CertCoalesce: Efficient Certificate Pool for NDN-Based Systems

Sanjeev Kaushik Ramani
Florida International University
Miami, Florida
skaus004@fiu.edu

Alexander Afanasyev
Florida International University
Miami, Florida
aa@cs.fiu.edu

ABSTRACT

Named Data Networking (NDN) relies on public key signing to ensure integrity and authenticity for all data packets fetched in the network. One of the considerations for reliability of such signing is limiting the scope (what the key can sign) and time (how long the key can sign) of the public keys and their certificates, usually referred to as "least privilege principle." Traditionally, the public key certificates are issued for relative long periods of times measured in months or years; which requires considerations for certificate revocation, e.g, when the private key is lost or compromised. However, if the validity periods can be reduced to days or hours, the complex (and sometimes semi-broken) revocation mechanisms can be completely eliminated. This poster proposes such a mechanism—CertCoalesce certificates—to efficiently manage virtually unlimited pools of short-term certificates with limited networking, storage, and computational overheads. Specifically, a single certificate request with a "primary" key can be used to bootstrap the process of creating an unlimited number of short-term certificates for derivative private/public keys. Moreover, such certificates can be issued asynchronously—periodically pre-provisioned or upon request with an Interest—terminating issuance of future certificates when necessary. Moreover, CertCoalesce design owing to the underlying elliptic curve cryptography ensures that a compromised key from the pool of keys will not reveal information about other keys/certificates in the pool.

ICN '20, September 29-October 1, 2020, Virtual Event, Canada

ACM ISBN 978-1-4503-8040-9/20/09…$15.00
https://doi.org/10.1145/3405656.3420230

CCS CONCEPTS

• **Security and privacy → Security protocols**; • **Networks → Security protocols**; **Network security**; **Network privacy and anonymity**.

KEYWORDS

Information-Centric Networking, Named Data Networking, Butterfly key expansion, Certificate Pool.

ACM Reference Format:
Sanjeev Kaushik Ramani and Alexander Afanasyev. 2020. CertCoalesce: Efficient Certificate Pool for NDN-Based Systems. In *7th ACM Conference on Information-Centric Networking (ICN '20), September 29-October 1, 2020, Virtual Event, Canada.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3405656.3420230

1 INTRODUCTION

To realize least privilege principle for data authenticity, the Named Data Networking (NDN) architecture [3, 7] uses (a) structured naming to limit what data the key can sign and (b) validity periods to define how long a key (certificate) is considered valid to create a signature. Both require careful considerations for efficiency, usability, and damage control. Specifically, a coarser name granularity (higher level prefix for the key) or a longer validity (months or years) reduces overheads of getting the certificate but creates challenges to efficiently "revoke" the certificate when needed—requires complex revocation mechanisms. A finer granularity (more specific prefixes for the keys) or a shorter validity (days or hours) makes the damage control trivial, but can be potentially prohibitive, especially in highly constrained Internet-of-Things (IoT) environments.

In this poster, we propose CertCoalesce, a novel design that can efficiently manage virtually unlimited pools of public/private keys and corresponding certificates. CertCoalesce uses recently developed EC-based cryptographic constructs [1, 2] that allow independent yet secure derivation of unlimited sets of private keys, public keys, and certificates using a single "bootstrapping" ("Caterpillar"') request (Figure 1). With specialized derivation rules, producers can derive a private key for the specific name granularity and time validity, while the certificate issuers having ability to independently and

asynchronously (i.e., either periodically proactively or on demand) issue the corresponding certificates.

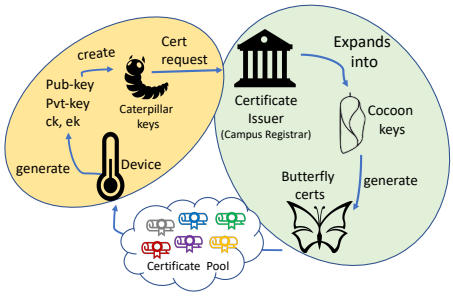

Figure 1: CertCoalesce certificate requisition (1 key-pair for a certificate pool)

With CertCoalesce, even a very constrained IoT devices can (virtually) own a large numbers of public/private key pairs and certificats, efficiently realizing least privilege principle of NDN. In other words, the device only needs to permanently and securely store the primary (Caterpillar) key; the active keys can be efficiently derived on demand. Moreover, devices do not need to store the certificates, just know the rules of how to construct their names, so they can be fetched from the issuer directly or any provisioned storage in the network.

2 CERTCOALESCE DESIGN

The overall structure of CertCoalesce certification process is highlighted in Figure 2. A typical CertCoalesce certificate request involves steps of *caterpillar* ("master") private key generation, *cocoon* public key derivation, *butterfly* certificate set generation, and derivation of the corresponding private keys.

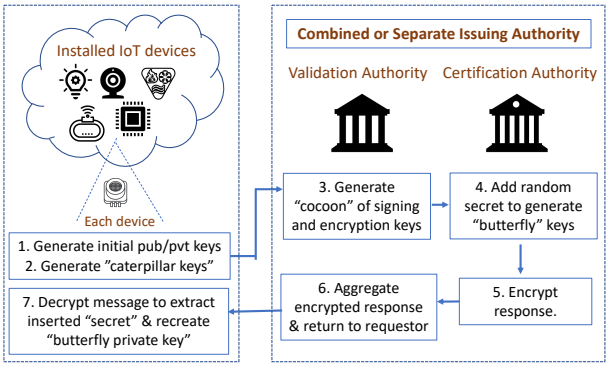

Figure 2: Overview of CertCoalesce design

2.1 Caterpillar Key Generation

CertCoalesce process starts with a target device generating an initial master key pair (caterpillar key that is stored) and a 128 bit AES key expansion function (Steps 1-2 in Figure 2). This master *caterpillar* key is needed to derive the private keys of the received certificate pool. Name of this key can follow the standard NDN conventions with a small specialization of the ID part is: "/<identity-namespace>/KEY/caterpillar-<keyId>"

2.2 Cocoon Keys Generation

After the public key part of the caterpillar key is delivered to the certificate issuer and the issuer successfully authorizes the request (e.g., using NDNCERT protocol [8]), the key is expanded into a set of *cocoon* keys (Steps 3-4 in Figure 2). The size of the cocoon set is based on the application and the requester knows this either as a preconfigured parameter as explicit notification from NDNCERT exchanges. Each key in the cocoon set is assigned name using a derivation function: "`/<identity-namespace>/KEY/cacoon-<derived(i)>`".

2.3 Butterfly Certificate Set Generation

In the following (Steps 4-6 in Figure 2), the issuer generates (proactively, periodically, or on demand) butterfly certificates using the cocoon keys. Specifically, by delaying issuance of a certificate, the issuer can effectively realize certificate revocation without the need for any special revocation mechanisms. Names of the butterfly certificates follow the general NDN conventions [5]: "`/<identity-namespace>/KEY/butterfly-<derived(i)>/Coalesce/<version>`".

2.4 Deriving Butterfly Private Keys

When the need arrives, the target device can generate necessary butterfly private key (e.g., a key that by convention, corresponds to the current time validity) and sign the data (Step 7 in Figure 2) using schemes like [6]. As an example for a 5 minute validity per certificate, the device will need to run this derivation a dozen times an hour if it continuously producing data.

3 SUMMARY

Adversaries in the system try to decipher the private keys corresponding to a received set of butterfly public keys in polynomial time [4]. However, the underlying cryptographic construct ensures that it is close to impossible to identify the derived private keys in polynomial time ensuring security against common attacks. Overall, CertCoalesce design involves creation of one pair of public and private keys for a pool of certificates ensuring minimal storage requirements for certificates and keys. The communication requirement for retrieving these large number of certificates is also reduced considerably as the CI can use the series of cocoon keys (validated using NDNCERT) to periodically generate derivative certificates and send them to the device for use.

4 ACKNOWLEDGEMENTS

This work was supported in part by the US National Science Foundation/Intel grant CNS 1719403.

REFERENCES

[1] [n.d.]. National Institute of Standards and Technology. Recommended elliptic curves for federal government use. https://csrc.nist.gov/projects/cryptographic-standards-and-guidelines

[2] [n.d.]. Security Credential Management System Proof–of–Concept. https://wiki.campllc.org/display/SCP/SCP1%3A+Butterfly+Keys

[3] Alex Afanasyev, Jeff Burke, Tamer Refaei, Lan Wang, Beichuan Zhang, and Lixia Zhang. 2018. A brief introduction to Named Data Networking. In *Proc. of MILCOM*.

[4] Steven D Galbraith and Pierrick Gaudry. 2016. Recent progress on the elliptic curve discrete logarithm problem. *Designs, Codes and Cryptography* 78, 1 (2016), 51–72.

[5] NDN Team. 2020. NDN Certificate Format Version 2.0. Online: http://named-data.net/doc/ndn-cxx/current/specs/certificate-format.html.

[6] Sanjeev Kaushik Ramani, Reza Tourani, George Torres, Satyajayant Misra, and Alexander Afanasyev. 2019. NDN-ABS: Attribute-Based Signature Scheme for Named Data Networking. In *Proceedings of the 6th ACM Conference on Information-Centric Networking*. 123–133.

[7] Lixia Zhang, Alexander Afanasyev, Jeffrey Burke, Van Jacobson, Patrick Crowley, Christos Papadopoulos, Lan Wang, Beichuan Zhang, et al. 2014. Named data networking. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 66–73.

[8] Zhiyi Zhang, Yingdi Yu, Alex Afanasyev, and Lixia Zhang. 2017. *NDN Certificate Management Protocol (NDNCERT).* Technical Report NDN-0050. NDN.