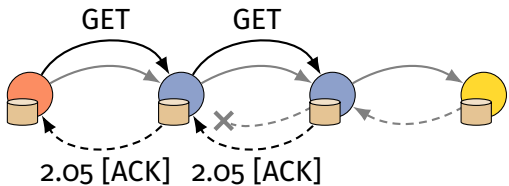
Stateful forwarding

- Proxy on each forwarding node
- Hop-wise message timeout
- Retransmissions on each forwarder



Content caching

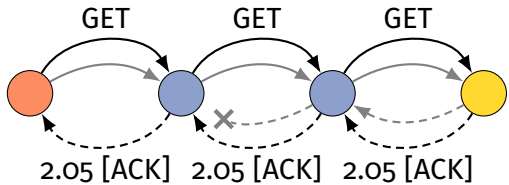
- Shortened request path `[ICN'18]`
- Reduced retransmissions



Extended CoAP Setup

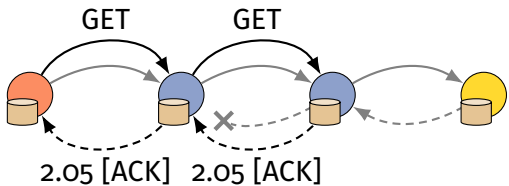
Stateful forwarding

- Proxy on each forwarding node
- Hop-wise message timeout
- Retransmissions on each forwarder



Content caching

- Shortened request path `[ICN'18]`
- Reduced retransmissions



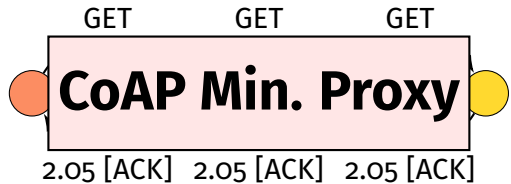
Decoupling of data from location

- Link-local IP addressing
- Forwarding via resource name

Extended CoAP Setup

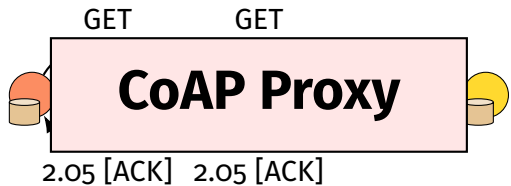
Stateful forwarding

- Proxy on each forwarding node
- Hop-wise message timeout
- Retransmissions on each forwarder



Content caching

- Shortened request path `[ICN'18]`
- Reduced retransmissions



Decoupling of data from location

- Link-local IP addressing
- Forwarding via resource name

Content Object Security with CoAP

OSCORE

- Object Security for Constrained RESTful Environments
- Proposed standard (RFC8613) since July 2019
- Builds on COSE: CBOR Object Signing and Encryption (RFC8152)

Content Object Security with CoAP

OSCORE

- Object Security for Constrained RESTful Environments
- Proposed standard (RFC8613) since July 2019
- Builds on COSE: CBOR Object Signing and Encryption (RFC8152)

Security Properties

- Authenticated Encryption with Associated Data (AEAD)
- Confidentiality, Integrity, Request—Response binding, Non-replayability

Content Object Security with CoAP

OSCORE

- Object Security for Constrained RESTful Environments
- Proposed standard (RFC8613) since July 2019
- Builds on COSE: CBOR Object Signing and Encryption (RFC8152)

Security Properties

- Authenticated Encryption with Associated Data (AEAD)
- Confidentiality, Integrity, Request—Response binding, Non-replayability

Cacheability

- Strong message binding prevents cache hits for subsequent requests

→ We use retransmission caches to recover messages of same transaction

Information-centric Properties with CoAP Proxies

- ☑ **Stateful forwarding**
- ☑ **Content caching**
- ☑ **Decoupling of data from location**
- ☑ **Content object security**

Protocol Performance Evaluation

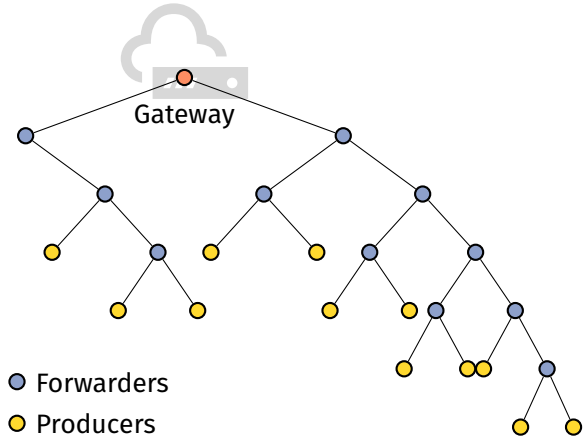
Testbed Setup

Hardware M3 node in IoT Lab testbed, IEEE 802.15.4

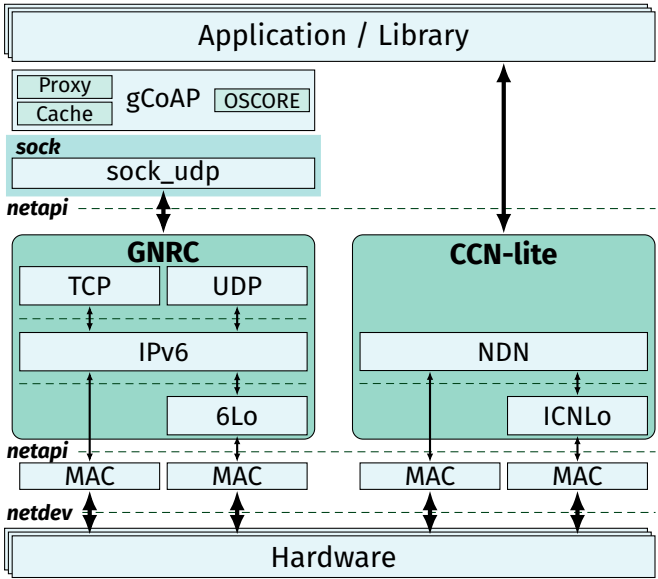
Software RIOT

Topology 12 producers, 11 forwarders

Scenario Gateway requests 2-byte temperature every $\approx$ 1 s



RIOT Network Stack



CoAP with Proxy

- Stateful proxying and caching in gCoAP

CoAP with OSCORE

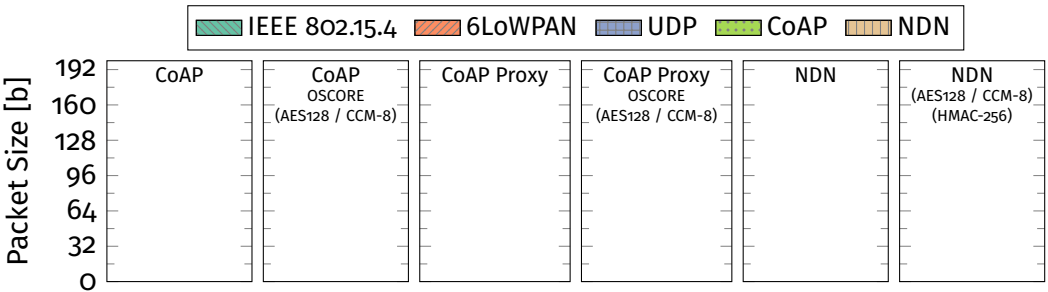
- gCoAP integrates libOSCORE package

NDN with CCN-lite

- CCN-lite integrates into RIOT networking

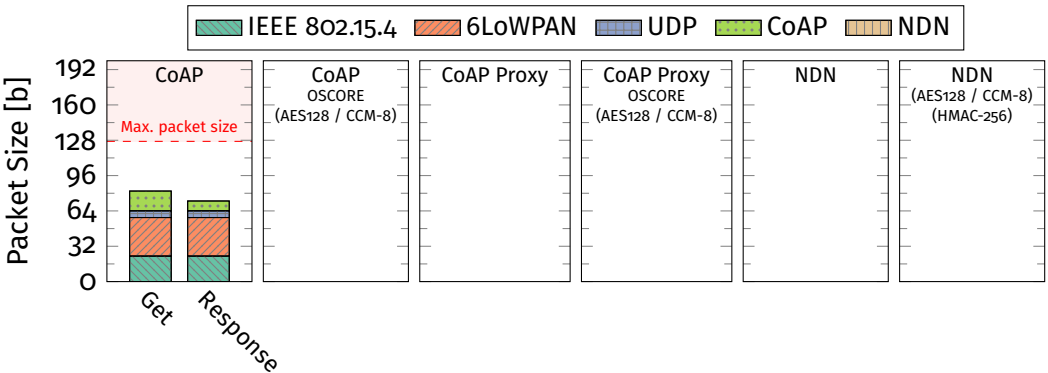
Packet Structure Dissection

- Maximum frame size for IEEE 802.15.4 is 127 bytes



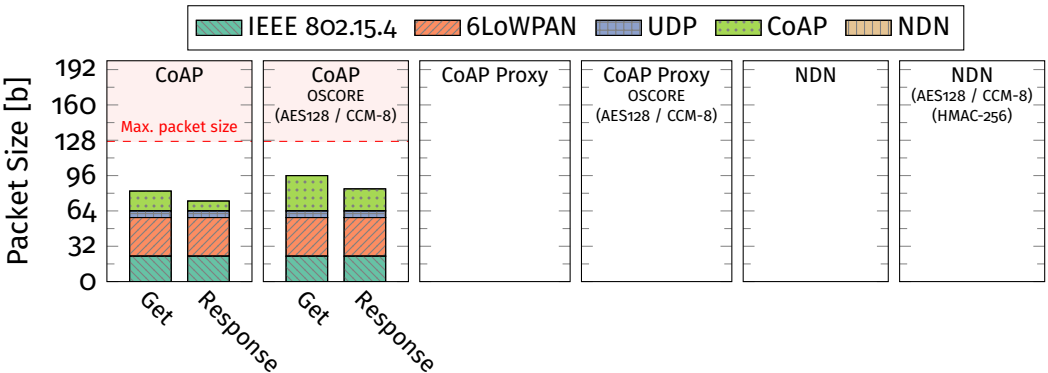
Packet Structure Dissection

- Maximum frame size for IEEE 802.15.4 is 127 bytes



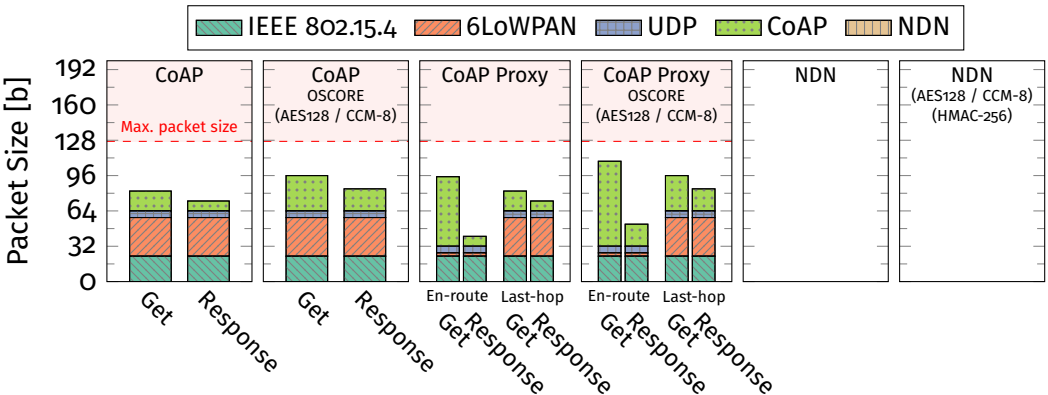
Packet Structure Dissection

- Maximum frame size for IEEE 802.15.4 is 127 bytes



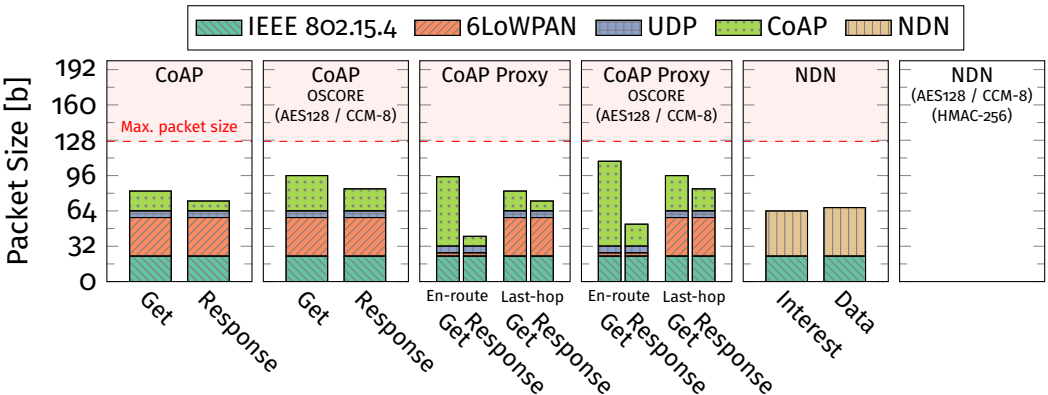
Packet Structure Dissection

- Maximum frame size for IEEE 802.15.4 is 127 bytes



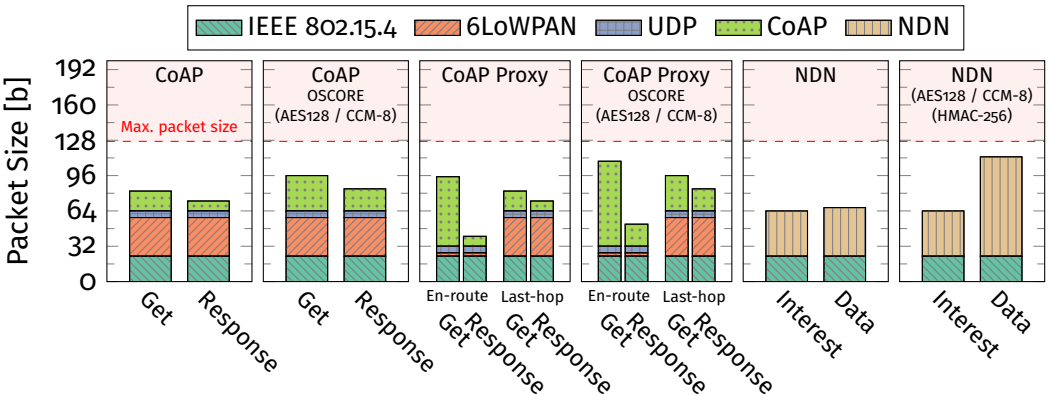
Packet Structure Dissection

- Maximum frame size for IEEE 802.15.4 is 127 bytes



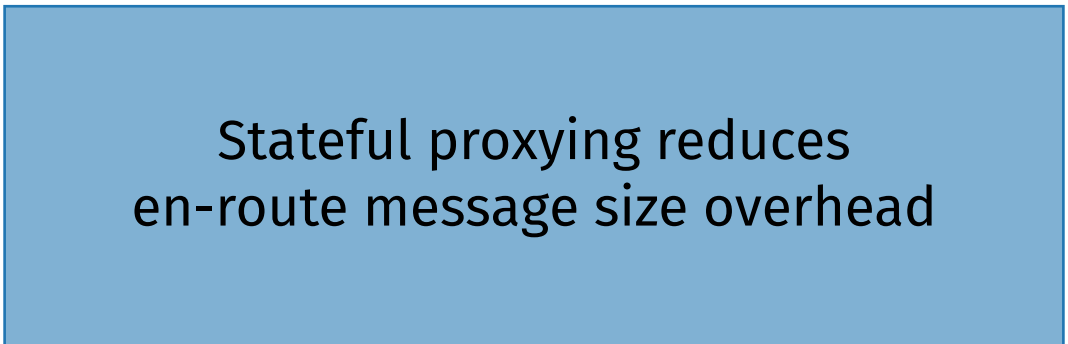
Packet Structure Dissection

- Maximum frame size for IEEE 802.15.4 is 127 bytes

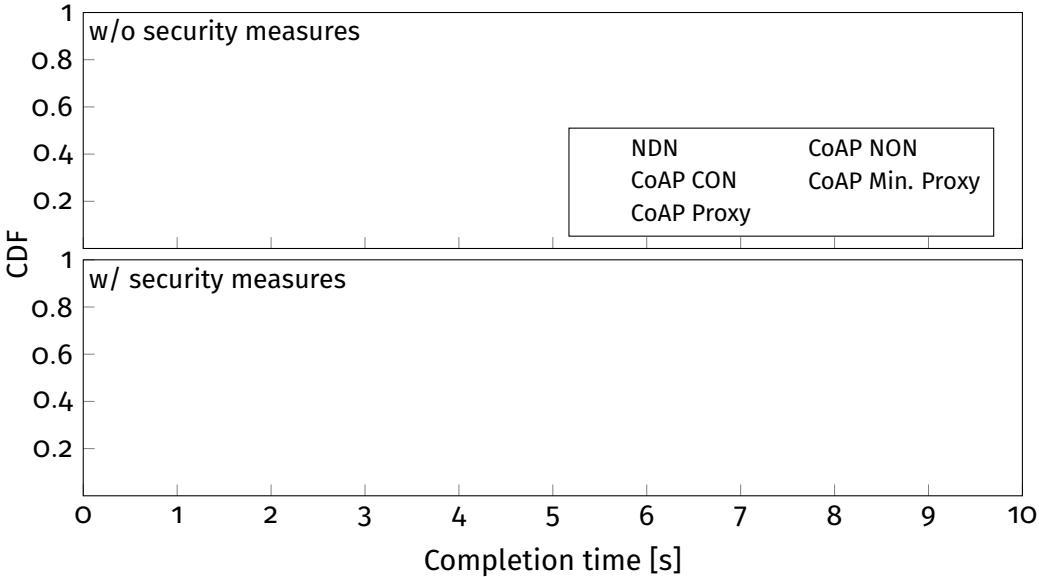


Packet Structure Dissection

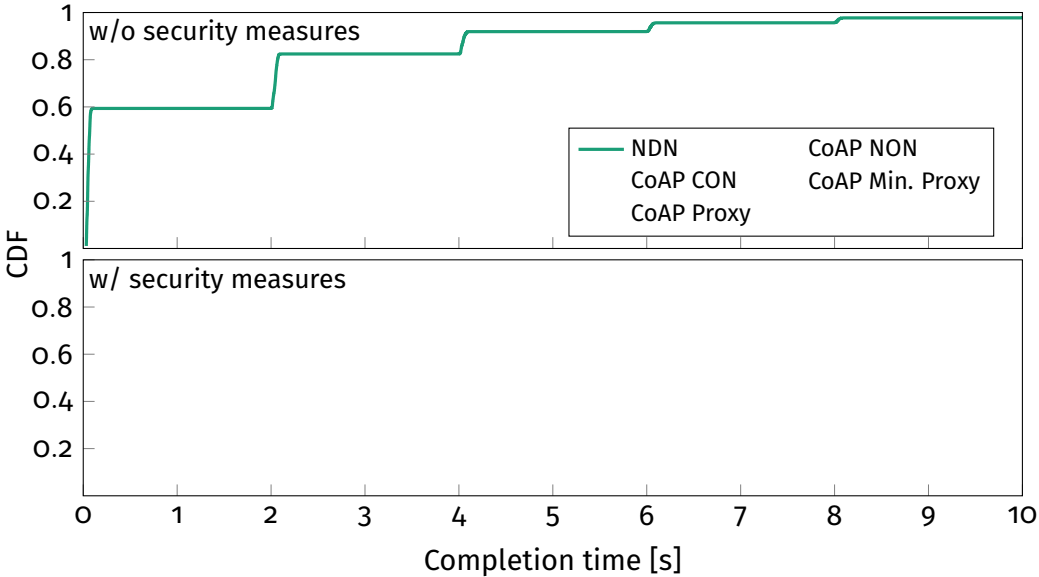
- Maximum frame size for IEEE 802.15.4 is 127 bytes



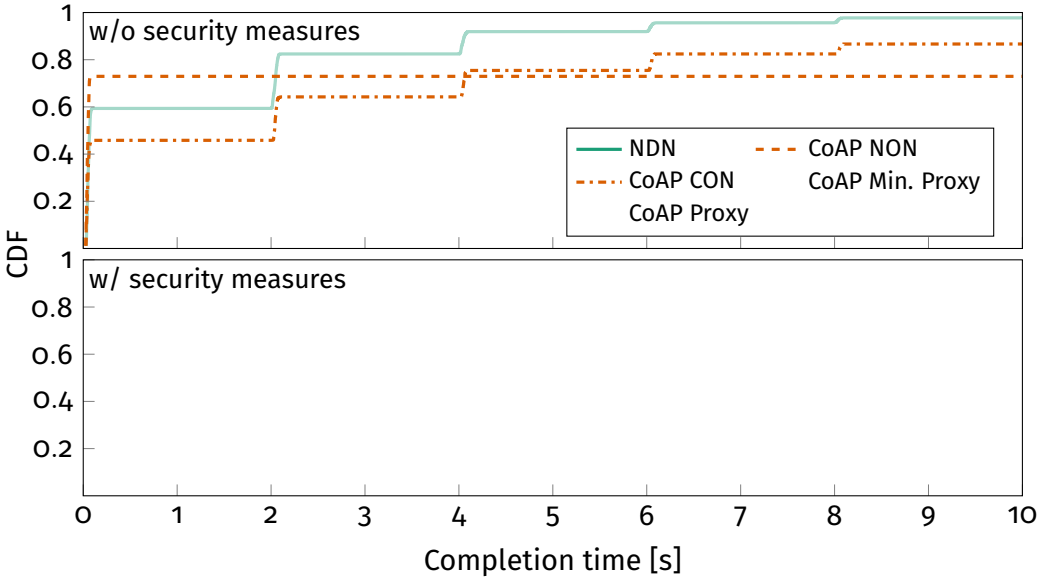
Time to Content Arrival



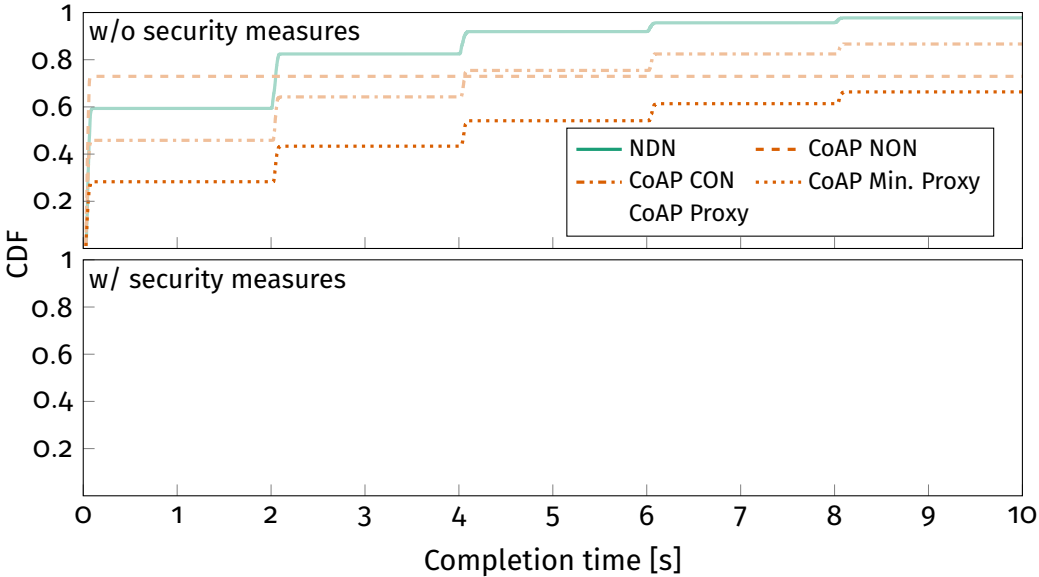
Time to Content Arrival



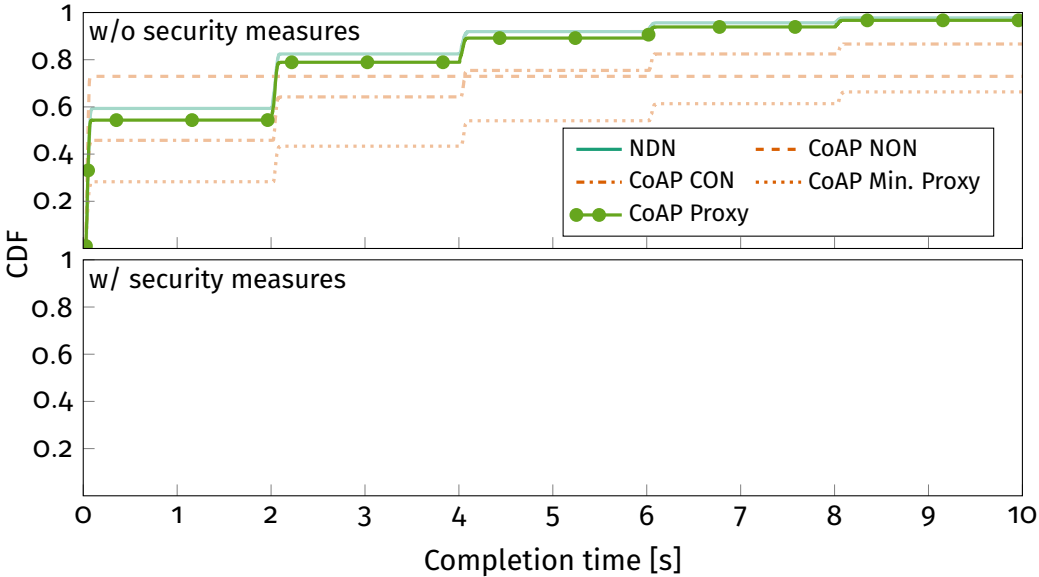
Time to Content Arrival



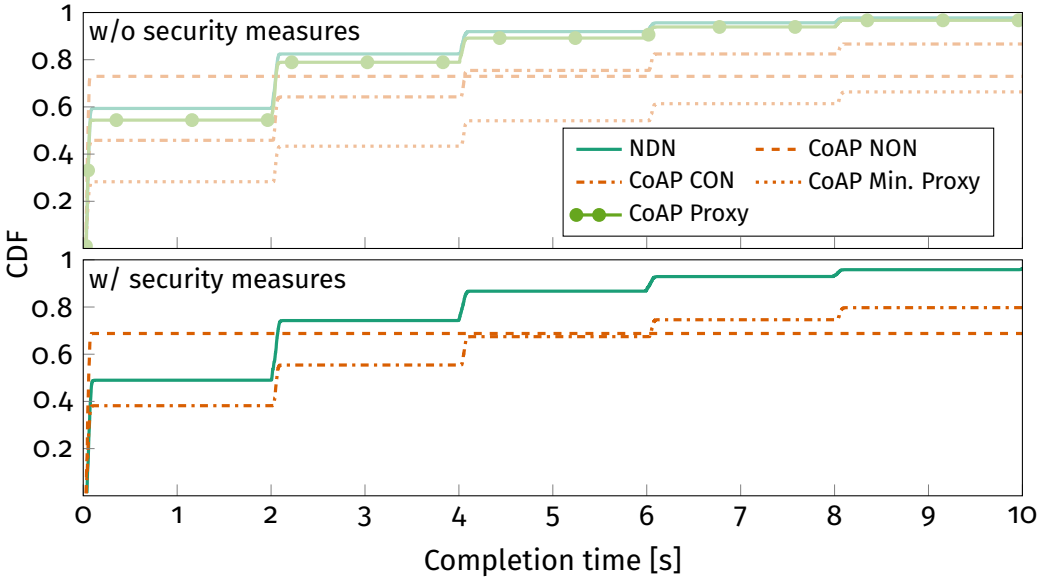
Time to Content Arrival



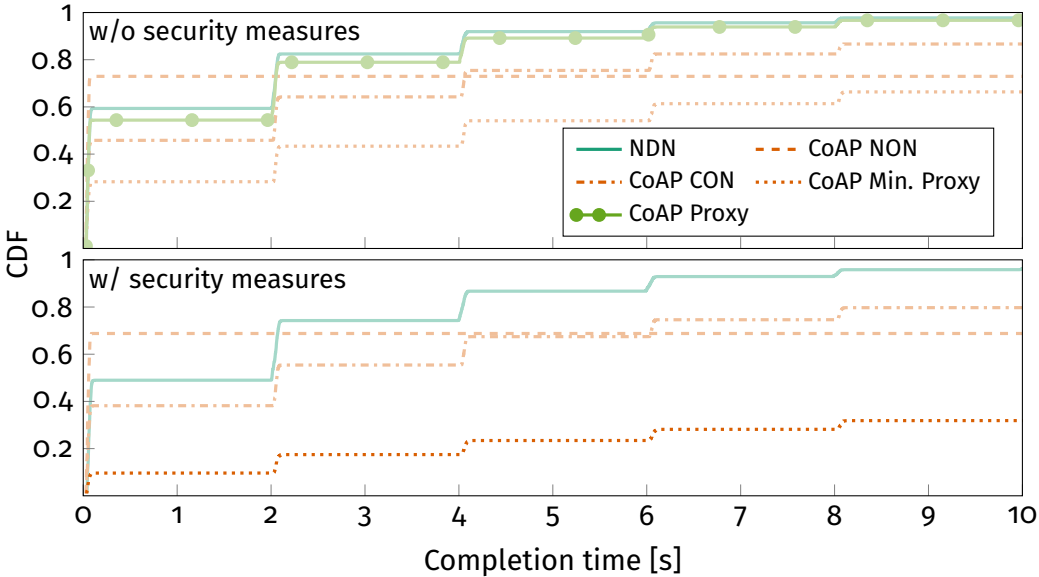
Time to Content Arrival



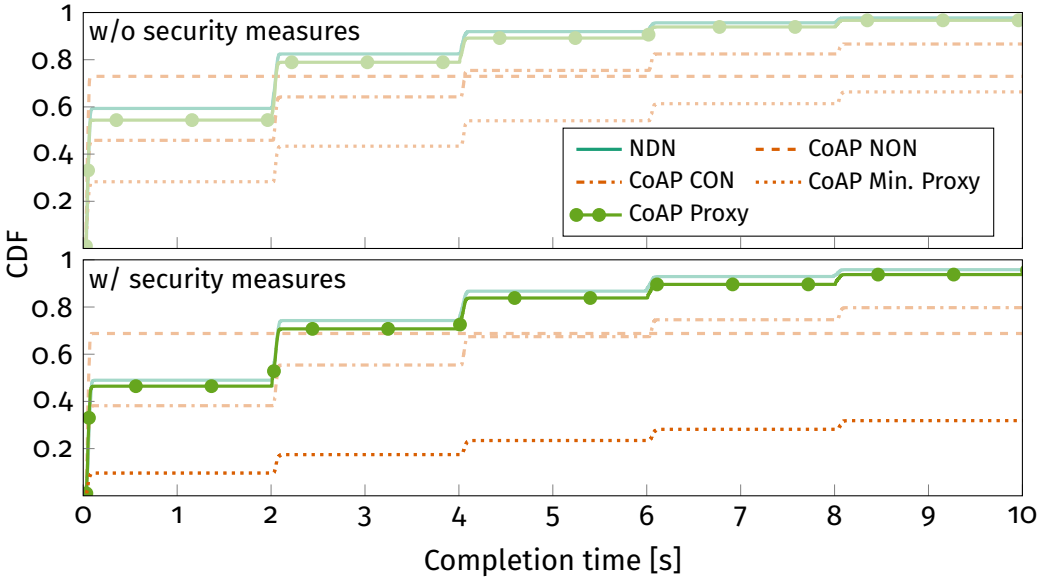
Time to Content Arrival



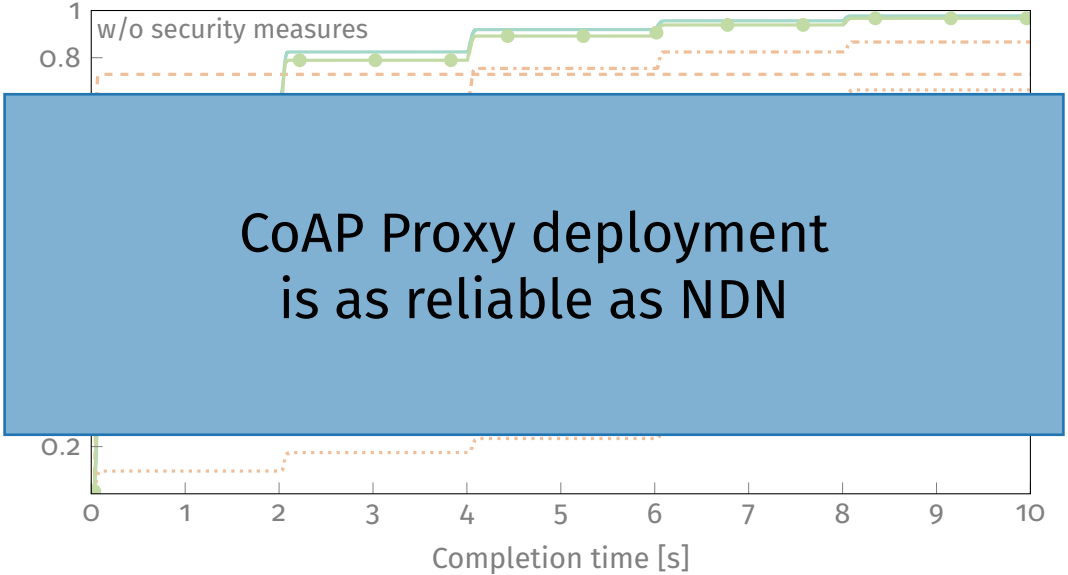
Time to Content Arrival



Time to Content Arrival

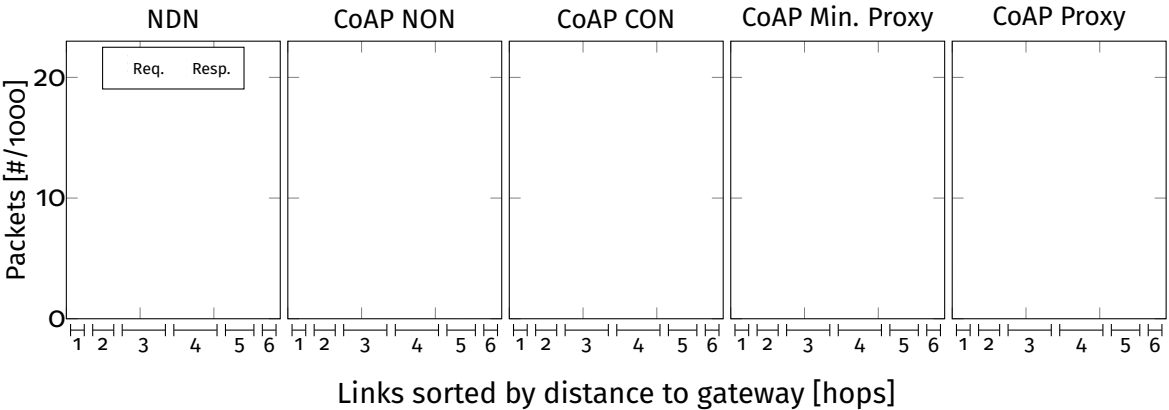


Time to Content Arrival

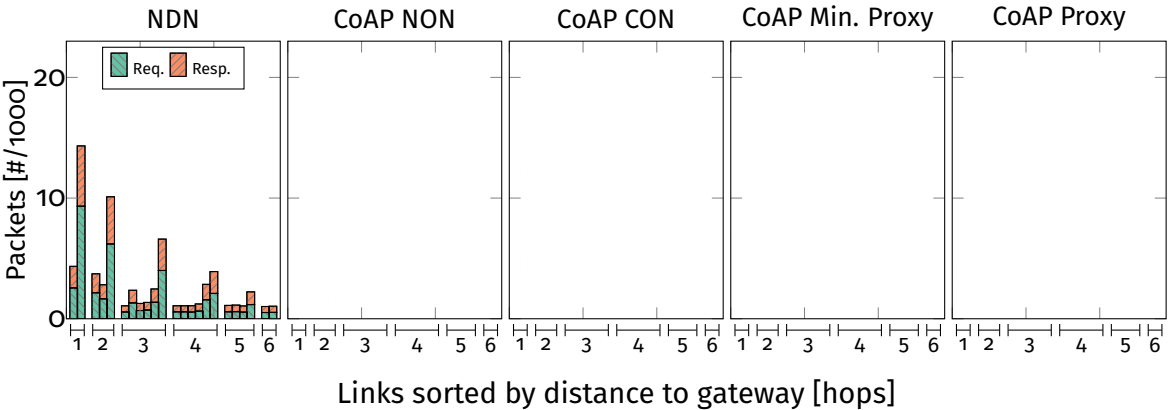


CoAP Proxy deployment
is as reliable as NDN

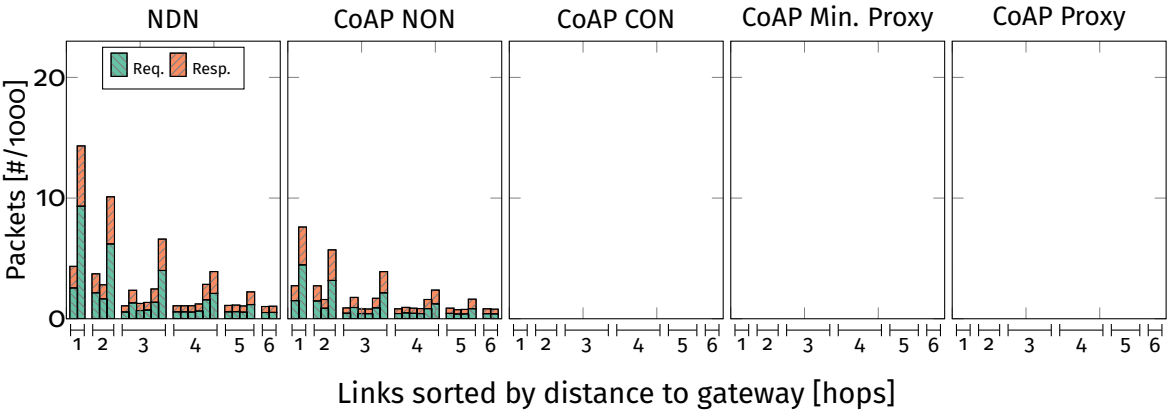
Link Stress for the Secured Protocol Variants



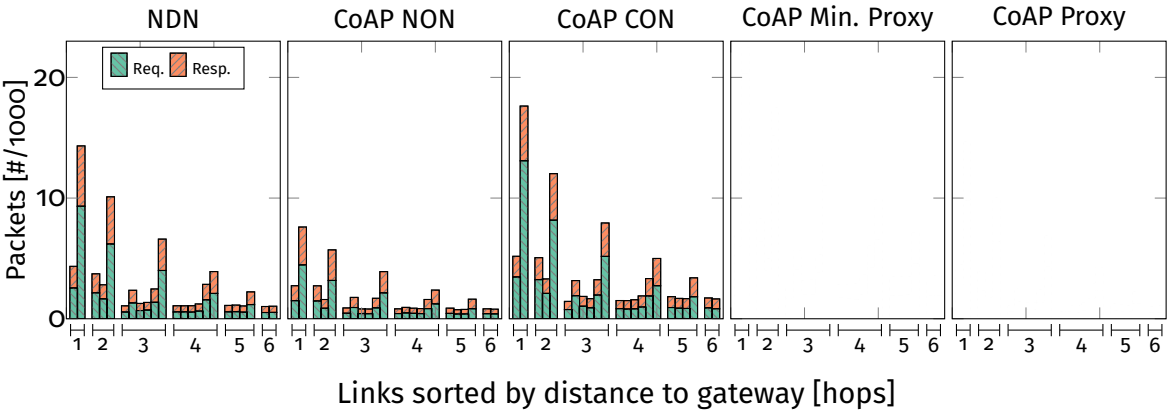
Link Stress for the Secured Protocol Variants



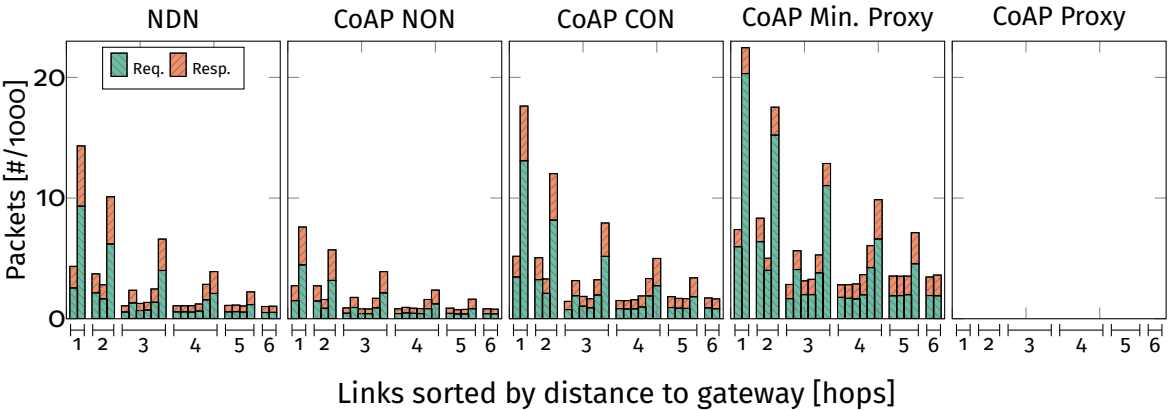
Link Stress for the Secured Protocol Variants



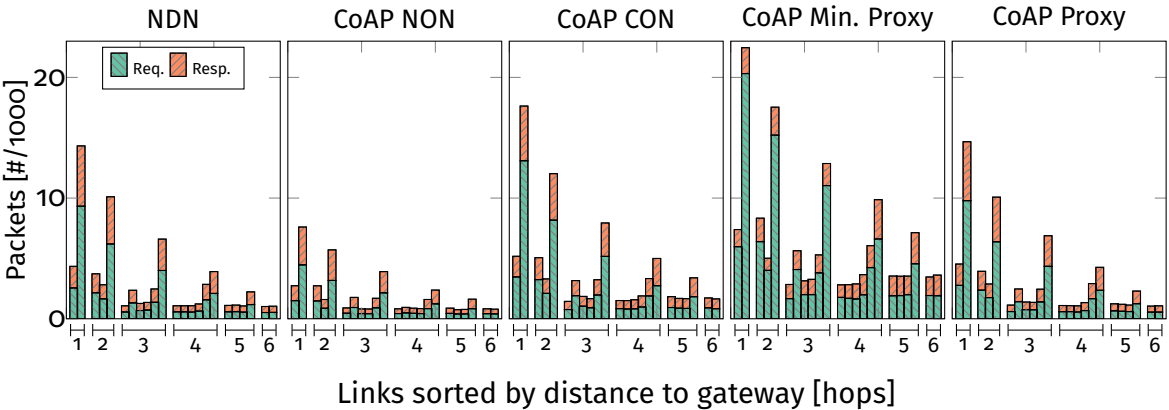
Link Stress for the Secured Protocol Variants



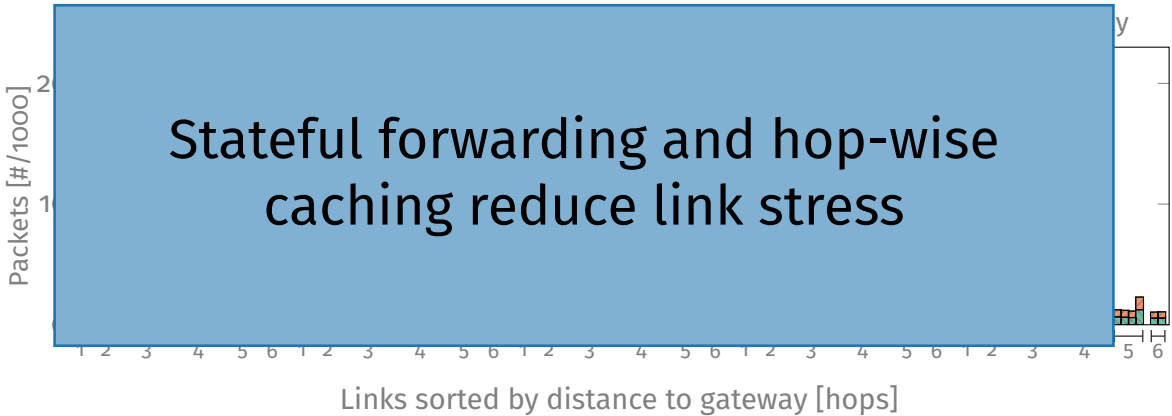
Link Stress for the Secured Protocol Variants



Link Stress for the Secured Protocol Variants

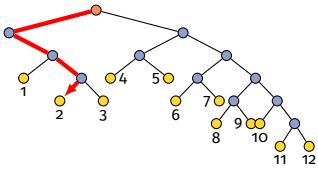


Link Stress for the Secured Protocol Variants

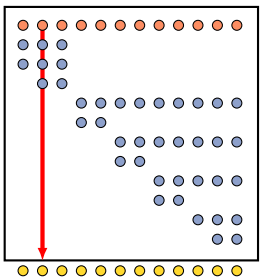


Stateful forwarding and hop-wise caching reduce link stress

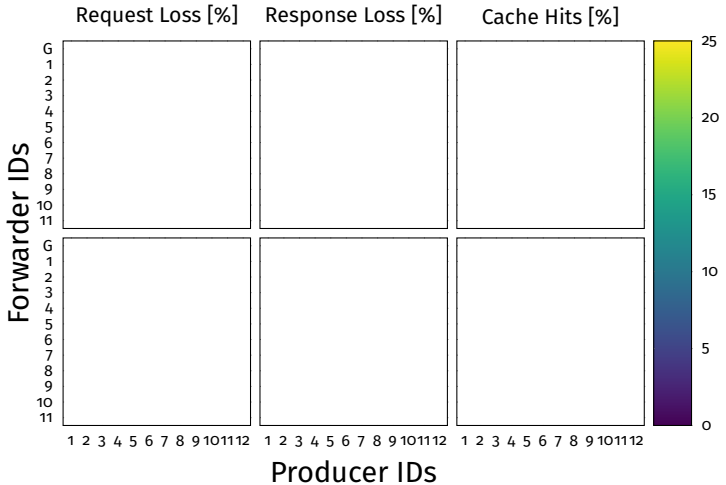
Cache Utilization for the Secured Protocol Variants



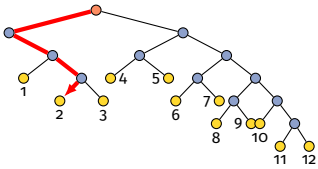
Gateway



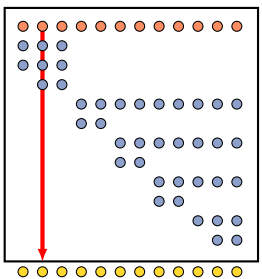
Producers



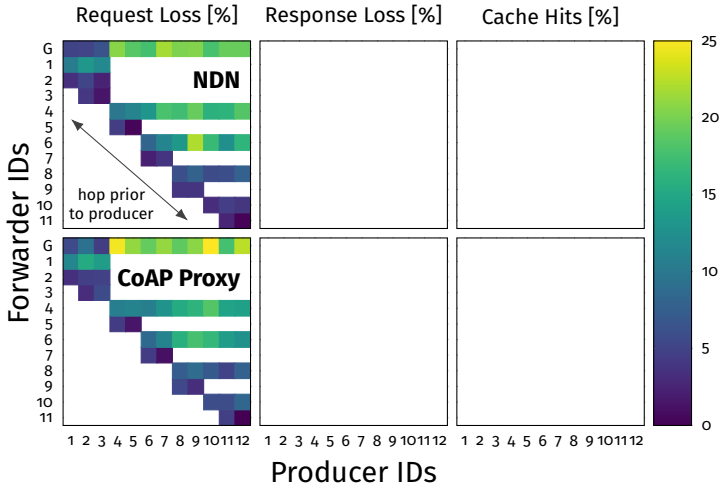
Cache Utilization for the Secured Protocol Variants



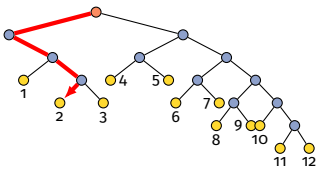
Gateway

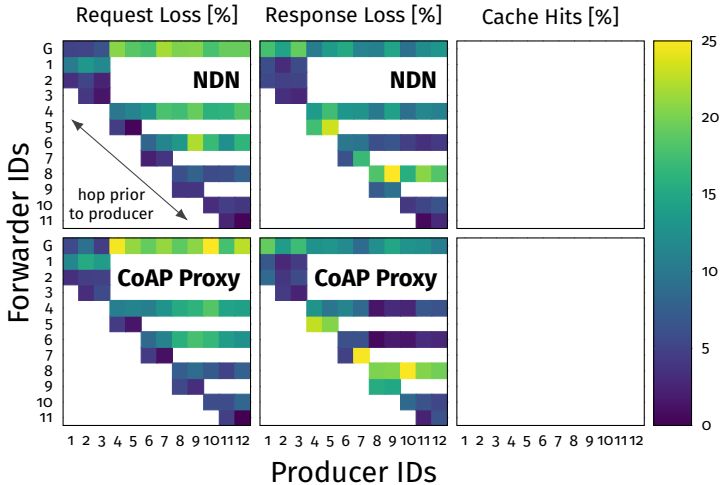


Producers

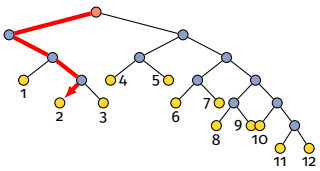


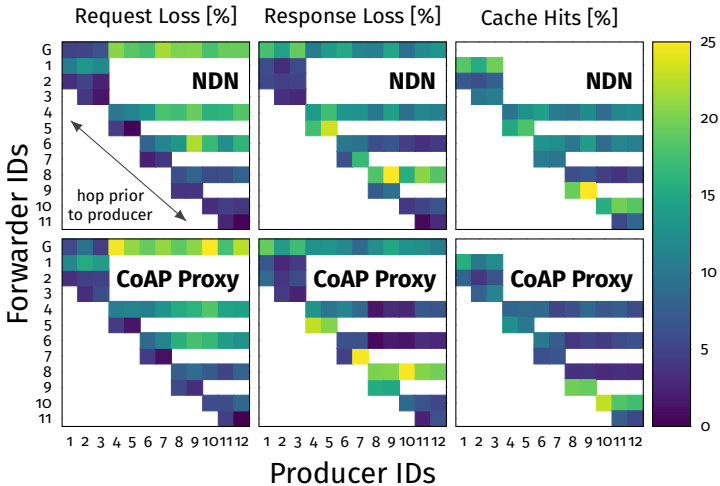
Cache Utilization for the Secured Protocol Variants



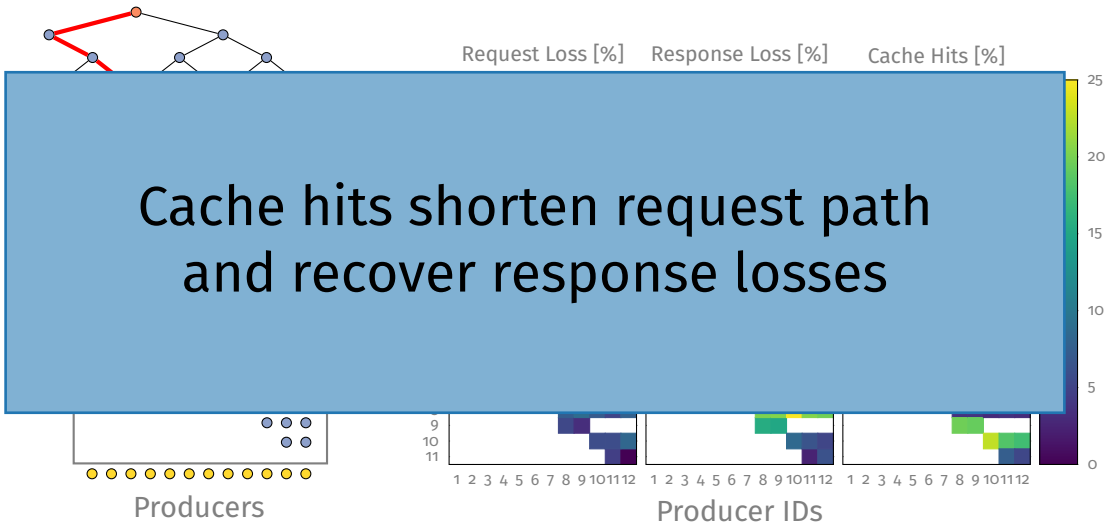


Cache Utilization for the Secured Protocol Variants





Cache Utilization for the Secured Protocol Variants



Conclusion & Outlook

Takeaways

- Information-centric WoT can be built with CoAP standard features
- Stateful forwarding and hop-wise caching improves reliability for CoAP
- Deployment chance for ICN principles in existing IoT infrastructure

Next Steps

- Explore ICN forwarding and caching policies in the context of CoAP
- Investigate multicast properties for an information-centric Web of Things

Thank You!

We support reproducible research.

https://github.com/inetrg/ACM-ICN-2020-COAP