CHIST-ERA

User empowerment for SEcurity and privacy in Internet of Things

Initial USEIT Architecture Definition

Deliverable number: D3.1

Version 1.0

Funded by the Future and Emerging Technologies (FET) CHIST-ERA programme of the European Union.

Project Acronym: USEIT
Project Full Title: User empowerment for SEcurity and privacy in Internet of Things
Call: 2015
Grant Number: 20CH21_167531
Project URL: http://useit.eu.org

Editor:	Antonio Skarmeta, Universidad de Murcia (UMU) – Spain
Deliverable nature:	Report
Dissemination level:	Public
Delivery Date:	2018-02-28
Authors:	Jan Camenisch, IBM Research – Zurich José Luis Hernández Ramos, University of Murcia Gregory Neven, IBM Research – Zurich Alexis Olivereau, CEA Nouha Oualha, CEA Antonio Skarmeta, University of Murcia
Peer review:	Jan Camenisch, IBM Research – Zurich

Abstract

This document provides an initial overview of the baseline architecture USEIT will be using as part of the platform for developing the new components envisioned in the proposal. The deliverable will first describe the network and security architecture for C-ITS as it is currently standardized and as the architecture being identified for smart object and IoT like *High Level Architecture* (HLA) by AIOTI based on a layered functional model (Network, IoT, Application).

The deliverable will also identify the main components that can be used to carry out each technology's functionality that USEIT will considered and also the platform considerations to be considered for deployment, either in the Cooperative Intelligent Transport Systems (C-ITS) and for the Smart Objects use cases

This document will be a living version with several updated over D3.3 and D3.6 in order r to integrate the research project developments and results, the architecture will be reviewed and developed through iterative cycles during the whole project duration, taking into account the outputs of other work packages as they become available.

USEIT wants to use the experience on previous results from projects, such as ABC4Trust, FutureID, Sociotal, Smartie, ITTSv6, and others, and focus on the identification of the security and privacy components needed to extend and support the objectives of USEIT vision.

Contents

List of Figures

Executive Summary

This document provides an initial overview of the baseline architecture USEIT will be using as part of the platform for developing the new components envisioned in the proposal. The deliverable will first describe the network and security architecture for C-ITS as it is currently standardized and as the architecture being identified for smart object and IoT like *High Level Architecture* (HLA) by AIOTI based on a layered functional model (Network, IoT, Application).

The deliverable will also identify the main components that can be used to carry out each technology's functionality that USEIT will considered and also the platform considerations to be considered for deployment, either in the Cooperative Intelligent Transport Systems (C-ITS) and for the Smart Objects use cases

This document will be a living version with several updated over D3.3 and D3.6 in order r to integrate the research project developments and results, the architecture will be reviewed and developed through iterative cycles during the whole project duration, taking into account the outputs of other work packages as they become available.

USEIT wants to use the experience on previous results from projects, such as ABC4Trust, FutureID, Sociotal, Smartie, ITTSv6, and others, and focus on the identification of the security and privacy components needed to extend and support the objectives of USEIT vision.

List of Acronyms

PKI	public-key infrastructure
CA	Certification Authority
PCA	Pseudonym Certification Authority

1 USEIT technical overview

1.1 Technologies for users' empowerment

1.1.1 MyData Model and citizen control

The use of personal data has become a world-wide mainstream business activity in the last years. Its value is such that according to The World Economic Forum, 'Personal data is becoming a new economic asset class, a valuable resource for the 21st century that will touch all aspect of society" [2]. As a matter of fact, in a survey provided by Accenture where nearly 600 business around the world were responding, 79 percent of them affirmed to collect data directly from individuals (thanks to the customer account, for instance), as well as from connected devices and third-party data suppliers.

By contrast, such businesses are usually accompanied with a control loss by the individuals which have little or no knowledge over how data about them and their activities is created or used. In fact, nowadays, individuals grant legal consent to organizations and software applications for the collection and use of their personal data by accepting the terms of the service they use without understanding them or even without reading them due to their length and complexity.

As in real life, in our digital life individuals should have legal rights and technical tools to manage personal data collected about them. This is an extension to the freedom of thought and expression that we all have as citizens.

On the other hand, the actual protection regulations prevent companies to create innovative services around personal data so they resort to ways to bypass them. In order to solve this situation MyData initiative has been proposed [3], encompassing a framework, principles and a model for a human-centric approach to empower individuals about their personal data management and processing.

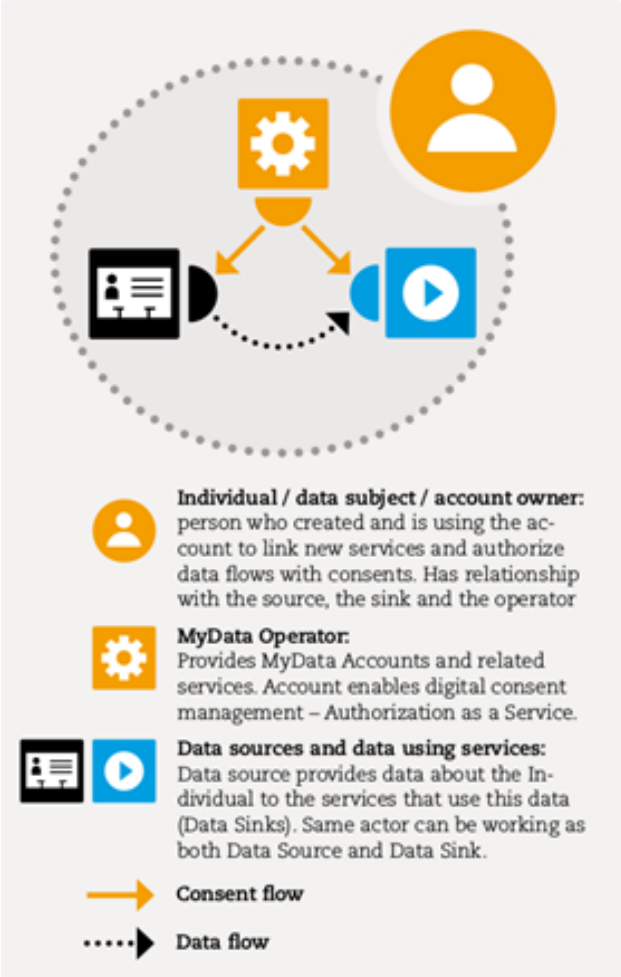
The fundamental principles on which the MyData initiative is based are as follows:

- Human centric control and privacy: Individuals are no longer passive targets, but empowered actors in the management of their personal lives both online and offline.
- Usable data: It is essential that personal data is technically easy to access and use – it is accessible in machine readable open formats via secure, standardized APIs (Application Programming Interfaces).
- Open business environment: A shared MyData infrastructure enables decentralized management of personal data, improves interoperability, makes it easier for companies to comply with tightening data protection regulations, and allows individuals to change service providers without proprietary data lock-ins.

MyData is a progressive approach to personal data management that combines digital human rights and industry need to have access to data. This approach benefits both sides: individuals and companies. For individuals, it provides easy-to-use and comprehensive tools for personal data management and transparency mechanisms that openly show how organizations use their data. For companies, it opens opportunities for new kinds of data-based businesses by facilitating the legal and technical access to pre-existing personal datasets when the individual is willing to give his/her consent. And in addition also for the civil society, it creates the necessary structures, processes and policies for protecting the rights of individuals and fostering the use of personal data in the development of innovative services.

The architecture proposed by the MyData initiative is based on interoperable and standardized MyData accounts. The proposed model allows individuals to control their personal data from a single place in an easy way. Such accounts will be provided by organizations that act as MyData operators, giving also the possibility to individuals to host their own accounts.

Figure 1.1: MyData Architecture



The data flows from a data source to a service or application that uses the data. It is worth mentioning that the flow of consents or permissions is separate from the actual flow of data as described in Figure 1.1. Actually, the primary function of a MyData account is to enable consent management – the data itself is not necessarily streamed through the servers where the MyData account is hosted.

1.1.2 XACML

For RBAC and ABAC models, the Policy-Based Access Control model (PBAC), enables the definition of access control rules in a policy-oriented way. PBAC, and consequently ABAC and RBAC, are usually deployed by using the XACML. XACML is a standard, declarative and XML-based language to express access control policies, which allows specifying the set of subjects which can perform certain actions on a specific set of resources, based on attributes of them. Under the XACML data model, the definition of access control policies is mainly based on three elements: PolicySet, Policy and Rule. A PolicySet may contain other PolicySets and Policies, whereas a Policy includes a set of Rules, specifying an Effect (Permit or Deny), as a result of applying that Rule for a particular request. The Target sections of these elements define the set of attributes from resources, subjects, actions and environment to which the PolicySet, Policy or Rule are applicable. Moreover, since different Rules might be applicable under a specific request, XACML defines Combining Algorithms in order to reconcile multiple decisions. In addition, a set of obligations (Obligations class) can be used to notify a set of actions to be performed related to an authorization decision. Figure 1 shows the XACML Policy

Language Model.

XACML architecture consists mainly of four elements:

- PEP (Policy Enforcement Point): it is responsible for performing access control, by making decision requests and enforcing authorization decisions.
- PDP (Policy Decision Point): it evaluates applicable policies and makes authorization decisions
- PAP (Policy Administration Point): it is used to create a policy or set of policies.
- PIP (Policy Information Point): it acts as a source of attribute values

1.2 USEIT crytographic approaches

1.2.1 CP-ABE

Attribute-Based Encryption (ABE) [3] represents the generalization of IBE, in which the identity of the participants is not represented by a single string, but by a set of attributes related to their identity. In both schemes, cryptographic keys are managed by a Trusted Third Party (TTP), usually called Attribute Authority (AA). ABE is gaining attention because of its high level of flexibility and expressiveness, compared to previous schemes. In ABE, a piece of information can be made accessible to a set of entities whose real, probably unknown identity, is based on a certain set of attributes. Based on ABE, two alternative approaches were proposed. In the Key-Policy Attribute-Based Encryption (KP-ABE) scheme [4], a ciphertext is encrypted under a set or list of attributes, while private keys of participants are associated with combinations or policies of attributes. In this case, a data producer has limited control over which entities can decrypt the content, being forced to rely on the AA entity issues appropriate keys for getting access to disseminated information. In contrast, in a CP-ABE scheme [5], a ciphertext is encrypted under a policy of attributes, while keys of participants are associated with sets of attributes. Thus, CP-ABE could be seen as a more intuitive way to apply the concepts of ABE; on the one hand, a producer can exert greater control over how the information is disseminated to other entities, On the other hand, a user's identity is intuitively reflected by a certain private key.

1.2.2 Privacy-ABCs

Privacy-preserving attribute-based credentials or Privacy-ABCs [6], sometimes also referred to as anonymous credentials [7], are a cryptographic primitive allowing data-minimizing authentication. The full feature set is beyond the scope of this paper, we restrict ourselves to the relevant concepts for our use here.

A Privacy-ABC credential is a list of attribute-value pairs certified by the issuer of the credential. For example, a credential issued to a vehicle could contain the vehicle class, make, model, or identification number. To authenticate to a verifier, the credential owner derives a so-called *presentation token* from the credential. Optionally, the presentation token can disclose a subset of the certified attributes and can be used to sign a message. Presentation tokens are unlinkable and untraceable, meaning that a verifier cannot tell whether two different presentation tokens were derived from the same or from different credentials, and that even the issuer cannot trace a presentation back to the issuance of the credential.

Modern Privacy-ABC schemes are only slightly less efficient than standard signature schemes. Creating and verifying a simple presentation token that reveals a subset of attribute values in the Identity Mixer scheme [7], for example, involves a single RSA multi-exponentiation.

Among the many other features of Privacy-ABCs, we mention three that are potentially useful in C-ITS scenarios. The first is that of pseudonyms, and in particular, scope-exclusive pseudonyms. Users in a Privacy-ABC system can optionally have a user secret key to which its credentials can be bound, in the sense that knowledge of the user secret key is required to create a presentation token. The user secret key can also be used to derive pseudonyms, which behave like public keys corresponding to the user secret key, in the sense that a verifier can check that a user knows the secret key corresponding to a given pseudonym. Unlike normal public keys,

however, one user secret key can be used to derive arbitrarily many unlinkable pseudonyms, in the sense that a verifier cannot tell whether two pseudonyms originated from the same user secret key or from two different ones. A Privacy-ABC presentation token can include a pseudonym and prove that the credential is bound to the user secret key from which the pseudonym was derived.

A particular type of pseudonyms are *scope-exclusive pseudonyms*, sometimes also known as domain pseudonyms, which are uniquely determined by the user secret key and a *scope string*. Meaning, for the same value of the scope string, there is only one scope-exclusive pseudonym that a user can generate, but a user's scope-exclusive pseudonyms cannot be linked across pseudonyms.

A second feature is that of *inspection*, whereby an attribute of the credential can be verifiably encrypted under the public key of a third party, called the inspector. The verifier can check that the correct attribute value was encrypted, but only the inspector can recover the actual value. When the encrypted attribute is linked to the user's identity, then this feature can be used to de-anonymize presentation tokens.

The last feature that we mention is (issuer-driven) *revocation*. Credential revocation in Privacy-ABCs is considerably more difficult than for regular certificates, as the unlinkability of presentation tokens prevents them from being checked against a revocation list. Still, there are a number of techniques [8] that can be used to revoke Privacy-ABC credentials, usually by letting users prove as part of the presentation token that their credential has not been revoked.

2 Initial USEIT Architecture and Platform instantiation

2.1 Architecture for C-ITS

We first describe the network and security architecture for C-ITS as it is currently standardized [9, 10], and then explain how the USEIT architecture fits into it. Mainly, the USEIT architecture follows the main design principles, but requires much less interaction with the pseudonym authority for a higher level of privacy.

2.1.1 Current C-ITS Reference Architecture

The C-ITS station reference architecture is standardized by ISO [11] and ETSI [12] describes an OSI-like network stack of C-ITS stations, which include vehicle ITS stations, roadside ITS stations, central ITS stations (i.e., centrally managed services on the network), and personal ITS stations (e.g., built into a smartphone or a dedicated handheld device). The network stack is the same for all these type of stations and also makes abstraction of the underlying physical communication layer, which could be cellular (2G, 3G), microwave (5 GHz IEEE 802.11p vehicular WiFi, 2.5 GHz IEEE 802.11n urban WiFi), satellite, infrared, 60 GHz millimeter-wave and possibly others.

The detailed network reference architecture is depicted in Figure 2.1. It is a layered architecture, much like the well-known OSI network architecture, but has cross-cutting vertical entities for management and, importantly, security. The security entity provides common security functionalities to all horizontal layers such as managing security credentials and cryptographic keys, managing firewalls and intrusion detection, and authentication and authorization. Security credentials such as cryptographic keys, authorization tickets, and certificates are stored and maintained in a dedicated secure environment called Hardware Security Module in Figure 2.1. For vehicle stations, this should not be thought of as a full server-grade HSM, but rather as a dedicated ASIC that may or may not have additional hardware protection against tampering. The atomic security operations offered by the security entity are random number generation, hashing, signing, verification, encryption, and decryption.

The V2X security architecture defined by the EU project PRESERVE [13] was adopted by ETSI and the Car2Car consortium and is depicted in Fig. 2.2. The architecture is a based on digital signatures in a public-key infrastructure (PKI). Each vehicle and each piece of roadside infrastructure is equipped with a unique signing key and a corresponding long-term certificate (LTC). The LTC is certified by the long-term Certification Authority (CA), whose public key on its turn is certified by a root CA. There can be several root CAs and long-term CAs in the system.

For privacy reasons, the messages sent out by vehicles are not signed directly with the vehicle's long-term key, but with short-lived pseudonyms. Each vehicle periodically generates a bundle of fresh signing keys and sends the public keys to a Pseudonym Certification Authority (PCA), e.g., via 3G communication. After letting the long-term CA check that the vehicle authenticated correctly using its (encrypted) LTC and that the vehicle's LTC is not revoked, the PCA issues pseudonym certificates (PC) for all signing keys. PCs expire after a certain time, but cannot be revoked. The recipient of a message verifies that the message is correctly signed under the PC and that the PC is signed by the PCA. The latter test can be skipped for subsequent messages that reuse the same PC.

Vehicles must change pseudonyms frequently to maximize their privacy, but how frequently is a matter of debate. The PRESERVE architecture suggests to change *every few minutes* [13]. Practical considerations on the availability and the bandwidth of network connectivity to the PCA, as well as storage space on the vehicle would impose lower change rates or reusing old pseudonyms. Research by Wiedersheim et al. [14] suggests, however, that vehicles can be tracked with almost perfect accuracy if they broadcast their location once per

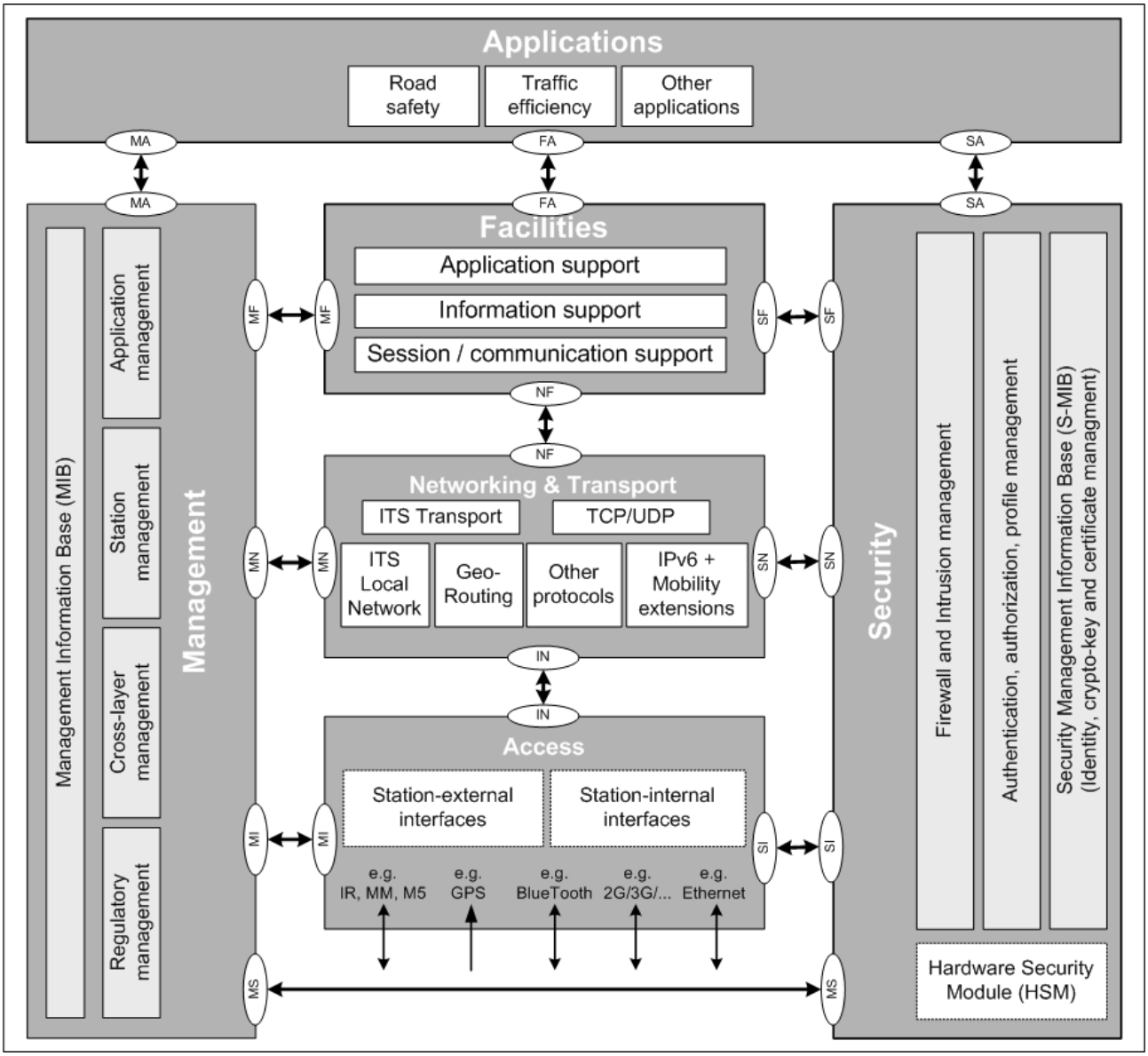


Figure 2.1: ITS Station network stack as specified in the reference architecture. (Source: ITSSv6 project [1].)

second but change pseudonyms less than once per ten seconds. A rather uncomfortable trade-off must therefore be defined between efficiency and privacy: more pseudonyms mean better privacy for vehicles, but also mean more computation and bandwidth load on the PCA as well as more secure key storage in the vehicle.

The PCA is not only a bottleneck in terms of efficiency, as just described, but also in terms of security. The PCA must be highly available to provide vehicles with new PCs when they need them, but at the same time must be highly secure because a compromise of its issuance key causes the entire system to collapse. These are difficult and very expensive requirements to meet; it is currently not clear who will take up the role of PCA and how its operating costs are to be financed.

2.1.2 The USEIT C-ITS Architecture

In Figure 2.3, we depict the USEIT C-ITS security architecture that uses Privacy-ABCs to eliminate many of the above-mentioned drawbacks of the PRESERVE architecture. A vehicle's long-term credential is a standard public-key certificate, as in the original PRESERVE architecture. At the beginning of each period (e.g., each

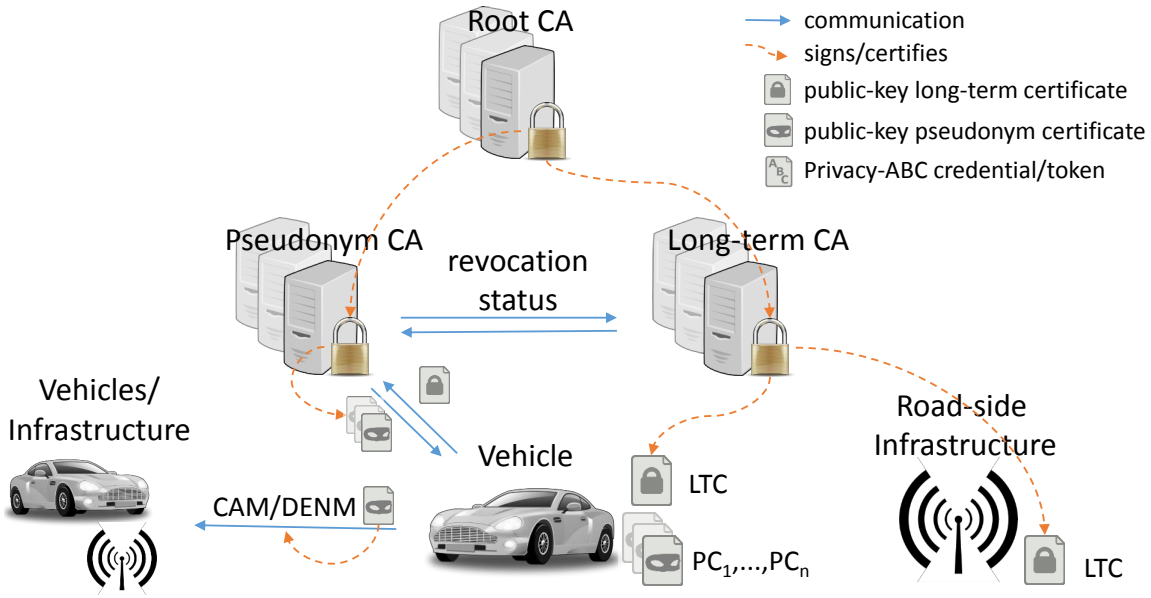


Figure 2.2: The PRESERVE security architecture for C-ITS.

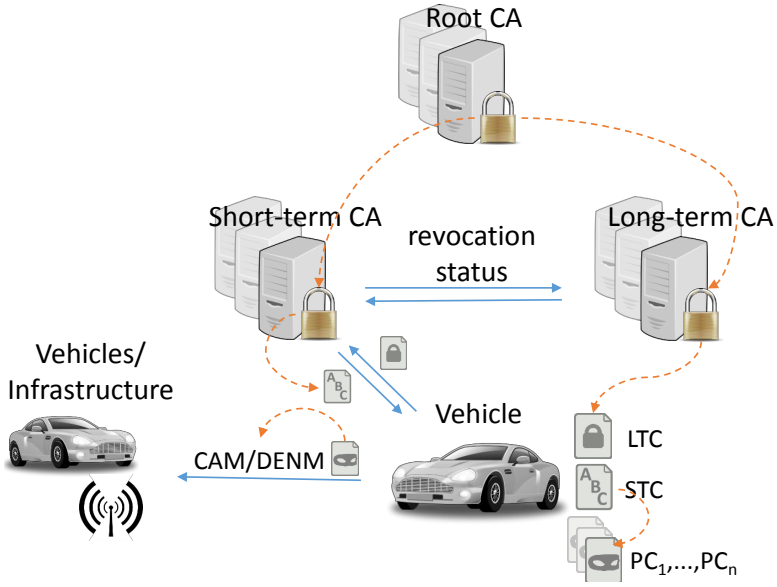


Figure 2.3: Short-term Privacy-ABCs. See Figure 2.2 for a legend of symbols.

month), the vehicle authenticates to a short-term CA using its long-term certificate. The short-term CA checks the revocation status of the LTC and issues a short-term credential (STC) to the vehicle, which is a Privacy-ABC credential with a single attribute that identifies the current period (e.g., the current month) as well as any other scoping limitations, e.g., geographic restrictions.

For each new pseudonym, the vehicle generates a standard signature key pair as before, but can "sign" its own pseudonym certificate by using the STC to create a presentation token that reveals the period identifier and signs the pseudonym public key. The vehicle signs outgoing CAM/DENM messages as before using the pseudonym's secret key, but uses the presentation token as part of the PC. The receiving ITS station verifies the pseudonym signature in the traditional way, checks the PC by verifying the included presentation token using the public key of the short-term CA, and checks that the revealed month is the current month.

We use the same approach as Singh and Fhom [15] to protect against Sybil attacks using scope-exclusive

pseudonyms. Namely, we ensure that each vehicle can only generate one valid pseudonym at any given time by using the current time slot as a scope string. Recall that each vehicle can only generate one unique scope-exclusive pseudonym for a given scope string, but that pseudonyms of the same vehicle across different scope strings are unlinkable. This feature can be used to flexibly derive new pseudonyms with a minimal impact on efficiency, while providing perfect resistance against Sybil attacks. This is a very powerful mechanism indeed when compared to the PRESERVE architecture, where the pseudonym pool size determines the exposure to Sybil attacks and limits the frequency of pseudonym changes. For example, if vehicles should switch pseudonyms every minute, then the current date and time (in minutes) can be used as a scope string. Thereby, each vehicle has a unique scope-exclusive pseudonym for each minute of the day, but its pseudonyms at different times remain unlinkable. If CAM and DENM messsages by the same vehicle should be unlinkable, then one can use different scope strings for both. Scope-exclusive pseudonyms can also be used for other purposes, e.g., to enforce linkability of vehicles around an intersection, then a canonical identifier of the intersection can be used as a scope string, possibly concatenated with the date or time of day to ensure unlinkability between different passings of the same vehicle.

In addition, the Privacy-ABC approach validates the unlinkability and untraceability requirements because even the short-term CA cannot track which vehicle signed a particular message, or tell whether two messages were signed by the same vehicle or by different vehicles. In case a security breach occurs on the network-exposed part of the short-term CA, then at most the updates for the current period are compromised, without affecting security for future periods. This supports the validation of the data retention requirement because of the limited time validity of the certificates.

Note that, unlike [15], we suggest to use the STC to certify public-key pseudonyms, and sign the CAM/DENM messages under this public key, rather than signing CAM/DENM messages with Privacy-ABCs directly. This has the advantage that the less efficient Privacy-ABC signing and verification algorithm only needs to be invoked the first time that a new pseudonym is generated, respectively encountered, instead of for every incoming message, thereby incurring the computational overhead of using Privacy-ABCs only for a tiny fraction of the messages.

Identifying misbehaving vehicles can be performed by using the inspection feature of Privacy-ABCs, possibly with the inspector's secret key secret-shared over multiple authorities to avoid that a single authority can break users' privacy. The verifiable ciphertext, however, can considerably increase the size of a presentation token. Alternatively, at initialization, vehicles could secret-share their user secret key over two or more authorities. As the scope-exclusive pseudonym is uniquely defined by the user secret key and the scope string, the authorities can recover the sender of a malicious message by engaging in a joint cryptographic protocol to determine which key was used to generate this pseudonym (typically involving one exponentiation per registered vehicle per authority).

To avoid the issuance key of the short-term CA to be exposed to online attacks, one can employ the credential update technique of Camenisch et al. [16, 15] instead of performing a full issuance protocol at each period. This allows the short-term CA to pre-compute per-credential update values offline, e.g., before the beginning of each time period, to store them on the online server, and to transmit them to correctly authenticated vehicles. The update values can only be used to update an existing credential; they are useless without an existing credential and in particular cannot be used to issue arbitrary new credentials.

To minimize the required interaction during credential updates even further, one could also use Verheul's "issue first, activate later" technique [17] and preload vehicles with encrypted STC credentials for many years. Vehicles receive the decryption key for each credential in a short message when the previous time period nears its end, e.g., at the end of every month. If a vehicle is revoked, it simply doesn't receive any further decryption keys.

2.2 USEIT Architectural approach

2.2.1 Architecting the IoT

In recent years, the constant evolution of IoT has produced a wide range of technologies and protocols, resulting in a fragmented landscape of solutions. Therefore, there is a real need to provide high-level architectures able to disengage from the technical details, in order to provide a common understanding of the security and privacy needs in IoT scenarios. Towards this end, in 2015 the AIOTI WG03 initiated the development of a *High Level Architecture* (HLA) for IoT by following a layered functional model (Network, IoT, Application). Furthermore, this working group is in close cooperation with AIOTI WG04, which is addressing policies issues related to security and privacy. In addition to AIOTI, currently there are other initiatives mainly focused on the definition of a high-level architecture for IoT. In particular, the purpose of the IEEE "*Standard for an Architectural Framework for the Internet of Things (IoT)*" (IEEE P2413)[1] is to define an architectural framework, addressing descriptions, definitions and common aspects in different domains IoT, in order to increase compatibility, interoperability and transparency of IoT systems. The proposed architecture is based on a three-tier approach (Sensing, Networking and Data Communications, and Applications). Moreover, the oneM2M initiative represents a joint effort with 14 partners (the *European Telecommunications Standards Institute* (ETSI) among others) in order to ensure efficient M2M deployments through the use of IoT. oneM2M provides a layered model (Network Services, Common Services and Application) that is mapped to a functional architecture composed of three entities with the same name. Furthermore, the *ITU Telecommunication Standardization Sector* (ITU-T) under the recommendation Y.2060 "*Overview of the Internet of Things*" has designed a reference model based on four levels (Device, Network, Service Support and Application Support, and Application) and two cross-layer levels (Security capabilities and Management capabilities), in order to group different functional aspects in each layer. In addition, the ITU-T Study Group 20 (SG20) "*Internet of Things (IoT) and its applications including smart cities and communities (SC&C)*"[2] is intended to develop standards to enable coordinated development of IoT technologies, and mechanisms for the interoperability of IoT applications and datasets employed by various vertically oriented industry sectors.

Moreover, in the scope of European research projects, the huge range of IoT application scenarios of IoT has led to the specification of different architectures that are usually tailored to be deployed on specific domains or addressing particular requirements. This was already identified as a significant barrier for IoT adoption on a broad scale and the main incentive for the development of coordinated efforts driven by the *Internet of Things European Research Cluster* (IERC). One of the first proposals to address this need of a common and harmonized IoT architecture was IoT-i [3], a European research project that dealt with the analysis of different architectures in order to create a joint and aligned vision of the IoT in Europe. This effort meant a step forward for the creation of a holistic environment that encourages a broader adoption of IoT. IoT-A[4] was a large-scale project focused on the design of an *Architecture Reference Model* (ARM) to be additionally instantiated by other IoT architectures through a set of specific tools and guidelines. Moreover, the focus of the architecture proposed by IoT6 [18] was to use the results of previous projects to design an IPv6-based service-oriented architecture, in order to achieve a high degree of interoperability among different applications and communication technologies. Additional architectures were proposed by other remarkable efforts at European level, such as BUTLER[5], SENSEI[6] or FI-WARE[7] based on the specific set of requirements from particular application domains. On the one hand, SENSEI focused on designing the service layer in wireless sensor and actuators networks. On the other hand, FI-WARE, under the FI-PPP program, designed an open platform based on an

[1] https://standards.ieee.org/develop/project/2413.html
[2] http://www.itu.int/en/ITU-T/about/groups/Pages/sg20.aspx
[3] http://postscapes.com/iot-i-iot-initiative
[4] http://www.iot-a.eu/public
[5] http://www.iot-butler.eu/
[6] http://www.sensei-project.eu/
[7] https://www.fiware.org/

architecture composed by components, which are referred as *Generic Enablers* (GEs).

This set of EU projects addresses the definition of an IoT architecture considering different levels of abstraction to fit in specific scenarios. Furthermore, these initiatives do not address security and privacy concerns from a holistic point of view. On the contrary, USEIT's architectural efforts will be mainly driven through the instantiation of the architecture proposed by IoT-A. The main motivation for choosing the ARM as a starting point is due to the fact that it provides a comprehensive definition of the IoT ecosystem, by proposing different models and architectures. In addition, IoT-A results are strongly supported by emerging initiatives, such as the IEEE P2413 or the initial definition of HLA provided by AIOTI WG03, for the specification of a reference architecture for IoT.

2.2.2 IoT-A and the Architecture Reference Model

IoT-A was a large-scale European project focused on the design of an *Architectural Reference Model* (ARM) [19], in order to enhance the interoperability among isolated IoT domains as a key step to move from an *Intra*net of Things to a real *Inter*net of Things [20]. The set of results derived from IoT-A embrace: a *Reference Model* (RM) to promote common understanding at high abstraction level; a *Reference Architecture* (RA) to describe essential building blocks and build compliant IoT architectures; and a set of Best Practices/Guidelines to help in developing an architecture based on the RA. In particular, the RA provides several views and perspectives focused on different architectural aspects. Among these views, the Functional View (shown in Figure 2.4) describes a set of *Functional Components* (FC), which are organized into nine *Functional Groups* (FG), as well as their responsibilities and interfaces [IoT-A. D1.5].

- Application FG. It represents users and services that interact with IoT systems.
- Device FG. It represents sensor, actuator or tags in an IoT domain.
- IoT Process Management FG. Its aim is to provide the functional concepts and interfaces necessary to augment traditional (business) processes with the idiosyncrasies of the IoT world.
- Service Organisation FG. It is intended to compose and orchestrate Services of different levels of abstraction.
- Virtual Entity FG. It contains functions for interacting with the IoT System on the basis of Virtual Entities (VEs) (i.e. the digital representation of a physical entity), as well as functionalities for discovering and looking up services that can provide information about VEs, or which allow the interaction with VEs.
- IoT Service FG. It contains IoT services as well as functionalities for discovery, look-up, and name resolution of IoT Services.
- Communication FG. It is an abstraction, modelling the variety of interaction schemes derived from the many technologies belonging to IoT systems and providing a common interface to the IoT Service FG.
- Security FG. It is responsible for ensuring the security and privacy of IoT-A-compliant systems

In addition, the Security FG is composed of five functional components: *Authentication*, *Authorization*, *Identity Management* (IdM), *Key Exchange and Management* (KEM) and *Trust and Reputation* (T&R).

- Authentication FC. It is mainly involved in user and service authentication by checking users' credentials and verifying assertions.
- Identity Management FC. It addresses privacy aspects by managing pseudonyms to enable anonymous interactions.
- Authorization FC. It is responsible for granting/denying actions based on access control policies, as well as for managing these policies.
- Key Exchange and Management FC. It is in charge of enabling secure communications among different entities by distributing cryptographic material and registering security capabilities.
- Trust and Reputation FC. It collects reputation scores and calculates service trust levels by requesting and providing reputation information

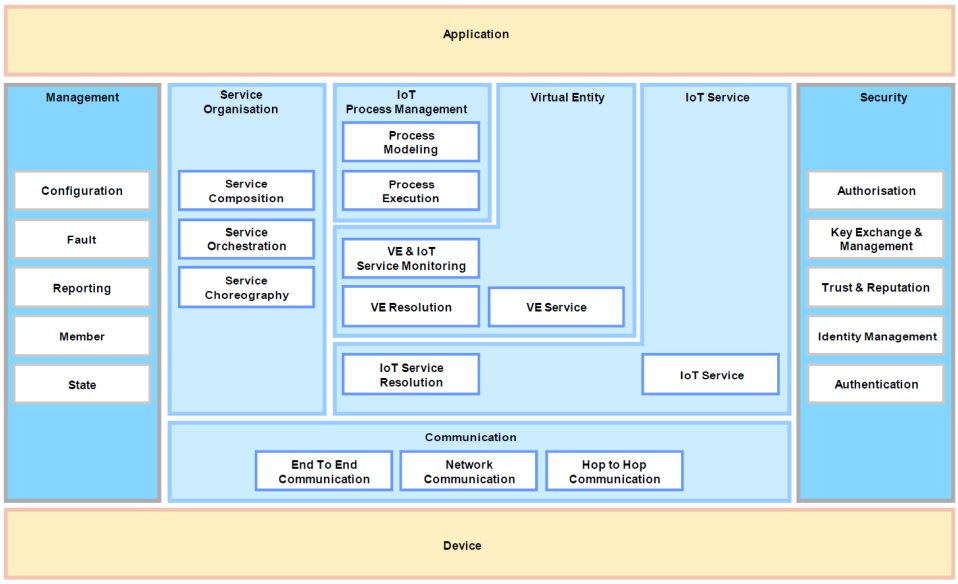


Figure 2.4: IoT-A Functional View

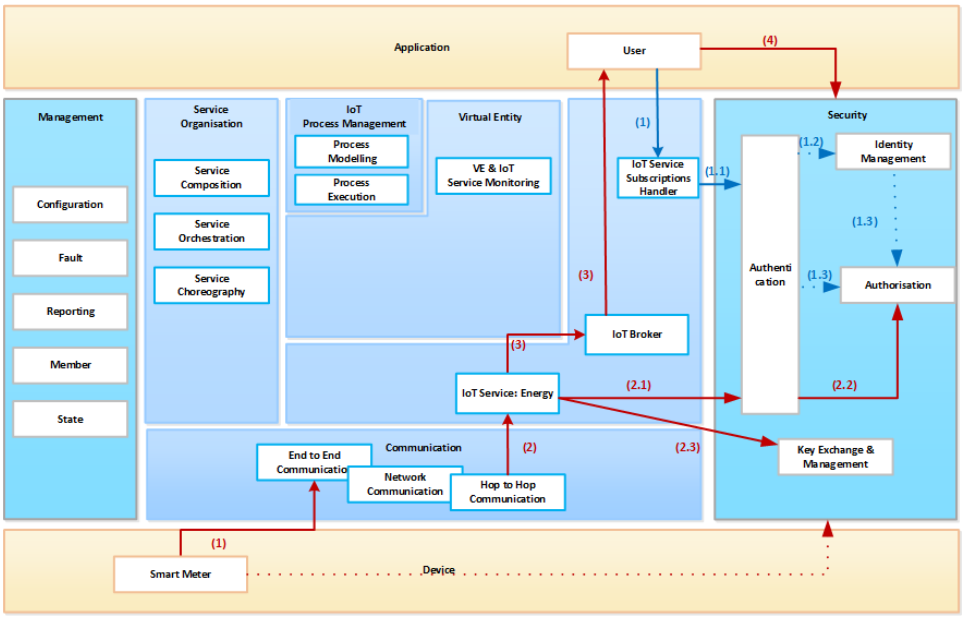


Figure 2.5: Information View for the USEIT Smart Building use case

2.2.3 USEIT Use case Information View

Subscription

- (1) The user sends a subscription request to the IoT Service Subscriptions Handler, so she can be notified when a new energy measurement is generated.
- (1.1) The IoT Service Subscriptions Handler authenticates the user. For this authentication, the user can employ traditional mechanisms, such as login/password or X.509 certificates, or
- (1.2) The user employs a cryptographic proof from an anonymous credential, so it cannot be unequivocally identified when suscribing
- (1.3) In any case, one she is authenticated, the attributes previously demonstrated are used to launch an

authorization process

Publish Data:

- (1) A device (e.g. a smart meter) generates a data measurement. The data is sent through the controller/gateway and through the network reaches The IoT Service. This communication should be done securely by using DTLS for instance. Furthermore, in case the device is powerful enough, it encrypts and signs the data before sending.
- (2) The raw data is sent to Push data Handler IoT service. This service authenticates and checks if the device is authorized to publish data in the IoT Broker. In particular:
- (2.1) For authentication purposes, the IoT Service can use PKC (e.g. by a signature) or by using MAC/HMAC in case of using SKC
- (2.2) For authorization purposes, the device can use of an authorization token that is validated by the IoT service
- (2.3) In case the data from the device is not protected, the IoT Service can make use of its capabilities to encrypt and potentially sign the data on behalf of the device before it is sent to the IoT Broker In case the device is authenticated and authorized, the IoT service converts the raw data into a formatted data (e.g. by using NGSI)
- (3) The IoT Service puts the updated data in the IoT Broker, which sends it to the subscribers.
- (4) Upon receiving the new measurement, the user makes use of the corresponding key to decrypt the received data, and validate the signature of it.

2.3 USEIT Platform

2.3.1 FI-WARE Introduction

FIWARE[8] is a European middleware platform that promotes the development and global deployment of applications for the Future Internet. FIWARE delivers a reference architecture, as well as the specification and implementation of different open interfaces called *Generic Enablers* (GEs). The FIWARE Catalogue contains a rich library of components (Generic Enablers) with reference implementations that allow developers to put into effect functionalities such as the connection to the Internet of Things or Big Data analysis

2.3.2 Main FI-WARE Components for Security and Privacy

To include a brief description of some of the main FI-WARE components that will be used within USEIT to enable security and privacy

- Orion Context Broker. The Orion Context Broker is an implementation of the Publish/Subscribe Context Broker GE, providing the NGSI9 and NGSI10 interfaces. Using these interfaces, clients can do several operations:
  - Register context producer applications, e.g. a temperature sensor within a room
  - Update context information, e.g. send updates of temperature
  - Being notified when changes on context information take place (e.g. the temperature has changed) or with a given frequency (e.g. get the temperature each minute)
  - Query context information. The Orion Context Broker stores context information updated from applications, so queries are resolved based on that information.
- Keyrock IdM. Keyrock IdM is an open source implementation of the IdM system defined in FIWARE. It relies on standard protocols, such as SAML and OAuth, to provide authentication and authorization

[8] https://catalogue.fiware.org

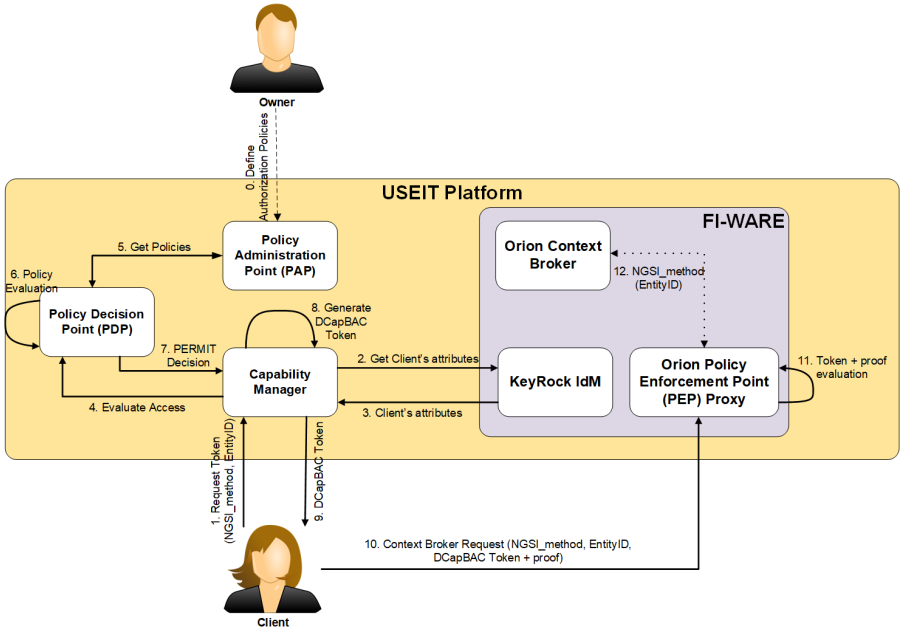
features, which allows to manage users' access to networks, services and applications. The IdM GE is also responsible for the user profile management, as well as SSO and identity federation across different service domains. Keyrock relies on the OpenStack IdM implementation called Keystone, and extends it by providing an implementation of the SCIM standard. SCIM is intended to reduce the cost and complexity of user management operations through a common user schema, extension model and REST API with a rich, but simple set of operations.

- Orion PEP Proxy. The Orion Policy Enforcement Point (PEP) is a proxy meant to secure independent FI-WARE components, by intercepting every request sent to the component, validating it against the Access Control component.

2.3.3 Access Control and CP-ABE Integration Example

Figure 2.6 shows an integration example related to access control functionality. In particular, we have made use of XACML for the implementation of the *Policy Administration Point* (PAP) and the *Policy Decision Point* (PDP) components, which have been deployed as web services. In addition, we have added the *Capability Manager* as the component for generating DCapBAC tokens in case of receiving affirmative authorization decisions from the PDP. These tokens are based on the proposal presented in [21].

Figure 2.6: Access Control Integration on FI-WARE



Before the description of the main required interactions, it should be noted that we consider two main external actors:

- The Owner is the user or service in charge of defining access control policies, in order to guarantee only authorized users will be able to provide information to the platform.
- The User represents a data consumer, which aims to access data from the platform.

In addition to these external entities, USEIT platform integrates different components, including FI-WARE GEs previously mentioned. In this way, the functionality of DCapBAC has been enabled through the definition of different components within the SMARTIE platform in order to automate the DCapBAC tokens generation process. In particular, we have made use of XACML for the implementation of the policy administration point (PAP) and the policy decision point (PDP) components, which have been deployed as web services. In addition,

we have added the Capability Manager as the component for generating DCapBAC tokens in case of receiving affirmative authorization decisions from the PDP.

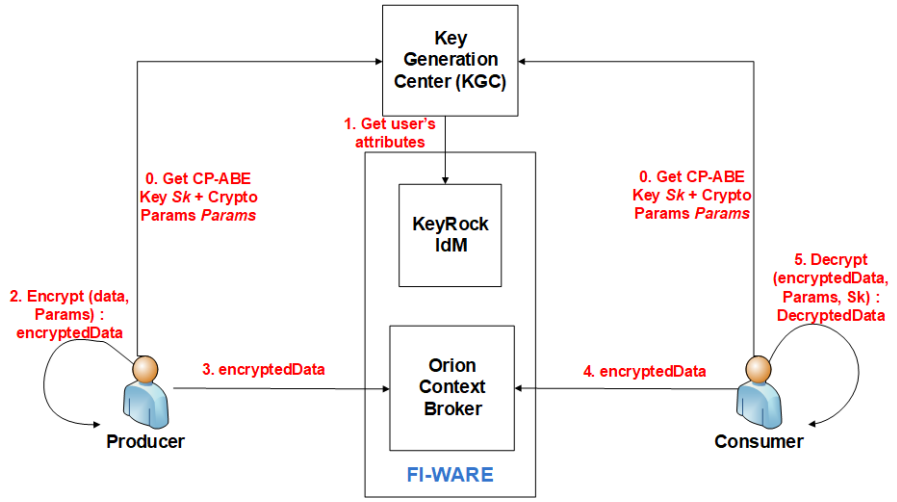
This way, when a user tries to get access data from the Orion Context Broker, it requests a token for this action, by querying the Capability Manager (step 1). Then, this component asks the KeyRock IdM to get the user's identity attributes (steps 2-3). Then, the Capability Manager asks the PDP (step 4) to determine whether the requested credential must be generated. The PDP uses the policies defined by the Owner in the PAP (step 5) and evaluates them against the user's request (step 6). In the case of an affirmative decision (step 7), the Capability Manager generates a token for the user (step 8) that is finally delivered (step 9). This token includes the user's public key, as well as a specific action (e.g. NGSI queryContext method) over a data hosted by the Orion Context Broker, as an access right. Additionally, it includes time restrictions, delimiting the validity period for this credential. Then, the user can use this token to access the data being requested (step 10). This token is evaluated by the PEP Proxy (step 11) and, in case of a successful evaluation, the request is forwarded to the Orion Context Broker (step 12).

2.3.4 CP-ABE integration

In the case of CP-ABE functionality, it can be realized by three main components. In particular, the CP-ABE Key Generation Center (KGC) is responsible for generating and distributing CP-ABE keys, which are employed by users (acting as consumers) to decrypt data from the platform. Furthermore, the KeyRock IdM and the Orion Context Broker are intended to provide the same functionality described in the previuous section

In this case, different users (data producers and consumers) get CP-ABE cryptographic material from the KGC (step 0). It should be noted that the KGC, upon receiving a request, gets the identity attributes associated to the user from the KeyRock IdM (step 1). That way, the CP-ABE private keys are associated to the set of identity attributes of each user. Furthermore, they receive the corresponding public parameters that are used for encryption and decryption processes.

Figure 2.7: CP-ABE integration on FI-WARE



Previous processes are only performed in case new cryptographic material is required (e.g. because expiration or revocation). Once users have received such material, they can share their encrypted data through the Orion Context Broker. Therefore, a data producer can update his information in this central entity (step 2-3) by using CP-ABE encryption under a specific policy or combination of identity attributes (e.g. people working at UMU). This way, confidentiality is ensured on an end-to-end basis, since the Orion Context Broker cannot get access to the shared data. At the same time the same data can be protected to be accessible to a set users; in particular, those satisfying the CP-ABE policy that was used to encrypt the data. Upon receiving the encrypted data, users

can use the CP-ABE decryption process by employing their provate keys to get access to disseminated data (step 4). In this case, it should be noted that even in dynamic conditions (e.g. frequent change of policy), additional key management tasks are not required.

3 Conclusions

Deliverable D3.1 has provided a review of the main architecture approaches related to the Cooperative Intelligent Transport Systems (C-ITS) and for the Smart Objects use cases. The references models of these architectures have been also instantiated over concrete platform solutions for C-ITS and another one for IoT based on FIWARE enablers.

This document will have a continuation on next version D3.3 and D3.6 that will provide updates main system components and their functionality, interaction patterns, interfaces, and the underlying communications links, providing a continuous review of architecture through iterative cycles during the whole project duration, taking into account the outputs of other work packages as they become available.

The objective at the end, it is to have systematic analysis and its technical implications to design an appropriate architecture that will support identified use cases while leveraging the existing C-ITS and IoT architectures to the furthest extent possible, reflecting the work to be carried out in task 3.1 and task 3.2.

Bibliography

[1] T. Ernst, M. Tsukada, J.-H. Lee, E. Thierry, F. Pereiguez, and A. F. Skarmeta, "D2.4: Final system specification," ITSSv6 Project, 2014.

[2] W. E. Forum, "Personal data: The emergence of a new asset class," https://www.weforum.org/reports/personal-data-emergence-new-asset-class, 2011.

[3] A. Sahai and B. R. Waters, "Fuzzy identity-based encryption," in *2005*, ser. LNCS, R. Cramer, Ed., vol. 3494, May 2005, pp. 457–473.

[4] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *ACM CCS 06*, A. Juels, R. N. Wright, and S. Vimercati, Eds. ACM Press, Oct. / Nov. 2006, pp. 89–98, available as Cryptology ePrint Archive Report 2006/309.

[5] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007*. IEEE Computer Society Press, May 2007, pp. 321–334.

[6] J. Camenisch, M. Dubovitskaya, A. Lehmann, G. Neven, C. Paquin, and F. Preiss, "Concepts and languages for privacy-preserving attribute-based authentication," in *Policies and Research in Identity Management - Third IFIP WG 11.6 Working Conference, IDMAN 2013*, ser. IFIP Advances in Information and Communication Technology, S. Fischer-Hübner, E. de Leeuw, and C. Mitchell, Eds., vol. 396. Springer, 2013, pp. 34–52.

[7] J. Camenisch and A. Lysyanskaya, "An efficient system for non-transferable anonymous credentials with optional anonymity revocation," in *Advances in Cryptology — EUROCRYPT 2001*, ser. Lecture Notes in Computer Science, B. Pfitzmann, Ed., vol. 2045. Springer, 2001, pp. 93–118.

[8] J. Lapon, M. Kohlweiss, B. D. Decker, and V. Naessens, "Analysis of revocation strategies for anonymous idemix credentials," in *Communications and Multimedia Security, CMS 2011*, ser. Lecture Notes in Computer Science, vol. 7025. Springer, 2011, pp. 3–17.

[9] T. ETSI, "102 637-2," *Intelligent Transport Systems (ITS)*, pp. 637–2, 2010.

[10] ETSI, "Intelligent transport systems (ITS); security; security header and certificate formats," TS 103 097 V1.2.1, 2015.

[11] ISO, *Intelligent transport systems – Communications Access for Land Mobiles (CALM) – Architecture*, ISO TC204 WG16 Std., April 2010, iSO 21217:2010(E).

[12] *Intelligent Transport Systems (ITS); Communications Architecture*, ETSI Std., September 2010, eTSI EN 302 665 V1.1.1 (2010-09).

[13] N. Bißmeyer, S. Mauthofer, J. Petit, M. Lange, M. Moser, D. Estor, M. Sall, M. Feiri, R. Moalla, M. Lagana, and F. Kargl, "PREparing SEcuRe VEhicle-to-X communication systems deliverable 1.3: V2X security architecture v2," 2014.

[14] B. Wiedersheim, M. Zhendong, F. Kargl, and P. Papadimitratos, "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough," in *Wireless On-demand Network Systems and Services (WONS) 2010*, 2010, pp. 176–183.

[15] A. Singh and H. S. Fhom, "Restricted usage of anonymous credentials in vehicular ad hoc networks for misbehavior detection," *Int. J. Inf. Sec.*, vol. 16, no. 2, pp. 195–211, 2017.

[16] J. Camenisch, M. Kohlweiss, and C. Soriente, "Solving revocation with efficient update of anonymous credentials," in *Security and Cryptography for Networks — SCN 2010*, ser. Lecture Notes in Computer Science, J. A. Garay and R. D. Prisco, Eds., vol. 6280. Springer, 2010, pp. 454–471.

[17] E. R. Verheul, "Activate later certificates for V2X – combining ITS efficiency with privacy," Cryptology ePrint Archive, Report 2016/1158, 2016.

[18] S. Ziegler, C. Crettaz, L. Ladid, S. Krco, B. Pokric, A. F. Skarmeta, A. Jara, W. Kastner, and M. Jung, "IoT6 – Moving to an IPv6-Based Future IoT," in *The Future Internet*. Springer Science Business Media, 2013, pp. 161–172. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-38082-2_14

[19] A. Bassi, M. Bauer, M. Fiedler, T. Kramp, R. van Kranenburg, S. Lange, and S. Meissner, *Enabling Things to Talk*. Springer Science Business Media, 2013. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-40403-0

[20] M. Zorzi, A. Gluhak, S. Lange, and A. Bassi, "From today's INTRAnet of things to a future INTERnet of things: a wireless-and mobility-related view," *Wireless Communications, IEEE*, vol. 17, no. 6, pp. 44–51, 2010.

[21] J. L. H. Ramos, M. P. Pawlowski, A. J. Jara, A. F. Skarmeta, and L. Ladid, "Toward a lightweight authentication and authorization framework for smart objects," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 4, pp. 690–702, 2015.