FIU
FLORIDA INTERNATIONAL UNIVERSITY

NAMED DATA NETWORKING

CertCoalesce: Efficient Certificate Pool for NDN-Based Systems

Sanjeev Kaushik Ramani (Florida International University)
Alexander Afanasyev (Florida International University)

ACM Conference on Information Centric Networking 2020
Virtual Conference
September 30, 2020

FIU
FLORIDA INTERNATIONAL UNIVERSITY

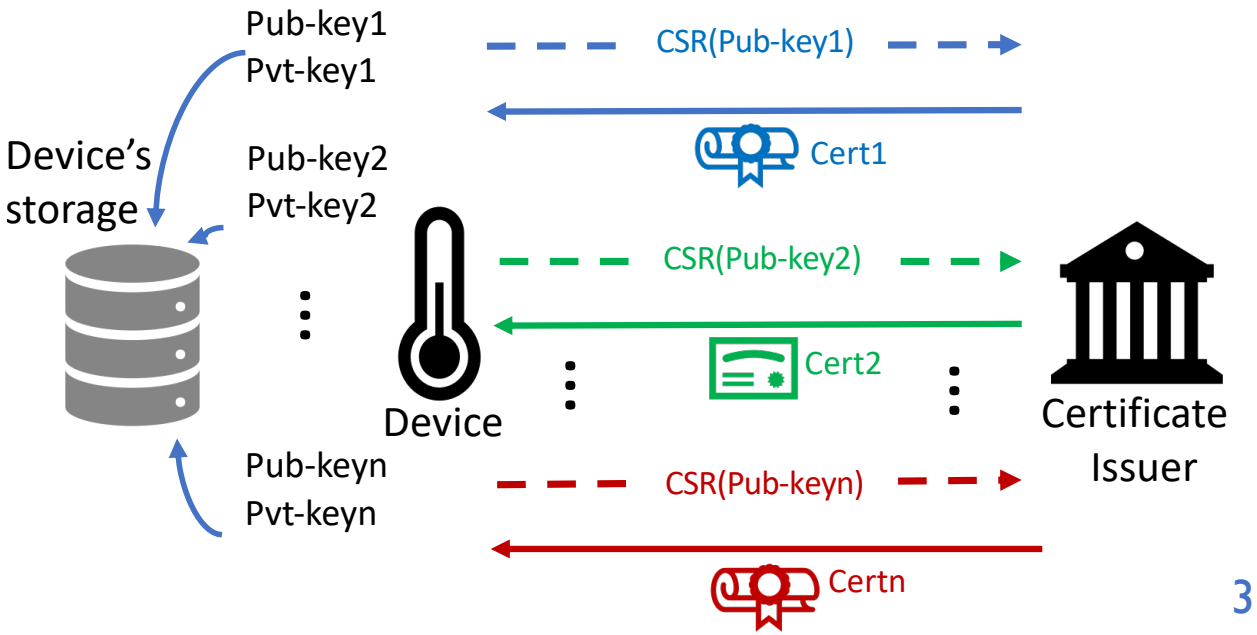
Need for Large Certificate Pools

- Least privilege separation
  - Separate key/certificate for /foo/bar/a, /foo/bar/b, /foo/bar/c/d, …
- Limiting exposure keys
  - Can eliminate the need for complex revocation mechanisms

FIU
FLORIDA INTERNATIONAL UNIVERSITY

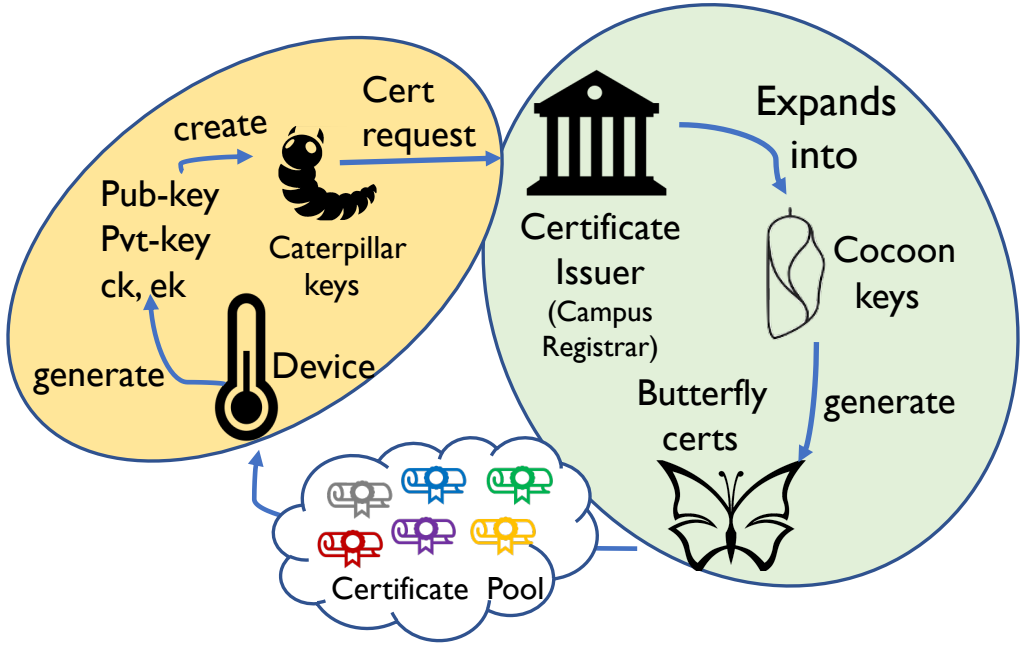
Inefficiencies of a Traditional Certificate Framework

- Unique public/private key pairs
  - Processing and storage requirements
- Each key requires a separate request
  - Network overhead
  - Storage requirement



CertCoalesce Traits

- A single* "master" caterpillar key bootstraps virtually unlimited set of private/public keys
  - Still require processing power, but less
  - Really need to store only the caterpillar key; specific key can be re-generated when needed
  - Still preserving forward secrecy
    - A compromised key does not reveal other keys/certificates in the pool
- A single certificate request with a caterpillar public key can issue pool of butterfly certificates
  - Butterfly certificate corresponds to a specific key in the caterpillar set
  - Only the device can infer the private version of the caterpillar keys

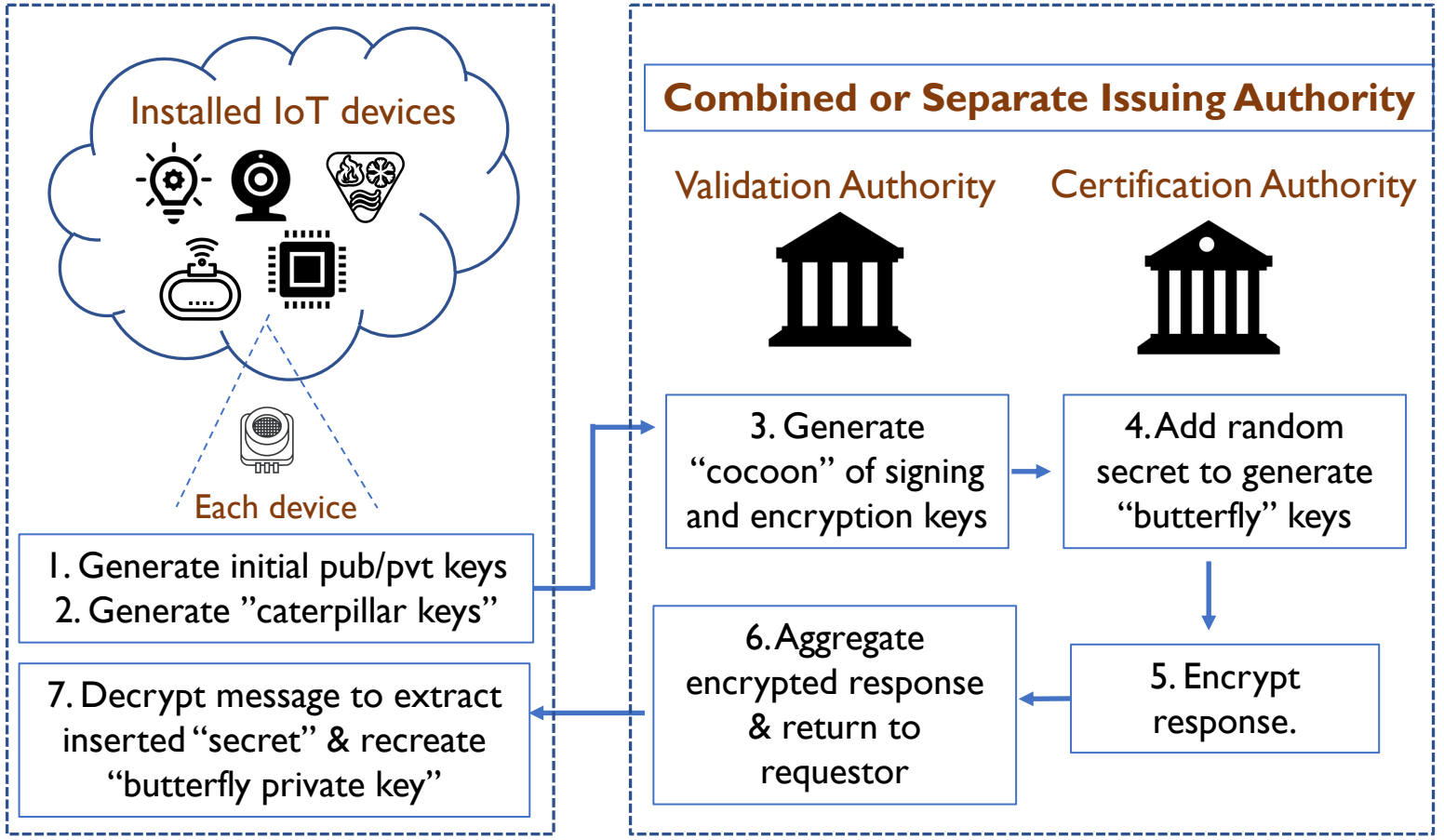


Based on elliptic curve cryptography

"Infinite" pools of private keys/certificates using a very limited storage requirements

CertCoalesce Design

Master caterpillar key required to derive private keys of received certificate pool. Name of this key with a small specialization of the ID part is: "/<identity-namespace>/KEY/caterpillar-<keyId>"



Size of cocoon set is based on the application and requester knows this (as preconfigured parameter or explicit notification from NDNCERT exchanges). Each key in the cocoon set is assigned name using a derivation function: "/<identity-namespace>/KEY/cocoon-<derived(i)>"

issuer generates (proactively, periodically, or on demand) butterfly certificates using the cocoon keys. Names of the butterfly certificates: "/<identity-namespace> /KEY/butterfly-<derived(i)>/Coalesce/<version>"