Intertrust: Establishing Inter-Zone Trust Relationships

Tianyuan Yu, Xinyu Ma
UCLA
USA

Hongcheng Xie
City Univ. of Hong Kong
China

Yekta Kocaoğullar
Sabancı University
Turkey

Lixia Zhang
UCLA
USA

ABSTRACT

An NDN network is made of named entities with various trust relations between each other. Entities are organized into trust zones. Each trust zone contains the entities under the same administrative control. This work-in-progress explores an approach to establishing trust relations *between* trust zones.

CCS CONCEPTS

• **Networks** → **Security protocols**; • **Security and privacy** → **Authentication**;

KEYWORDS

Named data networking, Information-centric networking, Trust management

ACM Reference Format:
Tianyuan Yu, Xinyu Ma, Hongcheng Xie, Yekta Kocaoğullar, and Lixia Zhang. 2022. Intertrust: Establishing Inter-Zone Trust Relationships. In *ACM ICN 2022 Demos and Posters (ICN '22), September 19–21, 2022, Osaka, Japan.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3517212.3559489

1 INTRODUCTION

In NDN [9] experimental deployment, we encountered scenarios where different trust zones [3] must establish trust relations to interact. A recent example is mGuard [1, 5] trial deployment on the NDN Testbed [2], where mGuard producers need to make routing announcements [6] into the Testbed. The NDN Testbed belongs to the Testbed trust zone, where routers are bootstrapped with the Testbed trust anchor and the anchor issued certificates, and verify routing announcements from other Testbed nodes through the shared trust anchor. On the other hand, all mGuard entities belong to the mGuard trust zone and they authenticate each others data, including routing announcements, using the shared mGuard trust anchor issued certificates.

To enable mGuard entities utilizing the Testbed for data exchanges, mGuard routers must be able to announce prefixes to the Testbed. This requires two functions: enabling the Testbed routers to authenticate packets generated by mGuard entities, and defining security policies that specify what functions mGuard packets are allowed to perform within the Testbed zone. This task of interconnecting mGuard and the Testbed zones raises a new research question of *how to establish inter-zone trust relations and secure communications between zones*.

This poster reports our initial investigation in answering the above question by proposing an inter-zone trust framework for establishing trust relations between different trust zones to secure inter-zone communications. Specifically, our proposed framework supports two procedures:

- *Zone Authentication* is designed to authenticate the trust anchor of an external trust zone.
- *Zone Authorization* is designed to check whether a Data packet from the external trust zone is produced by the legitimate producer.

2 BACKGROUND AND RELATED WORKS

An NDN trust zone is controlled by a single trust anchor which defines the trust schema for all entities within its zone [4]. This trust zone concept allows an NDN network to manage trust relations of networked entities under a single administrator, which plays the role of *trust zone controller*. A trust zone controller can administrate the trust zone as a result of two properties:

A zone controller owns the trust anchor. A trust anchor is a self-signed certificate and the termination point of cryptographic verifications within the zone [11]. Each entity within the zone instals the trust anchor and obtains its certificate and trust schema during its security bootstrapping [7]. The trust zone controller is able to autonomously administrate the trust zone because all zone entities can authenticate each other in the zone through the shared trust anchor.

A zone controller defines trust policies. Handling of all data packets must be authorized by the zone controller through the *trust schema* [8], which contains all trust policies an entity needs. For example, in a smart home IoT system [3, 10], the trust zone controller defines security policies that limit the access to high-value entities (e.g. locks) to the authorized residents but not by other entites(e.g. smart light bulbs).

3 INTERTRUST DESIGN

In this section, we describe our Intertrust model as a derivation of the intra-zone model in Section 2. To facilitate the demonstration, we define two trust zones as Z_a and Z_b, administrated by the zone controllers C_a and C_b respectively. Two self-signed certificates T_a and T_b represent the trust anchors of each zone, and two entities E_a and E_b have been bootstrapped into Z_a and Z_b respectively.

ICN '22, September 19–21, 2022, Osaka, Japan
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9257-0/22/09…$15.00
https://doi.org/10.1145/3517212.3559489

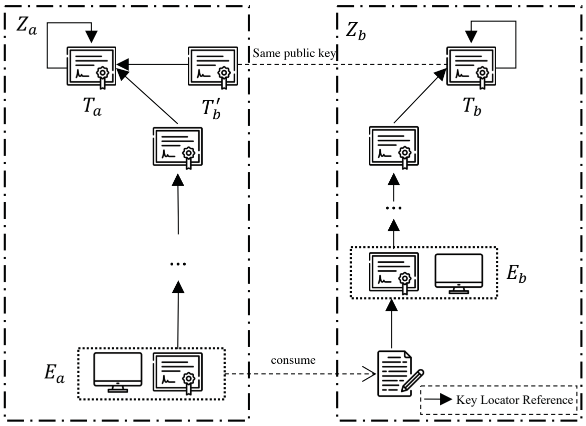

Figure 1: E_a and E_b are in zone Z_a and Z_b, respectively. Between T_a and E_a certificate, T_b and E_b certificate, there exists intermediate certificates. E_b produces a data packet and E_a consumes it.

Rationale. In order for E_a to consume data produced by E_b, E_a needs to validate E_b's certificate and check the trust schema to ensure E_b is a legit producer. As E_a's trust relations are managed by C_a, it is natural to let C_a authenticate E_b and add the trust policies for E_b. However, this mechanism will not be scalable if E_a is communicating with a large number of entities in Z_b. A solution can be *zone authentication and authorization*, which means C_a issues C_b a certificate to establish all Z_b entities' authenticity within Z_a. Then C_a specifies the trust policies for communicating with Z_b entities.

3.1 Authentication and Authorization

Keeping our rationale in mind, we designed Intertrust with a two-step layout: *zone authentication* and *zone authorization.* In this section, we introduce the concept of zone authentication and zone authorization with a model shown in Figure 1. We consider a scenario where E_b produces a data packet and E_a wants to consume it. E_a checks the data KeyLocator and learns that E_b is the producer. In this example, successful data consumption requires two procedures as prerequisites:

Zone Authentication. Since E_b's verification chain terminates at T_b, E_a needs to know whether T_b is trustworthy. This requires that Z_a authenticates Z_b by allowing T_b as an external termination point of cryptographic verifications. We refer to this procedure as *Zone Authentication.* To perform Zone Authentication, C_a must authenticate C_b and obtain T_b first. After obtaining T_b, C_a will sign a specific NDN certificate for T_b's public key, called Proof of Zone Recognition (PoR). A PoR certificate is named as */<Z_b name>/KEY/<T_b keyid>/<Z_a name encoded>/<version>*. Within the certificate name, *<Z_b name>* indicates Z_b name is an recognized zone name, *KEY* represents a keyword component, *T_b keyid* is the zone public key identifier, *<Z_a name encoded>* is Z_a in TLV encoded format, and *<version>* indicates the version number. Thus, T_b can be accepted as an external termination point by Z_a if it obtains its PoR from C_a. All entities in Z_b are authenticated by Z_a after Zone Authentication, as all the verification chains of their data packets terminate at T_b.

Zone Authorization. After E_b obtains its authenticity with Z_a through zone authentication, E_a needs to verify whether E_b is a legitimate producer in Z_a's trust model. This requires C_a (i) obtaining knowledge on Z_b internal naming convention; (ii) defining trust rules on the data produced by Z_b entities can be validated; (iii) installing the latest trust schema on all Z_a entities. We refer this procedure as *Zone Authorization.* After Zone Authorization, all Z_a entities can validate data produced by Z_b entities.

3.2 Data Consumption

After the above two prerequisites, when E_a consumes a data packet from E_b, E_a can perform the normal signing chain verification from E_b's certificate to T_b based on the latest trust schema. When E_a verifies T_b, it exploits the naming convention to fetch T_b's PoR under Z_a and verifies it with T_a. If T_b's PoR under Z_a passes, the corresponding data packet from E_b will be accepted by E_a.

3.3 Case Study

In this section, we use an example based on Figure 1 to discuss our proposed design. We assume that T_a's name is "`/A/KEY/1/self/v=0`" and T_b's name is "`/B/KEY/2/self/v=0`". An entity E_a in Z_a wants to consume a data packet *d* from an entity E_b in Z_b. To authenticate Z_b, the controller C_a should authenticate C_b and fetch T_b. Then, C_a will sign a PoR "`/B/KEY/2/</A>/v=0`", where "`</A>`" is the TLV encoded name "`/A`" as a single name component. C_a will also fetch Z_b's naming convention and update Z_a's trust schema.

When E_a receives *d*, it will verify it from E_b's certificate to T_b based on the latest trust schema. When the verification chain reaches T_b, E_a will use its zone name "`/A`" and the naming convention to get the PoR's name "`/B/KEY/2/</A>/v=0`" and fetch it. After fetching the PoR, E_a will use its trust anchor T_a to verify it. If verification passes, E_a will accept *d*.

4 DISCUSSION AND FUTURE WORK

Intertrust establishes trust relations between trust zones and enables secure inter-zone communications. It shares similarities with intra-zone bootstrapping (as defined in [7]). Both processes require out-of-band authentication and defining trust schema for the party. However, the authentication in Intertrust can be unilateral (*i.e.,* the consumer zone unilaterally authenticates the producer zone). Also, the external party in Intertrust already possesses a name.

As the next step, we plan to implement Intertrust, and leverage Intertrust to build an interoperable global system that consists of multiple trust zones. We also look forward to experimenting Intertrust on NDN Testbed and pushing for its trust model decentralization.

REFERENCES

[1] Dulal, S., Ali, N., Thieme, A. R., Yu, T., Liu, S., Regmi, S., Zhang, L., and Wang, L. Building a secure mhealth data sharing infrastructure over ndn. In *Proceedings of the 9th ACM Conference on Information-Centric Networking* (2022).

[2] Named Data Networking Project. Ndn testbed, 2022. https://named-data.net/ndn-testbed/.

[3] Nichols, K. Trust schemas and icn: key to secure home iot. In *Proceedings of the 8th ACM Conference on Information-Centric Networking* (2021), pp. 95–106.

[4] Nichols, K. Trust schemas and icn: Key to secure home iot. ICN '21, Association for Computing Machinery.

[5] The mGuard Team. A secure real-time data distribution system with fine-grained access control for mhealth research, 2022.

[6] The NDN Team. Prefix announcement protocol, 2022.

[7] Yu, T., Moll, P., Zhang, Z., Afanasyev, A., and Zhang, L. Enabling plug-n-play in named data networking. In *MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM)* (2021), IEEE, pp. 562–569.

[8] Yu, Y., Afanasyev, A., Clark, D., claffy, k., Jacobson, V., and Zhang, L. Schematizing trust in named data networking. In *Proceedings of the 2nd ACM Conference on Information-Centric Networking* (New York, NY, USA, 2015), ACM-ICN '15, Association for Computing Machinery, p. 177–186.

[9] Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., claffy, k., Crowley, P., Papadopoulos, C., Wang, L., and Zhang, B. Named data networking. *SIGCOMM Comput. Commun. Rev. 44*, 3 (July 2014), 66–73.

[10] Zhang, Z., Yu, T., Ma, X., Guan, Y., Moll, P., and Zhang, L. Sovereign: Self-contained smart home with data-centric network and security. *IEEE Internet of Things Journal* (2022).

[11] Zhang, Z., Yu, Y., Zhang, H., Newberry, E., Mastorakis, S., Li, Y., Afanasyev, A., and Zhang, L. An Overview of Security Support in Named Data Networking. *IEEE Communications Magazine 56*, 11 (November 2018), 62–68.