A security protocol for Information-Centric Networking in smart grids

Bárbara Vieira
Radboud University
Nijmegen, The Netherlands
b.vieira@cs.ru.nl

Erik Poll
Radboud University
Nijmegen, The Netherlands
erikpoll@cs.ru.nl

ABSTRACT

The C-DAX project aims at providing a secure overlay network, as an overlay over an IP network, that provides an information-centric network (ICN) tailored to the needs and the capabilities of smart grids. This paper addresses how end-to-end security can be enforced in information-centric networks by proposing a protocol based on the concept of identity-based encryption, a type of public-key cryptography.

1. INTRODUCTION

The EU FP7 project C-DAX (*Cyber-secure Data and Control Cloud*)[1] investigates an information sharing solution for the monitoring and control of smart grids based on an information-centric networking (ICN) solution as an overlay of IP. The C-DAX solution will provide a distributed data-cloud tailored to the specific needs of smart grids. In particular, it is intended to efficiently support the massive integration of renewables and be able to cope with a heterogeneous set of co-existing smart grid applications, running on devices and communicating over networks with widely varying capabilities when it comes to communication and computation speeds. Precursors to the C-DAX solution are overlay networking solutions developed at Bell-Labs [10, 11] (originally called *SeDAX*).

The general requirements of C-DAX cover fundamental system requirements that are required for the basic operation of the platform, such as configuration, communication, data management and security. Further, additional requirements are considered in C-DAX by considering three representative use cases. The first two use cases focus on the communication and control of measurement devices in the field. The first use case considers RTUs (Remote Terminal Units) and IEDs (Intelligent Electronic Devices) communicating with the Distribution Control Center (DCC). The second use case considers the communication between PMUs (Phasor Measurement Units) dispersed in the electricity system and PDCs (Phasor Data Concentrators), state estimator units, and SCADA systems. The third use case considers the future retail energy market and the communications needs for negotiations between consumers, owners of the distributed generation units (say solar power stations and wind turbines), and possible intermediaries.

These use cases have a different number of parties involved in the communication (e.g., many PMUs and a few PDCs; many home meters and a few energy providers) and rely on different network topologies with different data rates (e.g., power line communication, optical fibre, etc.). One feature common to all use cases is that the messages being communicated are relatively small (especially when compared to ICN solutions for, say, distributing video content), although the volume of messages may be large. More information about the use cases is available in [15].

This document first sketches the basic ideas behind the C-DAX solution and considers the general security issues underlying communication models based on information-centric principle (in particular the one adopted in C-DAX) in the remainder of Section 1. It then proposes a security solution to enforce end-to-end security between smart grid applications running as C-DAX clients.

1.1 Information-Centric Networks and C-DAX

What is ICN?.

Whereas traditional networking solutions aim at providing point-to-point connections between locations, in an information-centric network (a.k.a. content-centric network) [1, 12] the content plays a central role, rather than the location where this content happens to be stored, its origin, or is destination. Hence, communication primitives do not involve naming schemes for the identities of senders and receivers, but rather a naming schemes for the content.

The main advantage of such a solution is that it provides more flexibility than traditional, host-centric solutions, especially when there are many parties exchanging and sharing information. Communication may not only be one-to-one, but can also be many-to-one, one-to-many, or many-to-many. ICN also has some inherent security advantages, discussed in more detail below.

The C-DAX platform.

Conceptually, one can see the C-DAX overlay as a distributed information-centric network. As illustrated in Fig. 1, the C-DAX platform consists of two major components: the

[1] `http://www.cdax.eu`

SEGS'13, November 8, 2013, Berlin, Germany.

http://–enter the whole DOI string from rightsreview form confirmation.

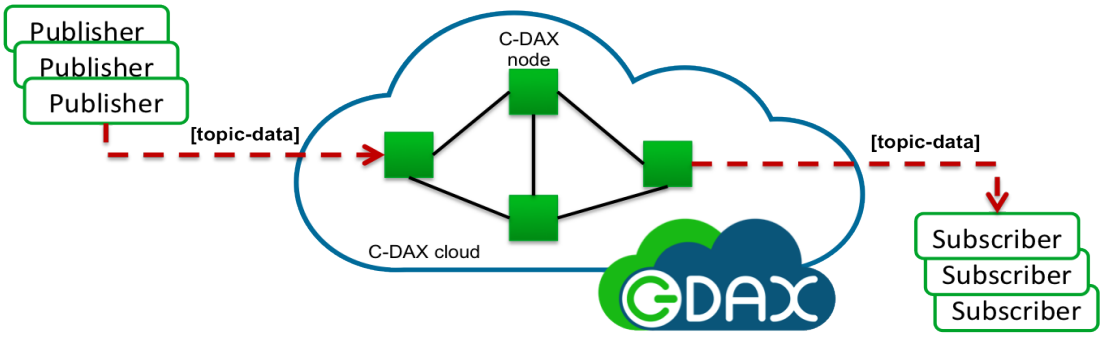

Figure 1: C-DAX **architecture**

C-DAX middleware that provides publisher-subscriber interfaces to clients hosting smart grid applications and the C-DAX cloud which consists of logically interconnected C-DAX nodes which are responsible for the resolution and delivery of messages exchanged between publishers and subscribers in a resilient, self-configurable, and scalable manner. The main idea of C-DAX is that, instead of applying host-centric and point-to-point communication, it supports group communication that is *data-centric* (i.e., its concepts are developed around the data being communicated) and *topic-based* (as the routing of data is based on topic identifiers).

Information is organised in so-called topics (i.e., elements of information sufficiently characterising data units easily identifiable by the clients) which are uniquely identified by a name and stored on certain C-DAX nodes. A publisher generates information for a specific topic (i.e., topic-data) and sends it to the C-DAX cloud. Subscribers can express interest in a specific topic and subscribe to information about it. C-DAX nodes are hosts geographically distributed in the C-DAX cloud that cache/store the published data and forward it to the interested subscribers. Topic-data is distributed over C-DAX nodes and possibly replicated at different locations. Neighbouring C-DAX nodes in the overlay can forward information to each other.

Simply put, one can think of the C-DAX overlay as a bulletin board, or a collection of bulletin boards, where clients can publish information on certain topics or subscribe to receive information on certain topics, in the style of the publish/subscribe paradigm [5]. However, the data is not in one location but is physically distributed over many places – over the cloud of C-DAX nodes – but in manner that is transparent to the clients.

Amongst all the requirements identified for the C-DAX platform, it must support confidentiality, integrity and authenticity of communication messages in an end-to-end and scalable manner. Accessing computation resources or data needs to be strictly controlled in a fine-grained way. The system must be protected against accidental failures or intentional cyber-threats such as Denial of Service (DoS) attacks and malware intrusion.

1.2 Security issues in ICN

An inherent security advantage of solutions such as the one adopted in C-DAX is that the clients of the C-DAX cloud – i.e., the senders and receivers of data – need not know each others IP addresses. This decoupling of senders and receivers reduces the risk of network-borne attacks. This is especially relevant as these clients are often machines with limited capabilities, such as measurements devices in the field or home meters, which do not have many resources to withstand attacks. Another advantage is that to improve availability in the case of failures (i.e., resilience), the C-DAX cloud can replicate data on different nodes placed at different locations, without clients having to be aware of this.

There are some subtle differences between securing traditional point-to-point communications and securing an information-centric networking solution like C-DAX.

For point-to-point communication there are standard security solutions such as TLS/SSL or IPSec. These typically involve the use of short-term symmetric session keys exchanged by means of longer-term asymmetric keys, combining the advantages of fast symmetric crypto with the convenience of key management of – slower but more flexible – asymmetric crypto. Moreover, as communications in information-centric networks typically involve more than two parties, sharing of symmetric keys has bigger security impact, as the group of parties sharing the key is larger.

Another difference is that in point-to-point communication we want to secure a stream of information (or two streams, one in each direction), whereas in an information-centric network we have to secure individual messages. Standard solutions to secure data streams provide a standard notion of *session* integrity and confidentiality. This means that integrity of the order of messages in the stream is automatically guaranteed; changing the order of message in the stream, or replaying them, will be detected. When securing individual messages in an ICN solution we do not get these properties for free, but *sequence numbers* or *time stamps* have to be included to guarantee freshness of messages and the integrity of order of messages. Indeed, with two parties publishing data on the same topic, there is a difference between integrity of the sequences of messages each individual publisher publishes, and integrity of the interleaving; the former can be guaranteed by sequence numbers per publisher, but the latter cannot.

So, as we will detail further in this document (Section 2), for an information-centric solution we cannot rely on standard security solutions such as TLS/SSL, but we have to come up with a scheme for signing and encrypting data to ensure integrity and confidentiality. On the positive side, any such security solution for an information-centric network will secure content rather than connections, and hence

naturally provides the *end-to-end security*, as advocated in smart grid standards such as IEC 62351 [21].

2. REASONING ABOUT SECURITY SOLUTIONS FOR ICN

2.1 Content-based security

In standard communication models (i.e., host-to-host communication) trust in the content is intrinsically tied to the trust in the host *where* the information comes from and *how* the content was retrieved. As stated before, the assurance that the data came from the intended source and was not tampered with neither eavesdropped during the transmission, is given by standard cryptographic protocols such as TLS/SSL or IPsec.

ICN decouples *where* of the information comes from, from *what* type of content one wants to retrieve. Hence, security constructions to authenticate the content itself are much more relevant than schemes that can only be used to authenticate its source. *Content-based security* (as opposed to connection-based security) must then allow the users to retrieve and authenticate the information regardless of knowing where it comes from and how it is transported.

In the specific context of C-DAX, information has to be sent through the C-DAX cloud: publishers send topic-data to the cloud that will be forwarded (by the cloud) to the intended subscribers. Subscribers must be able to validate that topic-data actually originates from valid publishers, but they do not need to know publisher's location/identity. Moreover, the trust placed in the C-DAX cloud has to be as minimal as possible, i.e., subscribers must be able to authenticate the content without placing any trust in the cloud. More concretely, a C-DAX client must be able to verify the authenticity of the data received, irrespective of how the information is forwarded or retrieved. It is then imperative to reason about different solutions that can be used to secure information-centric communication models such as the one adopted in C-DAX.

2.2 Why do standard crypto protocols not suffice?

As in C-DAX, in the new generation of smart grid networks a massive generation of data is expected from different measuring devices. For instance, the future power grid will support advanced monitoring infrastructures (e.g., advanced phasor measurement units) that will be able to provide real-time information to the SCADA system to develop a new class of optimal control functions.

Trying to adapt the standard cryptographic protocols to enforce end-to-end security between C-DAX clients has several implications. First of all, a pertinent observation is that C-DAX supports (beside one-to-one) one-to-many and many-to-many communication types whereas standard protocols are designed to establish a secure session between two end-points. Although, solutions have already been proposed in the literature to extend these protocols to multicast group communication (e.g., standard RFC 5374 on *Multicast Extensions to the Security Architecture for the Internet Protocol*), such solutions do not scale when considering communications between many-to-many end-points, specially when very high data rates with very low latency is expected, as is the case in C-DAX (e.g., the second use case considered in C-DAX involves PMUs to provide real-time estimations of the state of the grid; here data throughput is extremely high and the allowed time delays are extremely small).

For instance, extending TLS to secure group communication (in C-DAX), would imply either establishing as many sessions as the number of the recipients or creating a group session key. Establishing a unique session with each recipient seems a more appropriate solution (as single keys imply weaker security guarantees); however, the number of encryptions of the same data (a.k.a. ciphertexts) needed, then grows linearly with the number of recipients. Moreover, using TLS for group communication (and additional multicast extensions) requires interactive communication between the end-points (i.e., publishers and subscribers) and does not provide natural support for public verifiability (e.g., if authentication in the cloud is crypto-based – signature or message authentication code – it is not straightforward how the C-DAX cloud itself can actually authenticate topic-data to filter out malicious traffic; notice that the cloud cannot decrypt and then re-encrypt topic-data, otherwise end-to-end security would be compromised).

Another important aspect is that standard protocols usually rely on public-key cryptography, which in turn relies on the use of PKI certificates to distribute and check public keys. These certificates can be either stored at the devices or sent at beginning of the communication. Storing a huge amount of certificates (or just even the associated public keys) has a big impact on scalability. On the other hand, certificate transmission over limited bandwidth communication lines (e.g., power line communication) may be impractical.

Summing up, standard protocols were not originally designed to enforce end-to-end security in communication models like the one adopted in C-DAX, so we have to consider new protocols for this.

2.3 Possible security solutions

We reason about different possible solutions to enforce end-to-end security between publishers and subscribers in C-DAX.

Symmetric keys per topic.

A very simple solution is to use symmetric keys per topic: all the clients that publish or subscribe to information on a specific topic have to hold the corresponding key. (Different keys could be used for generating message authentication codes and for the encryption/decryption). This solution is adopted in the SSTP protocol proposed by Bell-Labs to secure smart grid networks [8]. Although this solution is very efficient, clearly it gives low security guarantees. For instance, whenever one client is compromised all the past and future communications are also compromised. Moreover, if no access control policies are being enforced (for dedicated networks it might be case, e.g., dedicated networks within the substations) both publishers and subscribers can publish and subscribe and there is no way to detect if it was a valid publisher that actually published the information or not.

A more elaborated solution is to use diversified symmetric keys: a trustworthy third party generates a (symmetric) master key per topic and derives several symmetric keys from that key for each publisher and the master key is given to the subscribers. This solution is actually adopted in the REMP protocol, also proposed by Bell-Labs [9]. Section 7

provides a more detailed overview of the REMP protocol. Although a solution adopting diversified symmetric keys offers better security guarantees than just using a symmetric key per topic, it requires that subscribers are trustworthy entities (i.e., they will not use the master key to publish illegitimate topic-data on behalf of some publisher), which might not be the case, for instance in the retail energy market, where the type of clients varies from energy consumers to external companies providing different types of services (e.g., providing smartphone applications to manage home meters measurements).

Asymmetric keys per client.

Assuming that asymmetric (long-term) keys are given to each client and are assigned to their identity, a valid solution is to use them to enforce end-to-end security between C-DAX clients. In this solution, publishers encrypt topic-data with the subscribers public key and sign it with their own private key. Subscribers decrypt with their own secret key and verify with the public key of the publisher.

Naturally this approach presents some disadvantages: (1) publishers have to store all the public keys of subscribers; and (2) have to create as many encryptions of the same message as the number of subscribers, which introduces a huge overhead in the system performance. Besides, it completely neglects the advantage of information-centric networks, where publishers and subscribers do not have to know each other to be able to communicate.

Asymmetric keys per topic.

Adopting asymmetric keys per topic seems to be a better approach. Two pairs of public/secret keys are assigned to each topic. Each client holds a pair of public/private keys (PK, SK): Publishers have the encryption public key PK_e and the signing private key SK_s of the topic, and subscribers the corresponding encryption secret key SK_e and signing public key PK_s. Although a fine grained access control can then be ensured by the key distribution, this solution implies storing as many key pairs as the number of topics that a client publishes on or subscribes to. Besides, if content authentication based on signatures is not just done by the clients that are the end-recipients of this content, but is already done in the cloud by the nodes handling this data, this requires storing all the signing public keys PK_s for all existent topics, in all C-DAX nodes. If public keys are attached to certificates and the number of topics is extremely high, the space required to store all the certificates, which will increase linearly in the number of topics, will be large. In this paper, we propose a different solution based on assigning asymmetric keys per topic, but where the storage space required is minimal.

Contributions.

As previously stressed, current cryptographic protocols do not directly target data-centric communication architectures: decoupling location from identity imposes significant challenges to actually authenticate the content itself. Our effort in this paper is to contribute to this area, by adapting existing identity-based cryptographic schemes to provide end-to-end security between publishers and subscribers in the C-DAX overlay network. We introduce the notion of content-based signcryption (CBS) based on the concept of identity-based signcryption, and we propose a CBS scheme for information-centric based networks, by relying on two existing identity-based schemes for encryption and signatures.

Paper organisation.

Section 3 introduces the concepts of identity-based signcryption and content-based signcryption. Section 4 proposes an efficient scheme and how it can be applied to C-DAX. Then Section 5 reasons about its security and Section 6 evaluates the proposed scheme in terms of performance. Finally, Section 7 describes related work available in the literature and Section 8 concludes and presents directions for future work.

3. DEFINITIONS

3.1 Identity-based signcryption

The concept of identity-based signcryption (IBS) was introduced by Malone-Lee [14] and combines the notions of identity-based encryption and signcryption.

Signcryption was introduced by Zheng [24] in 1997; the idea is to combine the functionality of encryption and signature schemes in a single one, in a more efficient way.

Identity-based encryption (IBE) is an asymmetric encryption scheme where existing identifiers for entities in a system (e.g., email addresses or telephone numbers) are reused in the construction of public keys. The idea is that this avoids the need for certificates, as clients can use identifiers that they already know as public keys. This is an advantage for devices with limited storage space and communication links with limited bandwidth (as is the case for parts of the smart grid infrastructure), because there is no need to exchange and validate the certificates. All this does require a trusted authority, usually known as the *private key generator* (PKG), to issue all secret keys and some system parameters. The concept of IBE was introduced by Shamir [19] in 1984 but the first practical implementation was only proposed in 2001 by Boneh and Franklin [3].

Usually, IBE protocols are based on pairings (i.e., special bilinear maps defined over mathematical groups) and security relies on the hardness of solving mathematical problems such as the Bilinear Diffie-Hellman problem (BDHP). These concepts are introduced in Section 4.

3.2 Content-based signcryption

In order to try to tackle some of the problems identified in the solutions previously described, we propose a content-based signcryption (CBS) scheme for information-centric communication models derived from the original IBS scheme introduced by Malone-Lee [14]. As in the original scheme the secret keys are generated by a trusted third party, the so-called private key generator (PKG), and tied to the global system parameters.

The CBS scheme consists of four algorithms: the first algorithm is executed at the system setup (by the PKG) and derives the systems parameters (public and private parameters); the second algorithm is used to derived secret keys per topic-group, being executed every time a new topic is added to the system; and the third and fourth algorithms are used to publish and subscribe topic-data, respectively.

1. *Setup.* This takes as input a security parameter η and derives the public parameters of the system, as well

as two different master secret keys: one for encryption/signing MSK_s (the master secret key for encryption & signing) and another for decryption/verification MSK_d (the master secret key for decryption & signature verification). Each master secret key is associated to a (corresponding) master public key. The public parameters include the definitions of the message and ciphertext space, and public master keys. The PKG runs the setup algorithm, but does not reveal the master secret keys.

2. *Key extraction.* This takes as input a topic identifier ID (a bit-string used as a public key), the public parameters (generated in the setup) and the master secret keys. Derives an encryption-signing secret key associated to ID from MSK_s and a decryption-verification secret key (also associated to ID) derived from MSK_d. Each topic has a public key (the identifier ID) and two secret keys: one for encryption-signing SK_s and another for decryption-verification SK_d.

3. *Signcrypt.* This takes as input the topic identifier ID, the encryption-signing secret key SK_s associated to ID, the public parameters and a message, and outputs the ciphertext encrypted and signed with the pair (ID,SK_s).

4. *Unsigncrypt.* This takes as input the decryption-verification secret key SK_d, the topic identifier ID, the public parameters and the ciphertext (including the signature). If the signature verifies, then it decrypts and outputs the message, otherwise an error value $\perp$ is output.

For consistency it is required that:

$$if\ \textbf{signcrypt}(m, ID, SK_s) = \mathcal{C}$$
$$then\ \textbf{unsigncrypt}(\mathcal{C}, ID, SK_d) = m.$$

Realising CBS in C-DAX .

To apply the CBS described above to C-DAX we need a trusted third party to play the role of PKG. As it is shown in Figure 2[2], there is a security server performing this function, i.e. generating the system parameters and issuing the secret keys of the C-DAX clients.

At the system setup, the security server generates the public parameters and the master secret keys. Afterwards, for each topic identifier it derives an encryption-signing secret key SK_s and decryption-verification secret key SK_d. Every publisher that is allowed to publish information on a certain topic gets the corresponding SK_s from the security server, and every subscribers that is allowed to subscribe to information on a topic gets SK_d. All system entities can get the public parameters from the security server.

To publish data in the C-DAX cloud, each publisher has to encrypt and sign the message with SK_s. On receiving data from the cloud, the subscriber can decrypt and verify it using SK_d. In short, for each topic the security server derives the triple (ID, SK_s, SK_d), but clients involved in the communication, i.e., publishers and subscribers, only have access to a single pair (ID,SK), where $SK = SK_s$ for publishers and $SK = SK_d$ for subscribers. The cloud itself (or any other entity than the security server, with no permission either to publish/subscribe) never gets access to the secret keys of the topic, otherwise, end-to-end security between publishers and subscribers could be compromised.

[2]For simplicity, we abstract publishers/subscribers and C-DAX nodes as single entities.

Remarks.

As CBS schemes are based on identity-based cryptography, they do not require certificates to authenticate public keys: the public key is the topic-name itself and all the C-DAX nodes and clients can have access to it.

Another interesting aspect is the clear distinction between the roles of publishers and subscribers: only publishers can write information in the cloud and the subscribers can read. This fine-grained access control is introduced in the CBS scheme by distributing different secret keys by the different players of the system. Observe that (in provable secure CBS schemes) a subscriber can never write valid information in the cloud without the associated encryption-signing secret key (because the signature generated at encryption time must be unforgeable) and (such as in the standard public key encryption schemes) publishers can never read information from the cloud without the associated decryption/verification key.

4. THE SCHEME

The simple CBS scheme we propose makes use of *bilinear maps*. We start the description of the proposed scheme by first introducing this concept.

Bilinear maps.

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups of order p (the order of the groups depends of the security parameter η), P the generator of $\mathbb{G}$, and $\hat{e} : \mathbb{G}\times\mathbb{G} \rightarrow \mathbb{G}_T$ an *admissible* symmetric bilinear map, where the following properties hold:

- *Bilinearity*: $\forall P, Q \in \mathbb{G}$. $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$;
- *Non-degenerate*: $\hat{e}(P, P) \neq id_{\mathbb{G}_T}$ (i.e., not all the pairs in $\mathbb{G} \times \mathbb{G}$ map to identity in $\mathbb{G}_T$);
- *Efficiently computable*: $\forall P, Q \in \mathbb{G}$ there is an efficient algorithm to compute $\hat{e}(P, Q)$.

As in [3], the modified Weil and Tate pairings are admissible applications, where $\mathbb{G}$ is a cyclic subgroup of an additive group defined by a supersingular elliptic curve $\mathbb{E}(\mathbb{F}_p)$ and $\mathbb{G}_T$ is a multiplicative cyclic subgroup of a finite extension of $\mathbb{F}_p$. More details on bilinear maps can be found in [17].

Description.

The proposed CBS scheme is composed by four polynomial-time algorithms:

Setup(η) Given the security parameter $\eta \in \mathbb{Z}^+$ the algorithm works as follows:

1. Generate a prime number p (which depends of η), two cyclic groups $\mathbb{G}, \mathbb{G}_T$ of order p, and an *admissible* symmetric bilinear map $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ as described above. Choose a random generator P of $\mathbb{G}$.

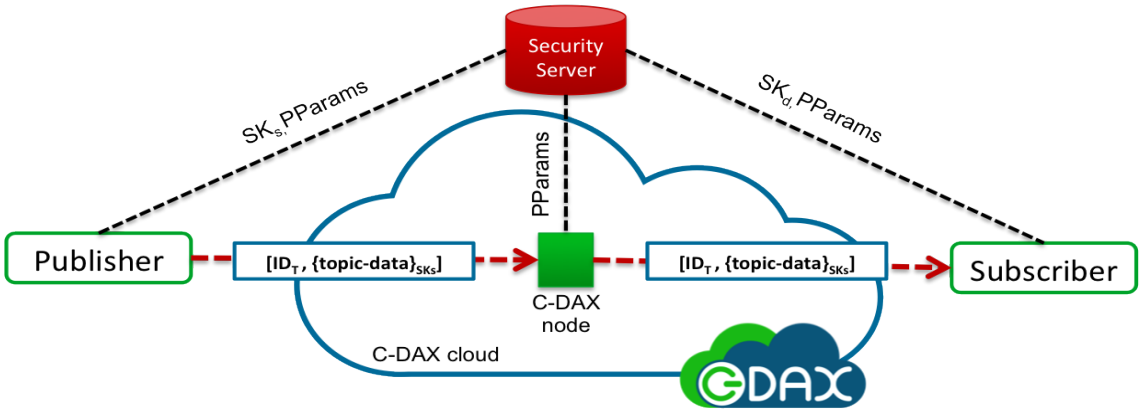

Figure 2: Content-based security for C-DAX

2. Pick a random t and b in $\mathbb{F}_p^*$ (notation: $t, b \leftarrow_\$ \mathbb{F}_p^*$) as master secret keys:
   - t is the decryption-verification master secret key
   - and b is the encryption-signing master secret key

   such that $t \neq b$ and set $T = tP$ and $B = bP$ as associated master public keys.
3. Choose three cryptographic hash functions as follows:
   - $H_1 : \{0,1\}^* \rightarrow \mathbb{G}^*$,
   - $H_2 : \mathbb{G}_T \rightarrow \{0,1\}^*$,
   - $H_3 : \{0,1\}^* \times \mathbb{G} \rightarrow \mathbb{F}_p^*$.

The message space is $\mathcal{M} = \{0,1\}^*$ and the ciphertext space is $\mathcal{C} = \{0,1\}^* \times \mathbb{G}^* \times \mathbb{G}^* \times \{0,1\}^*$. The public parameters are:

$$\texttt{params} = (H_1, H_2, H_3, \mathbb{G}, \mathbb{G}_T, \hat{e}, P, T, B, p).$$

KeyGen(params, id_A, t, b**)** Given the identifier $\texttt{ID}_A \in \{0,1\}^*$ of topic A, the decryption-verification key is simply $d_A = tQ_A$ and the encryption-signing key is $b_A = bQ_A$, where $Q_A = H_1(\texttt{ID}_A)$ is the public key of topic A.

Encrypt(params, b_A, id_A, m**)** To publish (encrypt and sign) a message $m \in \mathcal{M}$ on topic $\texttt{ID}_A$, the algorithm executes the following steps[3]:

$$\begin{aligned}
&Q_A \leftarrow H_1(\texttt{ID}_A)\\
&a \leftarrow_\$ \mathbb{F}_p^*\\
&k \leftarrow \hat{e}(Q_A, T)^a\\
&R \leftarrow aP\\
&c \leftarrow m \oplus H_2(k)\\
&h \leftarrow H_3(c||\texttt{ID}_A, R)\\
&S \leftarrow (a+h)^{-1} \cdot b_A
\end{aligned}$$

Output: the algorithm outputs the tuple $(c, R, S, \texttt{ID}_A)$.

Decryption(params, d_A, CT**)** To decrypt/verify a ciphertext $CT = (c, R, S, \texttt{ID}_A) \in \mathcal{C}$ the algorithm executes the following steps:

$$\begin{aligned}
&h \leftarrow H_3(c||\texttt{ID}_A, R)\\
&V \leftarrow \hat{e}(R + hP, S)\\
&k \leftarrow \hat{e}(d_A, R)\\
&m \leftarrow c \oplus H_2(k)
\end{aligned}$$

Then m is accepted as a valid message *iff* $V = \hat{e}(B, Q_A)$.

Note that the scheme is consistent since:

$$\begin{aligned}
\hat{e}(Q_A, T)^a &= \hat{e}(Q_A, tP)^a = \hat{e}(tQ_A, aP) = \hat{e}(d_A, R)\\
\hat{e}(R + hP, S) &= \hat{e}(aP + hP, S)\\
&= \hat{e}((a+h)P, (a+h)^{-1} b_A)\\
&= \hat{e}(P, bQ_A) = \hat{e}(B, Q_A).
\end{aligned}$$

This signcryption scheme results from the composition of two existing schemes. It directly derives from the Boneh-Franklin identity-based encryption scheme [3] and is an adaptation of the McCullagh-Barreto identity-based signature scheme [16].

An important observation is that encryption of a message m can be done using any symmetric cipher which takes as input a message $m \in \{0,1\}^n$ and a key $k \in \{0,1\}^n$ for some predefined length n (when considering $\mathcal{M} = \{0,1\}^n$, $\mathcal{C} = \{0,1\}^n \times \mathbb{G}^* \times \mathbb{G}^* \times \{0,1\}^*$ and $H_2 : \mathbb{G}_T \rightarrow \{0,1\}^n$). Nevertheless, the use of symmetric ciphers might affect the performance of the proposed protocol.

The proposed scheme and C-DAX .

Figure 3 shows how the proposed scheme can be used to enforce end-to-end security (end-to-end confidentiality and integrity) in C-DAX. After running the setup algorithm, the security server has to distribute the public parameters (PParams) to C-DAX clients and nodes[4]. Then it generates the publishing (i.e., encryption-signing) and subscribing (i.e., decryption-verification) keys for each topic. The publishing key b_A of topic A is given to the authorised publishers and the corresponding subscribing key d_A to the authorised subscribers [5].

[3] As usual $k \leftarrow_\$ \mathbb{F}_p^*$ denotes: choose a random k in $\mathbb{F}_p^*$ and $c||\texttt{ID}_A$ denotes the concatenation of c and $\texttt{ID}_A$.

[4] All C-DAX clients and nodes need to have a valid copy of the public parameters, since invalid copies can jeopardise the security of the system.

[5] Topic secret keys have to be securely distributed, but this can be seen as an orthogonal problem and we do not address it in this work.

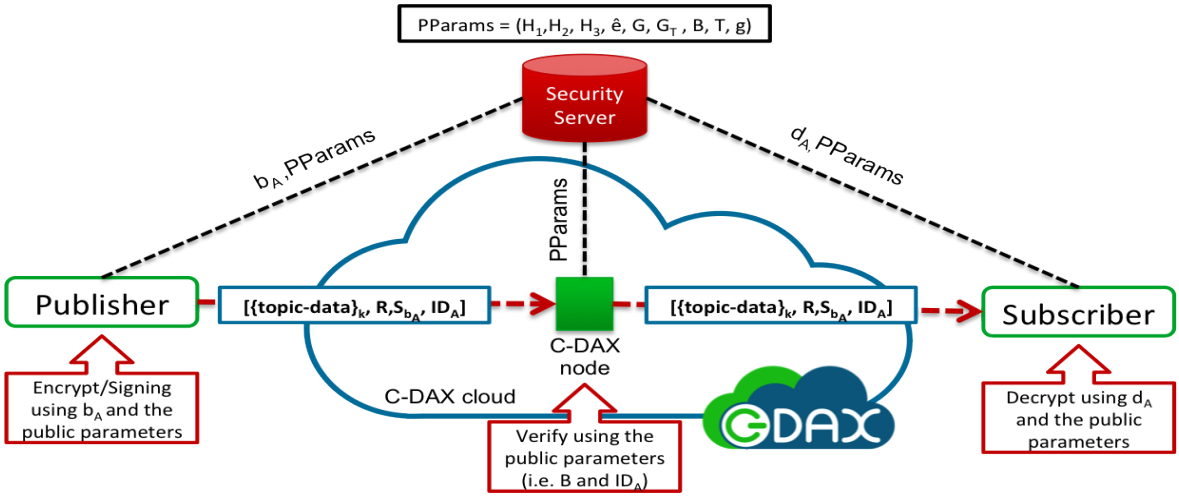

Figure 3: CBS scheme applied to C-DAX infrastructure

To publish, each publisher creates an encryption c (denoted as $\{topic-data\}_k$) of the topic data using topic's public key and generates a valid signature S using b_A. Then the publisher outputs the ciphertext $(c, R, S, \mathtt{ID}_A)$ and sends it to the C-DAX cloud. The cloud can authenticate the message (i.e., verify that it came from an authorised publisher) received from the publisher by computing $Q_A = H_1(\mathtt{ID}_A)$ and $h = H_3(c||\mathtt{ID}_A, R)$, and verifying if $\hat{e}(R+hP, S) = \hat{e}(B, Q_A)$. Since the value of $\hat{e}(B, Q_A)$ is always constant for all the topic-data on topic A, it can be pre-computed and stored at the node. If verification succeeds, the cloud can forward the message to the intended subscribers. Upon receiving topic-data, the subscriber first verifies if the signature S is valid and then decrypts c using the subscribing key d_A.

5. SECURITY

This section considers the security of the proposed scheme. We start by articulating the security properties that a signcryption scheme has to satisfy.

5.1 Security properties

A standard signcryption scheme must satisfy the security properties of both encryption and signature schemes:

- *Confidentiality* – it should not be possible for an adaptive attacker to recover the plaintext from the ciphertext without knowing the receiver's private key;
- *Unforgeability* – an adaptive attacker is not able to forge a valid signature without the knowledge of the sender's private key;
- *Authentication/Non-repudiation* – the sender cannot deny his/her signcrypted ciphertexts;
- *Integrity* – the receiver can verify that the message received was not modified;
- *Public verifiability* – any third party can verify the validity (i.e., authenticity) of the signcrypted ciphertext.

The standard way to prove that an asymmetric scheme satisfies the confidentiality and unforgeability properties is to prove that it is chosen ciphertext secure and existentially unforgeable against *adaptive attacks*[6]. Simply put, chosen ciphertext security for IBE systems considers an attacker who is able to adaptively choose the keys of identities he/she wishes to attack. Intuitively, an attacker is going to be challenged on identities of his/her choice and the scheme is chosen-ciphertext secure if given two messages and a ciphertext encrypted/signed under an identity of his/her choice, he/she is not able to discriminate which message corresponds to the ciphertext.

A scheme is said to be existentially unforgeable under adaptively chosen-message attacks if the attacker, having access to message-signature pairs of messages of his/her choice encrypted and signed under keys of identities of his/her choice, is not able to create a valid signature for a message of his/her interest. Authentication, non-repudiation, and integrity are satisfied if it is possible to prove that no one, apart the owner of the secret key, is able to forge a valid signature of a message of his/her interest. By proving that the scheme is existentially unforgeable, we also prove that the scheme satisfies these properties. Detailed definitions of chosen-ciphertext security and existential unforgeability of the original schemes are available at [3, 16].

5.2 Security evaluation

The security of cryptographic protocols usually relies on the difficulty of solving what is believed to be hard computational problems. In particular, the security of the scheme proposed relies on the intractability a few computational problems:

1. *Computational Diffie-Hellman Problem (CDHP):* For $a, b, c \in \mathbb{F}_p^*$ given $P, aP, bP \in \mathbb{G}$ compute $abP \in \mathbb{G}$, for $\mathbb{G}$ of order p.

2. *Inverse Computational Diffie-Hellman Problem (Inv-CDHP):* For $a \in \mathbb{F}_p^*$, given $P, a^{-1}P \in \mathbb{G}$, compute aP, for $\mathbb{G}$ of order p. The Inv-CDHP is polynomial-time equivalent to CDHP [22].

[6]In adaptive attacks, an attacker can adapt his/her queries according to the previous ones [18].

3. *Bilinear Diffie-Hellman Problem (BDHP)* For $a, b, c \in \mathbb{F}_p^*$ given $P, aP, bP, cP \in \mathbb{G}$ compute $\hat{e}(P,P)^{abc}$, for $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ an admissible bilinear map as described above and $\mathbb{G}$ and $\mathbb{G}_T$ of order p.

4. *Decisional Bilinear Diffie-Hellman Problem (DBDHP)* For $a, b, c \in \mathbb{F}_p^*$, given $P, aP, bP, cP, z \in \mathbb{G}$ decide whether $\hat{e}(P,P)^{abc} = z$ or not.

The proof of security is a reductionist security proof in the random oracle model [18] and it proves that it satisfies chosen-ciphertext security and existential unforgeability. Essentially, it considers the existence of random instances of DBDHP and Inv-CDHP problems (i.e., considers random (P, aP, cP, cP, z) and $(P, a^{-1}P)$, respectively) and simulates the hash functions (H_1, H_2 and H_3) as random oracles.

We must also emphasise that the proposed CBS scheme satisfies public verifiability, since signature verification only relies on public parameters. Recall that the sender generates a signature S of the encrypted message and signature verification only requires the ciphertext $(c, R, S, \mathtt{ID}_A)$, the master public key B, and the topic public key Q_A. The hash value h can be obtained from H_3 on input $(c||\mathtt{ID}_A, R)$, since H_3 is publicly known.

6. PERFORMANCE AND SYSTEM EVALUATION

Performance-wise this scheme is comparable to the Boneh-Lynn-Shacham (BLS) [4] signature scheme for signature verification. As in the BLS scheme our scheme only requires the computation of one pairing (the most expensive operation) to validate a signature. The signcryption algorithm requires 1 exponentiation, 1 inversion, and 2 multiplications, and the unsigncryption requires 2 pairings and 1 multiplication, since the pairings $\hat{e}(Q_A, T)$ and $\hat{e}(B, Q_A)$ can be precomputed and stored until the topic public key expires. As pointed out in [3], the performance of the encryption algorithm is comparable to the performance of ElGamal encryption in $\mathbb{F}_p^*$. On the other hand, unsigncryption is the most expensive part of the protocol.

Revocation of public keys can be done as suggested in [3]. Essentially, the validity of a public key is attached to its identifier, i.e., the topic public key is the topic-identifier concatenated with expiration date (e.g., the topic-public key is *topic-name* || *date*). Every time a public key expires a new secret key must be generated and distributed. In this particular case, key update (if the master secret keys were not compromised) only requires two multiplications. The advantage over a PKI is that there is no need to get new certificates every time a key is revoked, since the public key can be locally computed.

An interesting feature of the proposed protocol is that it can be used to enforce end-to-end security at different smart grid domains with dedicated private-key generators (i.e. C-DAX security servers) issuing keys to local clients. For instance, if each distributed network operator (DNO) has its own PKG, they can still communicate using the same C-DAX cloud, as long as the public parameters used to validate topic-data authenticity in the cloud are distributed in advance. In this case, the DNO has the freedom to specify local access control policies for each client and distribute the topic-keys accordingly. Permissions to publish or subscribe in the C-DAX cloud are then defined locally by the DNO and these are unknown to the cloud (i.e., the cloud is not aware of the access control policies that it is actually enforcing).

Another kind of fine-grained access control mechanism can be supported in the proposed CBS scheme by attaching attributes to the topic-public keys, i.e., the topic-public key is not just a name, but the name and a set of attributes. Clients allowed to publish/subscribe information about such topics have to hold the associated secret keys. Notice that in this case, even if all the clients are able to encrypt topic-data (since it is done using the topic public-key), only the ones holding the publishing key (which provides a valid signature of the encrypted data) can actually publish it in the C-DAX cloud. The same happens when subscribing: only the subscribers holding a valid subscribing key can decrypt topic-data encrypted under a set of attributes.

7. RELATED WORK

Zhang et. al [23] were the first to adapt the identity-based cryptography schemes to information-centric networks. Their model also considers the unique identifier as being the content-name (topic-name) instead of the entity name. For scalability reasons, the authors propose a protocol that combines PKI and IBE. The PKI is used to guarantee the authenticity of the public parameters of different domains (e.g., public key generators of different organisations): the publisher sends the public parameters signed with its private key and the subscriber can verify the authenticity by using the corresponding public key certificate. The IBE and IBS are used for end-to-end security. The authors implemented the proposed scheme on a location-based application for Android devices. They concluded that to encrypt topic-data the performance of pairing-based protocols is roughly comparable to using RSA.

The approach of Zhang et. al is more general than ours since it aims at giving an IBE solution for multi-purpose information-centric networks, while the scheme proposed targets specific needs of the C-DAX project. Recall that the C-DAX architecture considers the existence of nodes that can store, cache or replicate topic-data; the role of the cloud is not only forwarding data from publishers to subscribers, but the C-DAX cloud is also able to perform content-authentication. Besides, unlike Zhang et. al, in the proposed paper, key and public parameters distribution are seen as orthogonal problems and can be done in many different ways. For example, security parameters can be distributed offline (say when equipment is rolled out in the field) or even as it is suggested in [23], depending on the particular setting in which C-DAX is used.

In order to provide some kind of content integrity, Smetters et. al [20] proposed a model based on self-certifying names which essentially ties the security of the content to the trust in the host. The publisher chooses a *user-friendly* name (for the content) and ties it to the content through a digital signature. A disadvantage of this approach is that subscribers have to hold all the certificates of all publishers whose messages they want to be able to validate.

Kim et. al [8, 9] have proposed two protocols (in two independent papers) for securing information-centric networks similar to the one envisioned in C-DAX, which we partially discussed in Section 2.3. The SSTP protocol [8] aims at enforcing confidentiality and integrity and it is based on pre-shared symmetric keys and state-tokens. The pre-shared key is used to establish a session key (using the *Diffie-Hellman*

key exchange protocol) between two clients. The state-token is used to avoid storage of the session state (including the session key). This protocol provides lower security guarantees, as pointed out before at Section 2.3. Additionally, a limitation is that it does not provide end-to-end security between publishers and subscribers: the cloud, to be able to authenticate topic-data, needs to know the pre-shared key, and can then spoof the messages.

The REMP protocol [9] is an end-to-end protocol to provide confidentiality and integrity and relies on diversified symmetric keys. Subscribers hold a (symmetric) master key associated to the topic and publishers hold a key derived from that master key and the publisher identity. Upon receiving the topic-data, each subscriber has to derive such key to be able to verify the message authentication code of the message and decrypt the topic-data. Authentication in the cloud is similar to authentication in the Kerberos protocol and is based on access-tickets. Although REMP is more robust than SSTP, it also has some limitations: its main disadvantage is the fact that it is vulnerable to replay attacks and the mechanism to mitigate this problem – nodes checking for repeated messages – is not really scalable.

8. CONCLUSIONS

This paper proposes a content-based signcryption protocol for information-centric networks such as the one adopted in the C-DAX project. Although it is a theoretical protocol (i.e. not yet implemented and tested in the field), it presents several advantages when compared to earlier proposals [8, 9]. First of all, it enforces end-to-end integrity and confidentiality between the clients and provides a fine-grained access control at publishing/subscribing level: to each topic public key are associated two secret keys, one for publishing and another one for subscribing. Because the proposed scheme is publicly verifiable, the cloud can authenticate all the topic-data published by the clients, without knowing the secret keys and without any additional authentication tokens created by the publisher. The memory usage required to store the public parameters necessary for content-authentication is minimal and constant. The protocol targets the specific needs of the communication model adopted in the C-DAX project, but is not limited to this.

Directions for future work.

Although the solution proposed already tackles some of the problems that arise when developing protocols to secure information-centric networks, several issues still remain. For instance, it is not clear in the proposed protocol how to deal with re-keying when clients join or leave a topic, i.e., when a client leaves a topic and is no longer allowed to publish or read messages on it, or when a client joins a topic but is not allowed to read old messages. Recall that in the proposed protocol, all the topic keys are derived from the same master secret keys. A solution would be to generate master secret keys for each topic, to avoid replacing all the topic keys when a client joins or leaves a specific topic. Still, it is not clear if this solution scales and we leave it as future work. A desirable feature would be to have different secret keys per client for topic secret keys, i.e., different publishing and subscribing keys per client for each topic. This would avoid the need of re-keying all the clients (either publishers or subscribers) when only one client is compromised (or is leaving or joining a topic).

A drawback of the proposed CBS scheme and of IBE schemes in general is that the PKG controls the entire system, and thus is a single point of failure. It would be interesting to study how to overcome this problem. It might be possible to apply the concepts underlying certificateless signcryption [2] where the secret key escrow functionality is removed. Another important aspect that must be addressed in the future is to evaluate if the system scales when the number of topics grows linearly in time. This is going to be useful for some use cases, namely the retail market use case, where a dynamic creation of topics is expected.

The major disadvantage of the proposed CBS scheme is that it exhibits weak resilience against exposure of publishing keys to external parties. For example, if a publisher is compromised, anybody can be disguised as a legitimate publisher of a particular topic group. A possible solution to overcome this issue could be based on the concepts underlying Hierarchical Identity-based Encryption (HIBE) [7], by giving different publishing keys (derived from a single publishing master key) to each publisher within a topic group. For instance, the publishing key could be derived from the master publishing key and the unique identifier of each publisher. The problem of such approach is that the act of publishing cannot be *anonymous* anymore: to be able to verify the signature, every subscriber and the cloud itself have to know the publisher's identity. This means that publisher's identifiers have to be attached *in the clear* (i.e., unencrypted) to the topic-message. Investigating how to cope with this issue is left as future work.

We also plan to investigate possibilities for privacy-friendly aggregation or filtering of encrypted data. This is interesting in use cases that involve privacy-sensitive data (e.g., metering data of an individual household or data from electronic vehicles) and commercially sensitive data (e.g., price offers). Privacy-friendly solutions for aggregration have already been proposed for the smart grid [6, 13]. In the C-DAX architecture the nodes are an obvious location, with the required computing resources, for carrying out such aggregation or filtering.

8.1 Acknowledgments*

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7-ICT-2011-8 under grant agreement n° 318708 (C-DAX). The authors alone are responsible for the content of this paper.

9. REFERENCES

[1] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman. A survey of information-centric networking. *Communications Magazine*, 50(7):26–36, 2012.

[2] M. Barbosa and P. Farshim. Certificateless signcryption. In M. Abe and V. D. Gligor, editors, *ASIACCS*, pages 369–372. ACM, 2008.

[3] D. Boneh and M. K. Franklin. Identity-based encryption from the weil pairing. In J. Kilian, editor, *CRYPTO*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2001.

[4] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the weil pairing. *Journal of Cryptology*, 17(4):297–319, 2004.

[5] A. Carzaniga, M. Papalini, and A. L. Wolf. Content-based publish/subscribe networking and information-centric networking. In *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, ICN '11, pages 56–61, New York, NY, USA, 2011. ACM.

[6] Z. Erkin, J. R. Troncoso-Pastoriza, R. Lagendijk, and F. Perez-Gonzalez. Privacy-preserving data aggregation in smart metering systems: An overview. *Signal Processing Magazine*, 30(2):75–86, 2013.

[7] C. Gentry and A. Silverberg. Hierarchical id-based cryptography. In *Proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, ASIACRYPT '02, pages 548–566, London, UK, UK, 2002. Springer-Verlag.

[8] Y. Kim, V. Kolesnikov, H. Kim, and M. Thottan. SSTP: a scalable and secure transport protocol for smart grid data collection. In *Smart Grid Communications (SmartGridComm)*, pages 161–166. IEEE, 2011.

[9] Y. Kim, V. Kolesnikov, and M. Thottan. Resilient end-to-end message protection for large-scale cyber-physical system communications. In *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, pages 193–198, 2012.

[10] Y. Kim, J. Lee, G. Atkinson, H. Kim, and M. Thottan. SeDAX: A scalable, resilient, and secure platform for smart grid communications. *Selected Areas in Communications*, 30(6):1119–1136, 2012.

[11] Y. Kim, J. Lee, G. Atkinson, and M. Thottan. GridDataBus: Information-centric platform for scalable secure resilient phasor-data sharing. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, pages 115–120. IEEE, 2012.

[12] J. Kurose. Content-centric networking: technical perspective. *Communications of the ACM*, 55(1):116–116, 2012.

[13] K. Kursawe, G. Danezis, and M. Kohlweiss. Privacy-friendly aggregation for the smart-grid. In *PETS*, pages 175–191, 2011.

[14] J. Malone-Lee. Identity-based signcryption. *IACR Cryptology ePrint Archive*, 2002:98, 2002.

[15] M. Mampaey. Deliverable 2.1 - C-DAX Requirements - use case descriptions for domains 1, 2 and 3 and derived c-dax requirements, 2013. Available from `http://www.cdax.eu`.

[16] N. McCullagh and P. S. L. M. Barreto. Efficient and forward-secure identity-based signcryption. *IACR Cryptology ePrint Archive*, 2004:117, 2004.

[17] A. Menezes. An introduction to pairing-based cryptography. Lectures notes, 2005.

[18] D. Pointcheval. Provable security for public key schemes. In *Contemporary Cryptology*, Advanced Courses in Mathematics - CRM Barcelona, pages 133–190. Birkhauser Basel, 2005.

[19] A. Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and D. Chaum, editors, *CRYPTO*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer, 1984.

[20] D. K. Smetters and V. Jacobson. Securing network content. Technical report, PARC, October 2009.

[21] P. systems management and associated information exchange. Iso-iec 6235: Security, 2007. Available from `http://www.iec.ch/smartgrid/standards/`.

[22] F. Zhang, R. Safavi-Naini, and W. Susilo. An efficient signature scheme from bilinear pairings and its applications. In *Public Key Cryptography - PKC 2004*, volume 2947 of *Lecture Notes in Computer Science*, pages 277–290. Springer, 2004.

[23] X. Zhang, K. Chang, H. Xiong, Y. Wen, G. Shi, and G. Wang. Towards name-based trust and security for content-centric network. In *ICNP*, pages 1–6, 2011.

[24] Y. Zheng. Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption). In B. K. Jr., editor, *Advances in Cryptology - CRYPTO'97*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179, 1997.