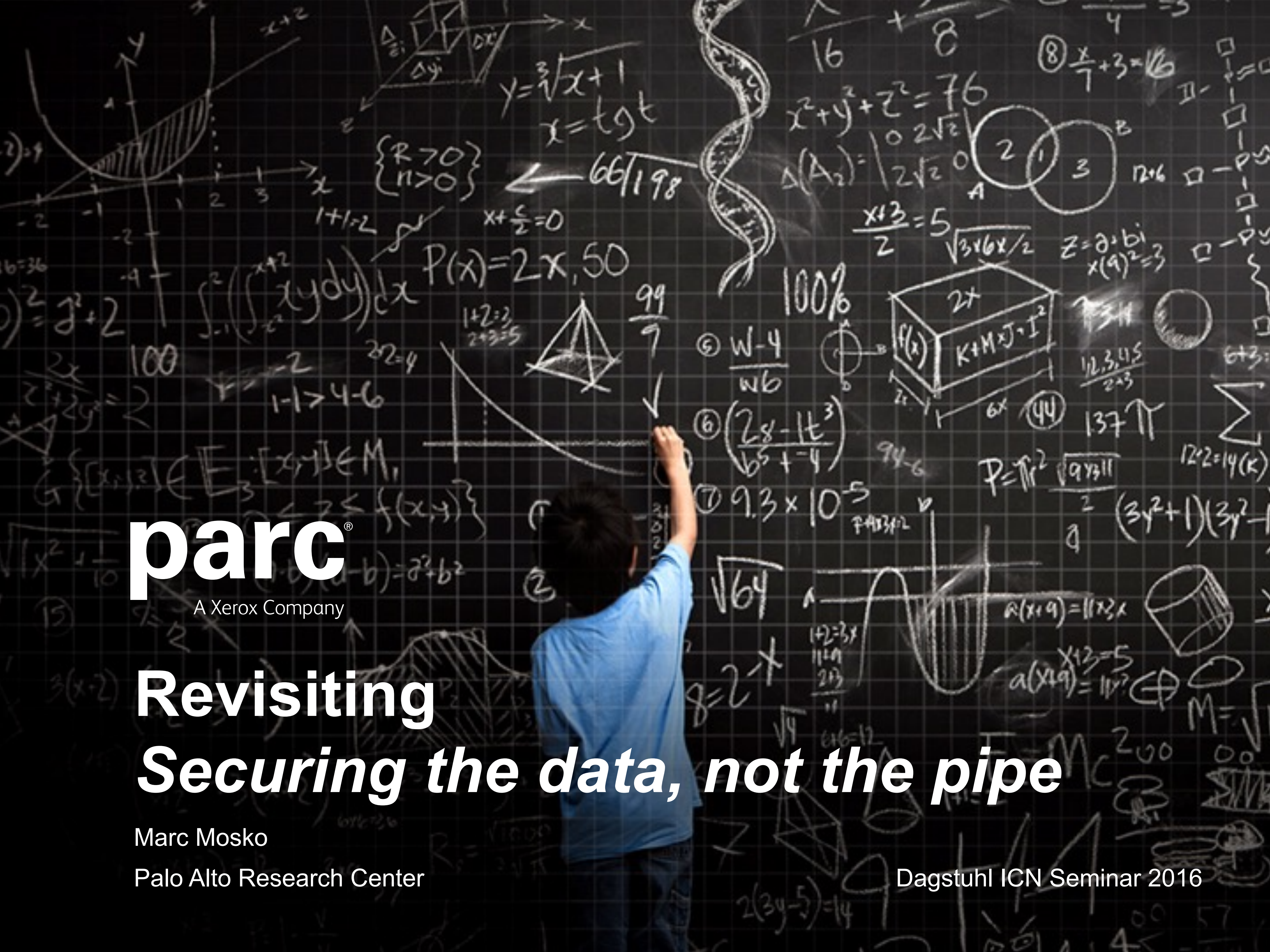


parc®

A Xerox Company

Revisiting

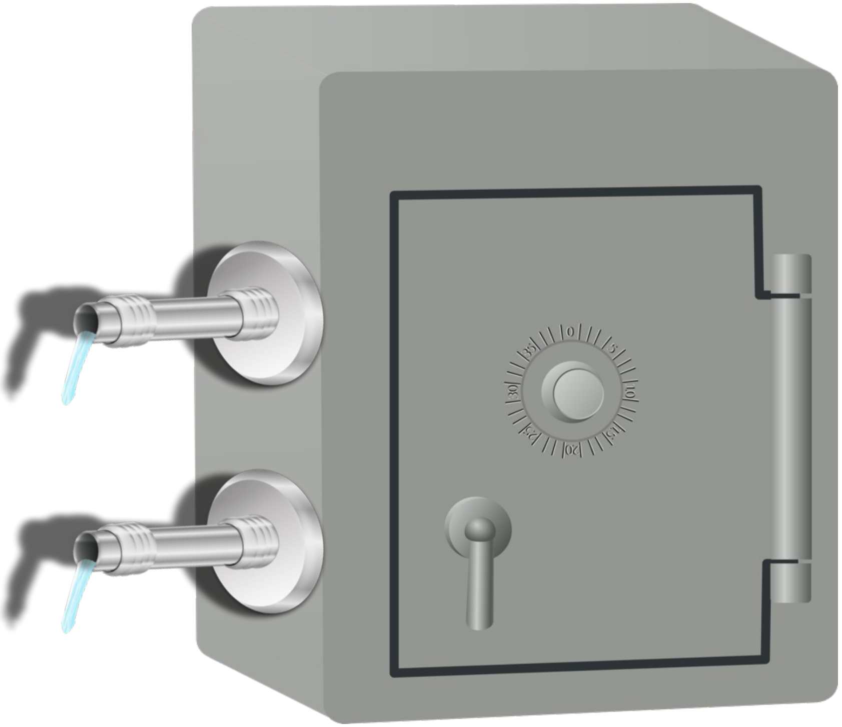
Securing the data, not the pipe

Marc Mosko

Palo Alto Research Center

Dagstuhl ICN Seminar 2016

Secure the pipe



Usually interpreted as both:

- Knowing the endpoints (at least server side)
- Confidentiality

Criticized:

- Trust is the endpoint
- Trust anchor is browser cert list

Secure the data

Usually interpreted as both:

- Knowing the producer
- Confidentiality

Criticized:

- Really? Take what you want?
- Often relies on long-term keys for object security (no forward secrecy)
- Plaintext signature, no privacy?

Some properties

- Authentication
  - Is it publically verifiable?
    - E.g. sign with public key after encrypt
  - Privately verifiable?
    - E.g. AEAD or encrypt after sign
- Privacy
  - What part of the packet is protected?
    - The payload?
    - The metadata?
    - The name (or part of?)
  - How much info might be leaked?
    - E.g. doing AES encryption of name components might still leak info due to short sizes and guessing

Some more properties

- Consumer set
  - Is it know at the time of encryption?
  - Is the publisher of the data the same as the key issuer?
    - E.g. NDN-NBAC has an “owner” separate from a “publisher” (e.g. sensor). The owner sanctions publishers for name spaces and only the owner deals out keys to consumers.
  - Is it a true group encryption/decryption algorithm, or is it some shared key with consumer key wrapping / encapsulation?

Forward Secrecy

(Perfect) Forward Secrecy: session keys established (and deleted) before a compromise of long-term keys cannot be recovered

Weak PFS: Session keys established without intervention (or eavesdropping) by attacker cannot be recovered after compromise of long-term keys

Why is ICN forward secrecy hard?

Publish-and-forget

- publish once, write to network, no re-encryption

Disconnected operation

- cannot execute on-line algorithm (e.g. authenticated DH, 3-message implicit DH, etc.)

It implies a “session”

- We don’t want those (maybe?)
- Some models have session (e.g. NDN-ACE, CCNxKE)
- Some models have many consumers, with arbitrary post-publishing adds (e.g. CCNx ACS, NDN-NBAC)

What's been done [in CCN/NDN] (1)

- CCNx (pre 1.0) ACS [2010]
  - Generate a group key pair
  - Wrap the group secret key under member's public key
  - Wrap a data key under the group public key
- NDN Named Based Access Control [2015, 2016]
  - Owner certified KEK and KDK
  - For data, owner certifies a signing private key and wraps with producer's public key. Verification key is public.
  - Owner wraps KDK under member's public key
  - Publisher wraps data key in KEK
  - Consumers unwrap using KDK

What’s been done [in CCN/NDN] (2)

- NDN Access Control in Challenged Environments [2015]
  - Uses access server (AS)
  - Actuator creates a ‘seed’ key with AS using 1-round authenticated DH.
  - Client requests key from AS. Gets a key derived from seed mixed with its name/seqnum using 1-round authenticated DH further derived via KDF for irreversibility (client cannot further delegate).
  - Client sends signed request to actuator with seed keyid + its name/seqnum so actuator can verify HMAC using proper derived key.

Therefore…

- As far as I know
  - All existing CCN/NDN “group” access control schemes still come down to end-to-end message in the *key distribution* stage (either 1 or 2 messages).
  - These use simple key wrapping.
  - Do they even use *good* key wrapping (e.g. RSA-OAEP, RSA-KEM-DEM, RSASVE, PSEC-KEM, etc.)?
    - The tech reports don’t say, have not looked at code.
- Are there any possible key wrappings that provide one-way authenticated forward secrecy in the key distribution step?

SignCryption

cost (sign & encryption) ≪ cost (sign) + cost (encryption)

Zheng, Yuliang. "Digital signcryption or how to achieve cost (signature & encryption) ≪ cost (signature)+ cost (encryption)." In *Annual International Cryptology Conference*, pp. 165-179. Springer Berlin Heidelberg, 1997.

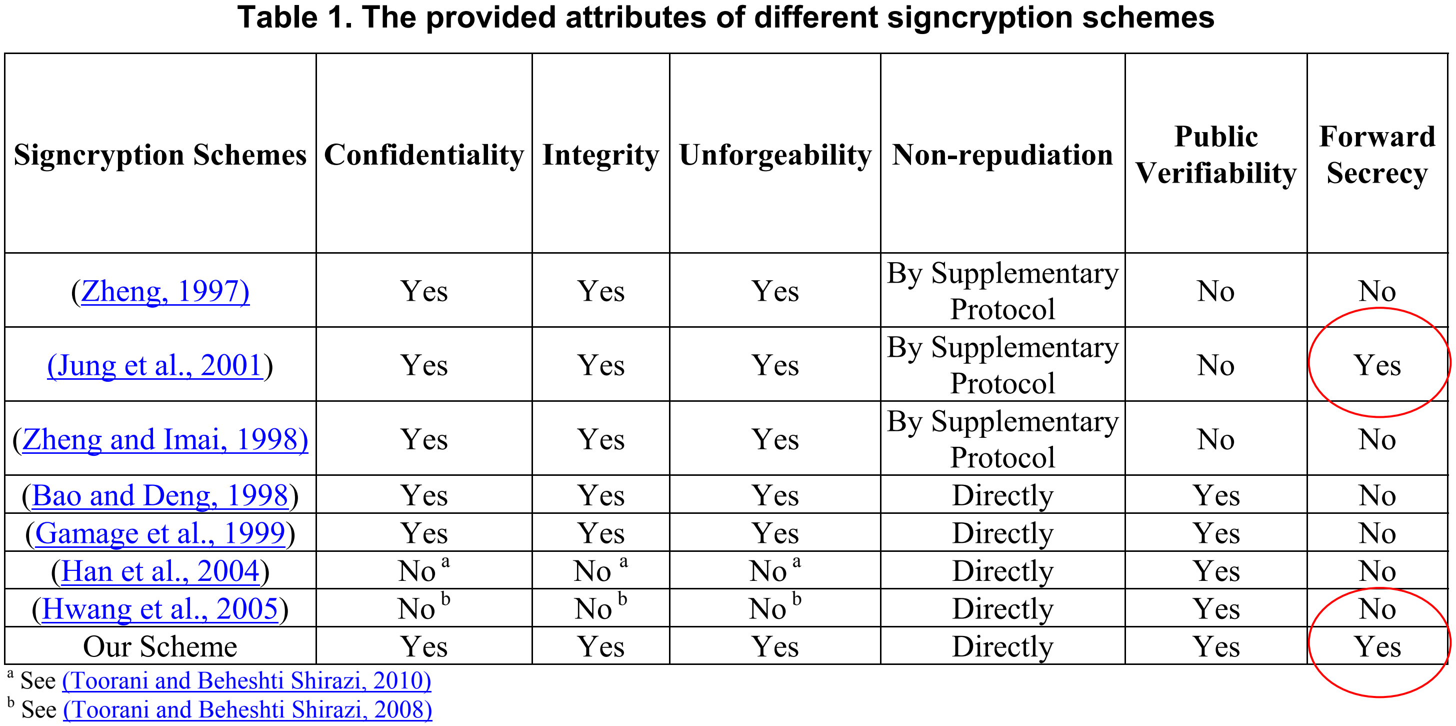


Table 1. The provided attributes of different signcryption schemes

Signcryption Schemes	Confidentiality	Integrity	Unforgeability	Non-repudiation	Public Verifiability	Forward Secrecy
(Zheng, 1997)	Yes	Yes	Yes	By Supplementary Protocol	No	No
(Jung et al., 2001)	Yes	Yes	Yes	By Supplementary Protocol	No	Yes
(Zheng and Imai, 1998)	Yes	Yes	Yes	By Supplementary Protocol	No	No
(Bao and Deng, 1998)	Yes	Yes	Yes	Directly	Yes	No
(Gamage et al., 1999)	Yes	Yes	Yes	Directly	Yes	No
(Han et al., 2004)	No [a]	No [a]	No [a]	Directly	Yes	No
(Hwang et al., 2005)	No [b]	No [b]	No [b]	Directly	Yes	No
Our Scheme	Yes	Yes	Yes	Directly	Yes	Yes

[a] See (Toorani and Beheshti Shirazi, 2010)

[b] See (Toorani and Beheshti Shirazi, 2008)

Toorani, Mohsen, and Ali A. Beheshti. "An elliptic curve-based signcryption scheme with forward secrecy." *arXiv preprint arXiv:1005.1856* (2010).

SignCryption

It simultaneously provides the attributes of message confidentiality, authentication, integrity, unforgeability, non-repudiation, public verifiability, and forward secrecy of message confidentiality.

… it has great advantages to be used for security establishments in store-and-forward applications and when dealing with resource constrained devices.

[underline added]

Toorani, Mohsen, and Ali A. Beheshti. "An elliptic curve-based signcryption scheme with forward secrecy." *arXiv preprint arXiv:1005.1856* (2010).

Ahirwal, Ramratan, Anjali Jain, and Y. K. Jain. "Signcryption scheme that utilizes elliptic curve for both encryption and signature generation."*International Journal of Computer Applications* 62, no. 9 (2013).

One-pass HMQV

We provide a formal analysis of the protocol’s security showing many desirable properties such as sender’s forward-secrecy and resilience to compromise of ephemeral data.

We note that the precise connection between key-exchange and signcryption was established in the work of Gorantla et al. [10], and a comprehensive theory of KEM/DEM for signcryptions was developed by Dent [5, 7, 6].

[underline added]

Halevi, Shai, and Hugo Krawczyk. "One-pass HMQV and asymmetric key-wrapping." In *International Workshop on Public Key Cryptography*, pp. 317-334. Springer Berlin Heidelberg, 2011.

And one more thing!

- What is the role of perimeter security?
  - Is the “take whatever you want” model sufficient?
  - Should we do more work on ACLs, etc., for ICN?
  - Can we do the equivalent (or modify) P4 [1] (to make a big leap to programmability of ICN) to specify ACLs?

[1] http://www.p4.org

THANK YOU

parc®

A Xerox Company