Reserved for picture in picture

NDN-MPS: Supporting Multiparty Authentication over NDN

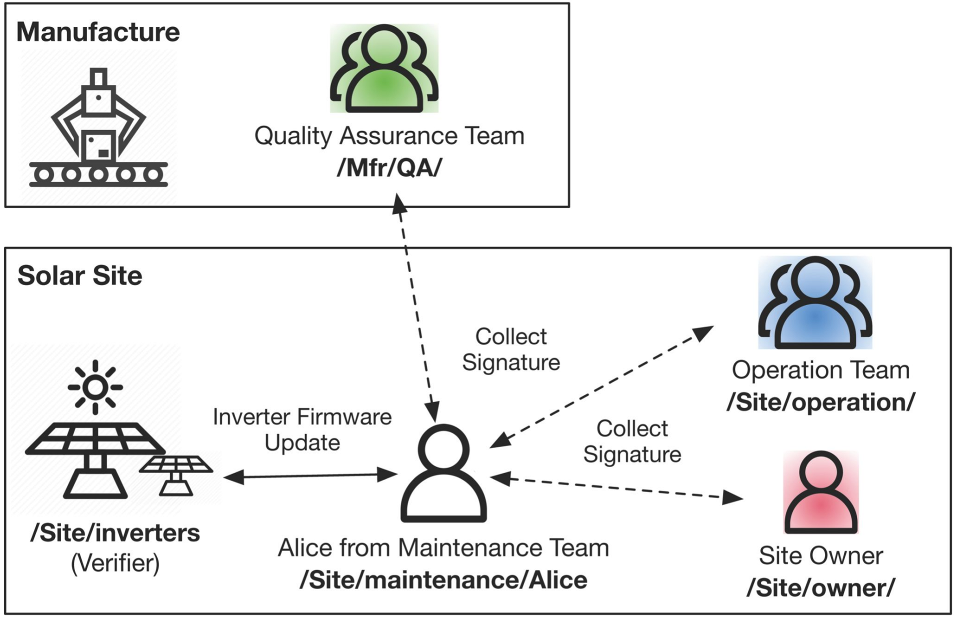
Zhiyi Zhang(**presenter**), Siqi Liu, Randy King, Lixia Zhang
UCLA, Operant Networks

ACM ICN 2021

Reserved for picture in picture

Multiparty Authentication

- Real world business decision involves multiple parties
- Real problem we met
  - Solar energy network system
  - Inverter software update command requires approvals from multiple parties
    - Site owner
    - Site operation team
    - Manufacture QA team

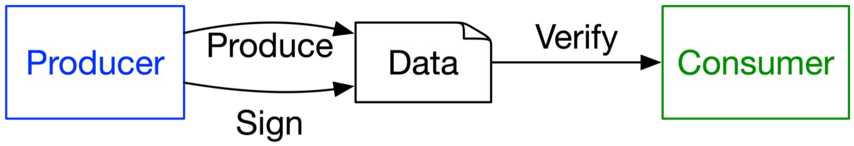


Switch from Prod-Con Trust Model to Multiparty Trust

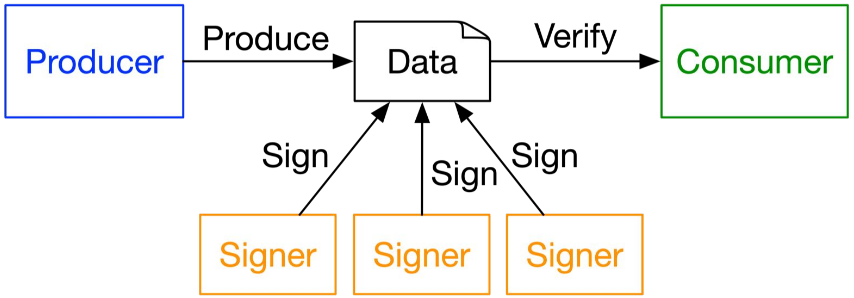
Reserved for picture in picture

- Third party signers who are not the content producer
- Verification against a list of signers
- Coordination among the signers

Producer-Consumer Model



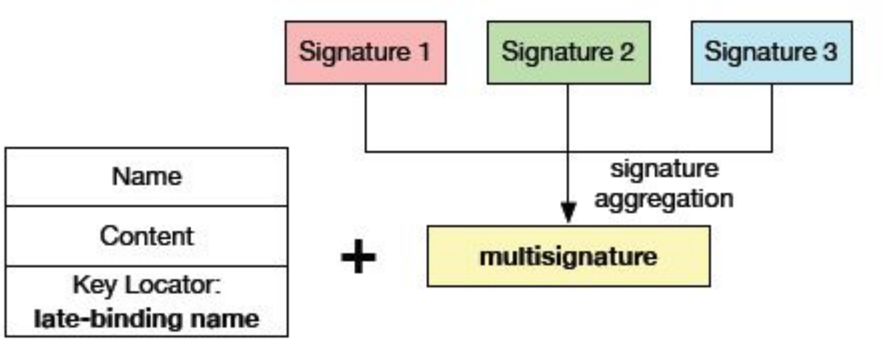
Multiparty Signature Model



Reserved for picture in picture

Crypto: Existing Schemes vs Multisignature

- Conventional solution: obtain a list of signatures from individual signers
  - Large packet/signature size O(n)
  - Long verification time O(n)
- Multisignature: multiple signatures can be aggregated into one
  - Single signature O(1)
  - Single verification operation O(1)



What is missing?

This can be addressed by existing trust schema support

Reserved for picture in picture

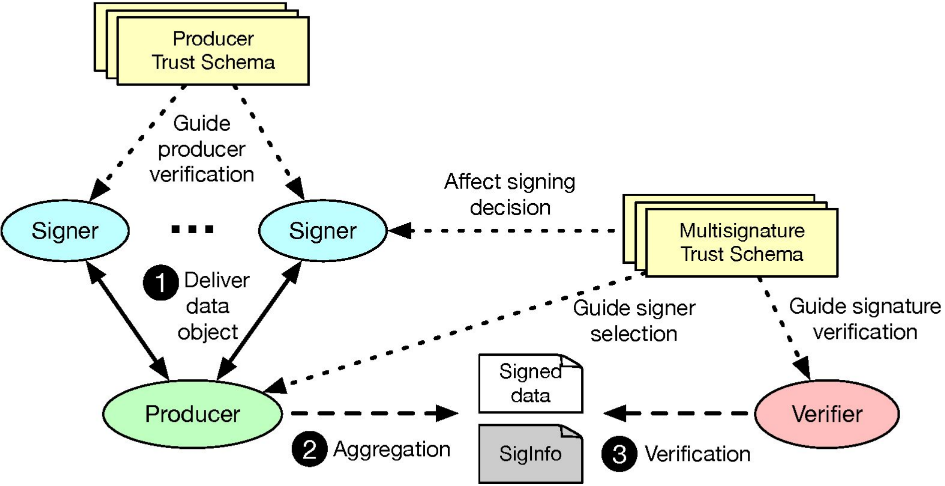
- Trust schema for each signer to verify the producer and vice versa
- Multiparty trust schema
  - To defines signing and verification rules that involve multiple signers (and trust anchors)
- Multisignature encoding
  - To encode signature and its multi-party specific signature information
- The coordination mechanism
  - To collect and aggregate signatures from individual signers

The rest are new issues

NDN-MPS: Toolkit for Multisignature based Multiparty Authentication

Reserved for picture in picture

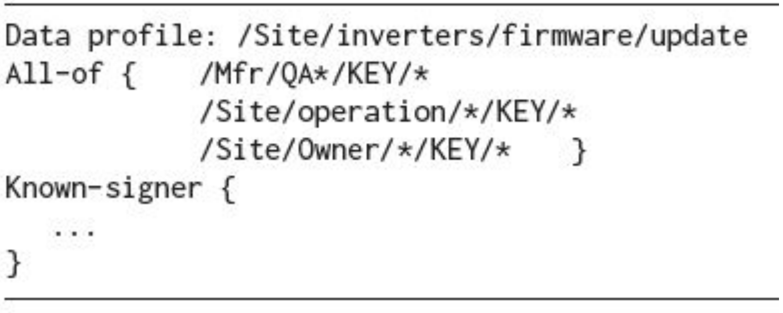
- Multisignature trust schema support
- An NDN-compatible multisignature encoding mechanism
- Two coordination mechanisms for multisignature generation
  - NDN Remote Procedure Call (RPC) bas coordination
  - NDN sync-based coordination



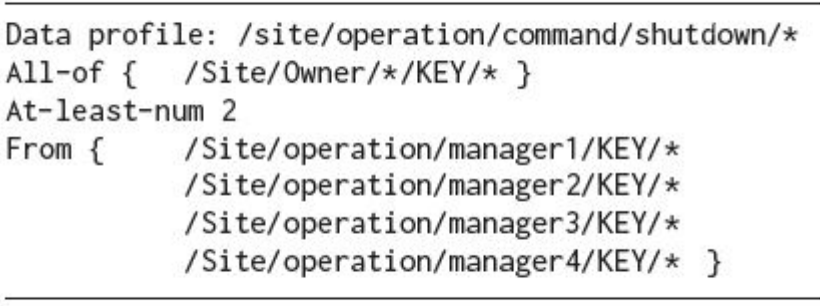
Reserved for picture in picture

Multisignature Trust Schema

- A list of required signer identities
- These signer’s certificate chains to one or more trust anchors
- Threshold policy: valid when k out of n signers sign the object
  - NDN-MPS support this with a system approach rather than using additional cryptographic primitives for simplicity of key setup and management



```
Data profile: /Site/inverters/firmware/update
All-of {    /Mfr/QA*/KEY/*
            /Site/operation/*/KEY/*
            /Site/Owner/*/KEY/*    }
Known-signer {
   ...
}
```

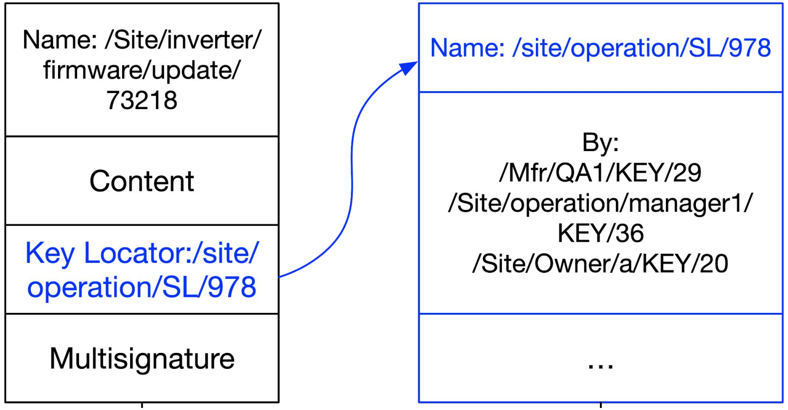


```
Data profile: /site/operation/command/shutdown/*
All-of {   /Site/Owner/*/KEY/* }
At-least-num 2
From {     /Site/operation/manager1/KEY/*
           /Site/operation/manager2/KEY/*
           /Site/operation/manager3/KEY/*
           /Site/operation/manager4/KEY/*  }
```

Reserved for picture in picture

Multisignature Encoding

- New signature type
- New key locator to keep information of multiple signers
  - Must be consistent among multiple signers
  - Must tolerate changes of signer list during the coordination:
    - One required signer /site/owner/*/KEY/*
    - The producer decides to go with /site/owner/alice first
    - When Alice is down, change it to /site/owner/bob


Name: /Site/inverter/firmware/update/73218
Content
Key Locator:/site/operation/SL/978
Multisignature
Name: /site/operation/SL/978
By:
/Mfr/QA1/KEY/29
/Site/operation/manager1/KEY/36
/Site/Owner/a/KEY/20
...

- Solve the problem with another layer of indirection: placeholder key locator

Multiparty Signing Coordination: What is needed?

Reserved for picture in picture

- First step: the producer publishes the unsigned data object to signers
- Second step: collect signature pieces from signers

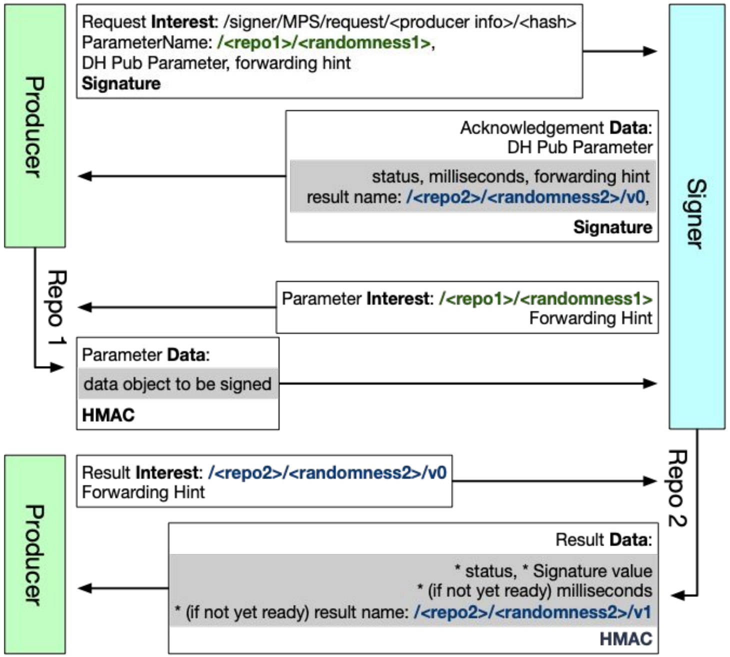
Security objectives:

- Authenticity
- Confidentiality: just like in prod-con trust model: content is not available until it is packetized

Reserved for picture in picture

Multiparty Signing Coordination: RPC

- RPC based: NDN-MPS RPC
  - Diffie-hellman key exchange in the first round trip to ensure confidentiality
  - Asynchronous: informed estimated processing time
  - Repo-friendly: both parameter and result can be published to repos



Request **Interest**: /signer/MPS/request/<producer info>/<hash>
ParameterName: /<repo1>/<randomness1>,
DH Pub Parameter, forwarding hint
Signature

Acknowledgement **Data**:
DH Pub Parameter
status, milliseconds, forwarding hint
result name: /<repo2>/<randomness2>/v0,
Signature

Parameter **Interest**: /<repo1>/<randomness1>
Forwarding Hint

Parameter **Data**:
data object to be signed
HMAC

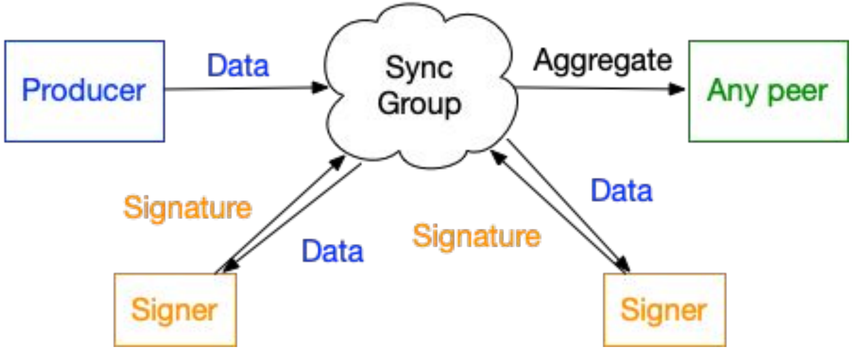
Result **Interest**: /<repo2>/<randomness2>/v0
Forwarding Hint

Result **Data**:
* status, * Signature value
* (if not yet ready) milliseconds
* (if not yet ready) result name: /<repo2>/<randomness2>/v1
HMAC

Multiparty Signing Coordination: Sync

Reserved for picture in picture

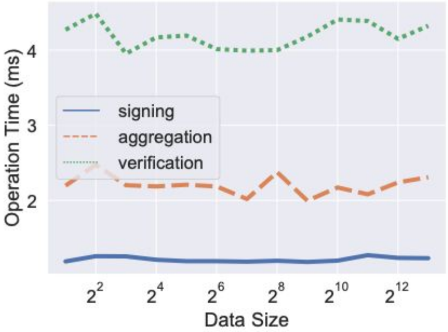
- NDN Sync based: E.g., SVS
  - Require group-level encryption
  - Require group identity management
- Two approaches work for different applications
  - Already use sync?
  - Want simple setup?

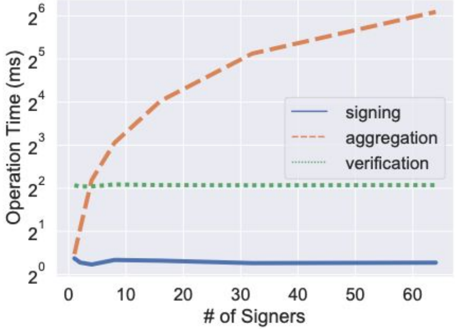


Reserved for picture in picture

Implementation and Evaluation

- A C++ library with usable APIs (works over ndn-cxx library)
  - BLS signature: no interactive key setup
  - Also integrated into ndncert as a multiparty-approved identity verification challenge
- Benchmark with different size of data and signer set
  - Confirmed O(1) signing and verification time
  - Confirmed O(1) signature size: 128 bit security requires 96 bytes signature regardless of # of signers





Thank you!

Q&A