IET Information Security

Research Article

IET Journals
The Institution of Engineering and Technology

Impact assessment of policy expressiveness of an optimised access control model for smart sensors

ISSN 1751-8709
Received on 11th January 2017
Revised 9th May 2018
Accepted on 12th October 2018
E-First on 20th February 2019
doi: 10.1049/iet-ifs.2018.5204
www.ietdl.org

Mikel Uriarte[1] ✉, Jasone Astorga[2], Eduardo Jacob[2], Maider Huarte[2], Oscar López[1]

[1]*Nextel S.A., Technological Park of Bizkaia, 48170 Zamudio, Spain*

[2]*Departament of Communications Engineering, University of the Basque Country UPV/EHU, Plaza Ingeniero Torres Quevedo 1, Ilbao (Bizkaia), 48013, Spain*

✉ *E-mail: muriarte@nextel.es*

Abstract: In the incoming internet of things (IoT) applications, smart sensors expose services to interact with them, to be parameterised, managed and maintained. Therefore, fine-grained end-to-end access control enforcement is mandatory to tackle the derived security requirements. However, it is still not feasible in very constrained devices. There is an innovative access control model that conveys an expressive policy language and an optimised codification for tight and flexible access control enforcement in very constrained devices. Such tightness enabled by the expressiveness of the policy language leads to detailed policy instances that might impact on the performance and therefore, in the feasibility and further applicability. In this context, this study assesses how the policy length impacts the performance of the establishment of a security association through the protocol named Hidra proposed by such an adapted access control model. Consequently, the notable results of the performance evaluation prove the feasibility and adequacy of this access control model for the new smart IoT scenarios.

1 Introduction

In the advent of the internet of things (IoT), access control and therefore security remains insufficiently solved in constrained devices. Existing approaches lack the required expressiveness of the policies to enforce fine-grained access control.

Regarding the security of IoT applications, two behaviours can be distinguished in sensor networks, where sensors and actuators are mainly expected to be cheap and constrained devices, constrained device sensors (CDSs) from now on, with low memory, low processor and usually relying on batteries. On the one hand, sensors are commonly configured to measure their environment and detect any significant change on it and then communicate as clients to a message broker with the proper labels. So in the communications between the sensors as producers and the message broker, both endpoints have identified each other previously and the security associations to assure confidentiality and integrity of data are static and known beforehand. Then, the message broker receives, normalises, relabels, routes and finally, based on a security policy, may release or not the messages to the final consumers identified as subscribers. Thus, this pushing behaviour from the sensors to the final subscribers through the aforementioned broker cannot be considered a direct end-to-end (E2E) communication.

On the other hand, sensors are expected to behave also as servers, and to offer end to end services to a priori unknown subjects in order to behave smart or process adaptive in large-scale deployments. In fact, the IoT concept envisions pervasive computing environments where a larger number of people and devices can discover and access services in things around them. For example, smart environments such as smart homes, cars, offices, elite sport training etc. consist of intelligent services on resources that are accessible to users through handheld devices or under *bring your own device* paradigm, as depicted in Fig. 1.

Therefore, the offered services in constrained devices are conceived to enable the tuning of a sporadic user experience, as well as management and maintenance of the CDS itself in several IoT domains. For example, in scenarios such as elite sport training monitoring or healthcare, where the user's profile is used to customise the sensibility and the identity in the CDS that uploads the monitored activity and health parameters.

Alternatively, in such smart scenarios, Industry 4.0 stands for the smartness of the assets [1] through information and communication technologies. The key aspect is the set of offered services equivalent to a remote administration. These services enable usage, operation, maintenance, and manageability in the life cycle and value stream of the connected things. Even more, there is an initiative called the web of things (WoT), which aims to evolve to wisdom WoT, to support smart web services [2].

The aforementioned maintenance and manageability features in a CDS are extensible to any IoT application sector and very demanded in multi-stakeholder scenarios where owners, manufacturers, integrators, developers, operators and service supporters stand for the service chain, and the operational cost (Opex) optimisation is a key objective. In fact, services on the CDS enable the tuning of the operation after commissioning, such as domain parameterisation, networking operation, functionality adaptation or extension, customisation (sensibility, accuracy, threshold, peering, periodicity etc.) or maintenance. For example, the scenarios where maintenance involves several parties and responsibilities, illustrate the opportunity and the criticism of the protection and nominal accounting of the exposed management services in the CDS.

The basic actors in the access control schema depicted in Fig. 1 are three: a subject (i), constrained or not, aiming to access a management or parameterisation service as a resource in a CDS (ii), with the collaboration of a trusted third party in a security association establishment step, namely the access control server (ACS) (iii). Usually, ACS and subject are not challenged by resource scarcity as in the case of CDSs, so they are in the less constrained level. However, CDSs that expose any service, need to tackle functional and security requirements with low memory, low processing capabilities, and very often relying on batteries. Concretely, the Internet Engineering Task Force defines ranges of constrained devices [3], as class 0 (C_0, <10 KiB of data size, <100 KiB of code size), class 1 (C_1, 10 and 100 KiB, respectively) and class 2 (C_2, 50 and 250 KiB), which are expected to be massively deployed and accessible in different applications.

In such open and dynamic applications, E2E policy based security is required to be enforced locally in the CDSs [4]. In fact, using security policies allows the security objectives to be tightly enforced and modified without changing the implementation of the

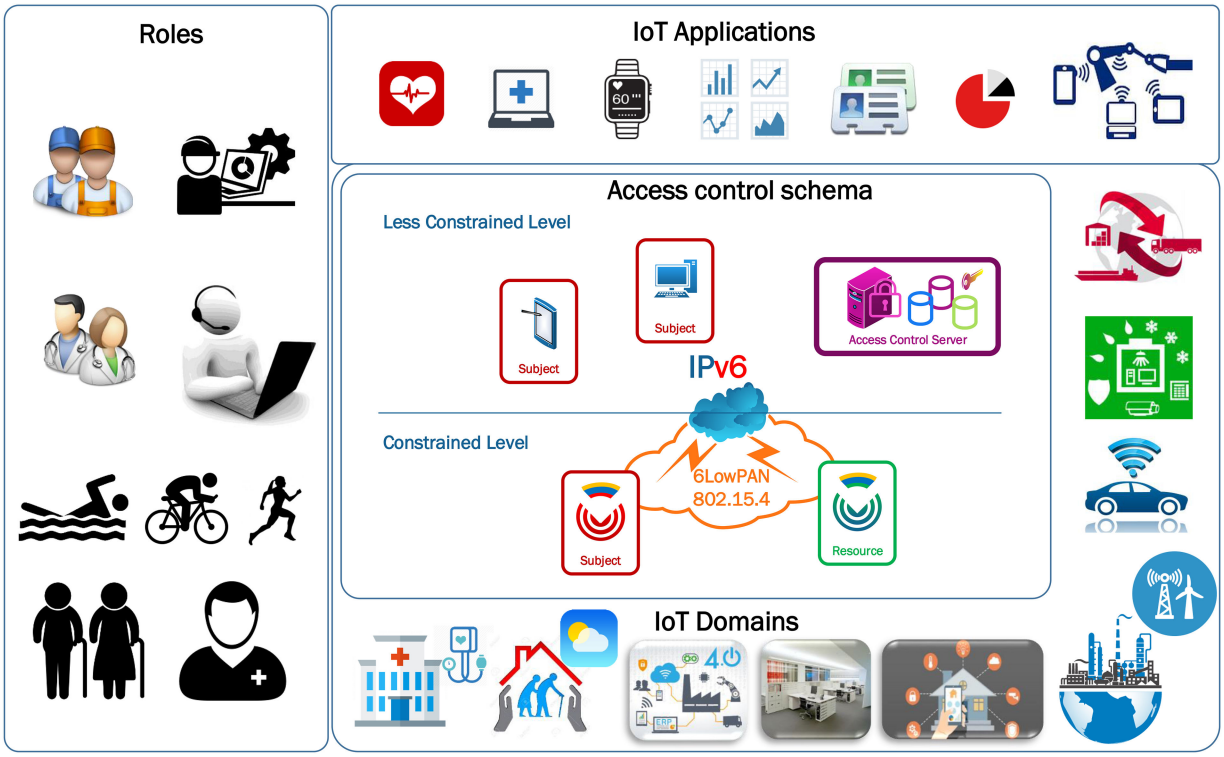

Fig. 1 *IoT applications support smart environments such as healthcare, elder care, cities, offices, logistics, transport, cars, elite sports etc. involving several roles which aim to access sensors in order to tune them through an effective but feasible access control schema*

involved entities. A policy language in this type of environment needs to be very expressive but lightweight and easily extensible.

A survey on the proposed security solutions for constrained devices summarised in Section 2, reflects a lack of adequacy and poor performance of access control security and activity tracking, due to the cost in power, communications and processor (CPU) that security mechanisms require. The expressiveness of currently feasible access control policies is very limited, while usually adopted access control list (ACL) approach is not scalable for E2E security in open scenarios. In fact, in the envisioned scenarios the number of potentially authorised subjects is high and unknown, and ACL-based access control means a big unaffordable storage in the CDS as analysed later in Section 3.2 as well as network overloading and energy consumption due to ACL updating tasks.

Given this background, there is an innovative access control model [5] that deals with a hybrid architecture and an expressive policy language for dynamic fine-grained policy enforcement in the sensor. This least privilege oriented enforcement is based on local context conditions and correspondent obligations, not only during secure session establishment but also afterwards while the security association is in use, in order to control the behaviour of the access. Such a dynamic policy cycle avoiding local storage requires an efficient message exchange protocol, named Hidra, in order to assure the mutual authentication, the expressive policy injection, the tight policy enforcement in the secure association establishment and the derived resource access, as well as the accounting for further tracking and auditing purposes.

In this context, this study assesses the impact on the performance of the tightness enabled by such an expressive policy language. This policy language supports fine-grained policies aiming at the least privilege principle enforcement but needs to tackle the challenges derived from the resource scarcity of the CDSs in order to guarantee its feasibility and consequent applicability.

Henceforth, the related work will be presented in Section 2 as the state of the art; the assessed access control model is briefly described in Section 3 for an easier exposition of the impact analysis of the policy length, covering (i) the policy language, (ii) the policy codification, (iii) the policy domain model, (iv) the architecture and the Hidra message protocol; the impact of the tightness on the performance is evaluated in Section 4; finally, the main conclusions of the paper are gathered in Section 5.

2 State of the art

In the last few years, the research area related to security in IoT has received a significant attention, dealing with the design of different architectures, security protocols and policy models. However, security still remains the main obstacle in the development of innovative and valuable services [6]. In fact, traditional security countermeasures cannot be applied directly to CDSs in IoT scenarios, because they are too resource consuming and not optimised for resource deprived devices. Additionally, existing feasible E2E access control approaches do not implement an expressive and therefore fine-grained and tight security policy enforcement [7].

For feasibility reasons, a centralised architecture based on traditional standards and protocols, where a central server with no resource constraints makes authorisation decisions for each access request, could be initially an examined option. However, this approach does not consider local context conditions in CDSs, and it implies high-energy consumption as well as network overhead due to continuous communications between the CDSs and the ACS.

For the local access control enforcement in the CDSs, instead of security rules coupled within the applications' logic, policy driven security management and enforcement have become the de facto approach in large-scale systems. In this regard, the tightness of the enforcement is enabled by the expressiveness of the relying policy language.

From the analysis of the most expressive foundational policy languages [8–10], Table 1 shows the main features that are conditions, obligations and re-evaluation. All of them support the *if–then* or *condition–action* paradigm with different attribute treatment. However, none of them is optimised for constrained devices, and consequently, they are not feasible in CDSs.

As representative by its level of adoption, extensible accessible control markup language (XACML) [11] relies on a generic authorisation architecture and specifies a complete and sound policy language to express and exchange authorisation policies represented in XML. However, XACML is too heavy for severely constrained devices. In fact, a CDS can hardly process an XACML

policy file of more than 50 lines of text conveying a single rule specification.

A recent alternative approach is the distributed capability based access control (DcapBAC) [12], where an unforgeable token exchangeable as a capability, grants access to its holder in a more agile way. However, the token is designed in an XML schema and it has not been validated in constrained devices.

In any case, this approach has been adopted by some other designs involving technologies specifically defined for IoT, which enable CDSs to make local authorisation decisions based also on local conditions [13], since the capabilities might include conditions represented as tuples (type, name, value). Per contra, this approach is based on public key cryptography (PKC), which is heavier than symmetric key cryptography (SKC) by means of resource consumption. Additionally, the conditions are limited to matching because the approach does not support expressions, its syntax is not optimised by means of codification since it uses JSON, it does not support the enforcement of additional obligations and it has been validated in not so constrained C_2 CDSs.

In this line, the delegated CoAP [14] authentication and authorisation framework [15] defines a token to distribute pre-shared keys, and if authorised, a handshake is done to establish a datagram transport layer security channel. Local authorisation policies are specified as conditions serialised in a concise binary representation (CBOR), instead of JSON, aiming at compacted payloads in CoAP protocol. However, CBOR is a general purpose serialisation solution and the resulting compression is not sufficiently optimised for security policies in very constrained C_0 and C_1 CDSs, where fine-grained access control is aimed through a higher but feasible policy language expressiveness, beyond the existing local attributes matching as conditions.

In another line, the usage control model (UCON) [16] and the attribute-based policy schema [17] extend traditional access control systems to a continuous protection of the resource during access by the definition of obligations to enforce usage control, but there is no approach addressing the feasibility in CDSs.

Attending to the protocols for the instant provisioning of the policy during the E2E security association in a secure session, Ladon [18], which is inspired in Kerberos that was designed for non-constrained scenarios, has been evolved for that purpose. In fact, Ladon is specifically designed for very constrained C_0 and C_1 devices, but it does not directly support the provisioning of an expressive policy.

Hidra protocol conveyed in the assessed access control model [5] evolves Ladon to enable the dynamic provisioning of expressive policy instances as well as accounting. In fact, currently no other suitable solution exists to provide authentication and fine-grained authorisation processes in the envisioned scenarios of constrained but manageable sensor networks, and additionally, neither of the above-considered approaches implements any accounting feature.

Hidra allows conveying a highly expressive policy language, which is ready to define detailed policy instances that consist of rules with conditions based on expressions beyond the simple matching of attribute values. For example, checking the battery level in a range, a counter below a threshold, or a change in a status variable result much more expressive than checking specific static values. Additionally, the policy language allows both re-evaluation and obligations. In fact, feasible approaches used so far do not support the definition of different rules related to specific actions on a resource. If the action extends on time, the approaches implemented thus far do not support a way to re-evaluate the rule and perform a usage control, or simply to check if the local context conditions remaining in proper status. Neither is there a way to run any additional task, called obligation, as the reaction to a rule evaluation, and therefore, features such as updating of counters for activity control, locking of data for transaction management *etc.* that are required for tighter and smarter enforcement are not solved by the approaches implemented until now.

These features of the policy language such as (i) granular rules based on expressive functions as conditions over the subject, resource, action, and context attributes, (ii) reactive obligations and (iii) usage control oriented re-evaluation in a lightweight way, which are related to Hidra are shown in Table 1.

In any case, this access control model that has been initially validated for C_0 CDSs with a simple policy instance requires an assessment of the impact of the expressiveness (and related tightness) of the policy instances on the performance in order to deeply evaluate its feasibility and further applicability.

Finally, the most recent access control related proposals, still far from the required expressiveness, have been validated in C_2 CDSs with minimal policy instances, and their feasibility has not been validated in C_0 and C_1 CDSs. There is no such assessment for the aforementioned access control models and protocols.

3 Access control model

Hereinafter a brief description of the access control model under evaluation is presented for a better understanding of the assessment of the impact of the expressiveness of the policy instances on the performance. A detailed description of the access control model can be found in [5].

Summarily, the assessed E2E access control model is based on an efficient policy language and codification, which are specifically defined to gain expressiveness in the authorisation policies and to keep the viability in very constrained C_0 and C_1 devices. Besides the policy language, the access control model conveys the E2E feasible security association between two mutually authenticated peers and consists of an architecture to enable multi-step authorisation as well as a protocol for the provisioning and enforcement of a dynamic security policy in the CDSs.

3.1 Authorisation policy language

In this section, the expressive policy language detailed in [5] is summarised. The goal of this policy language is to enable the enforcement of tighter access control policies in CDSs, overcoming the resource constraints. In fact, this policy language definition enables both to make granting decisions based on local context conditions and to react accordingly to the requests by the execution of additional tasks defined as obligations.

Table 1 Summarised overview of foundational policy languages

	Conditions			Obligations	Re-evaluation
	Role matching	Attribute matching	Expressive functions		
XACML	x	x	x	x	
Rei	x	x	x	x	
Ponder	x	x	x	x	
ASL	x				
OSL		x	x	x	
APPEL		x			
EPAL		x			
CapBAC		x			
UCON		x		x	x
Hidra	x	x	x	x	x

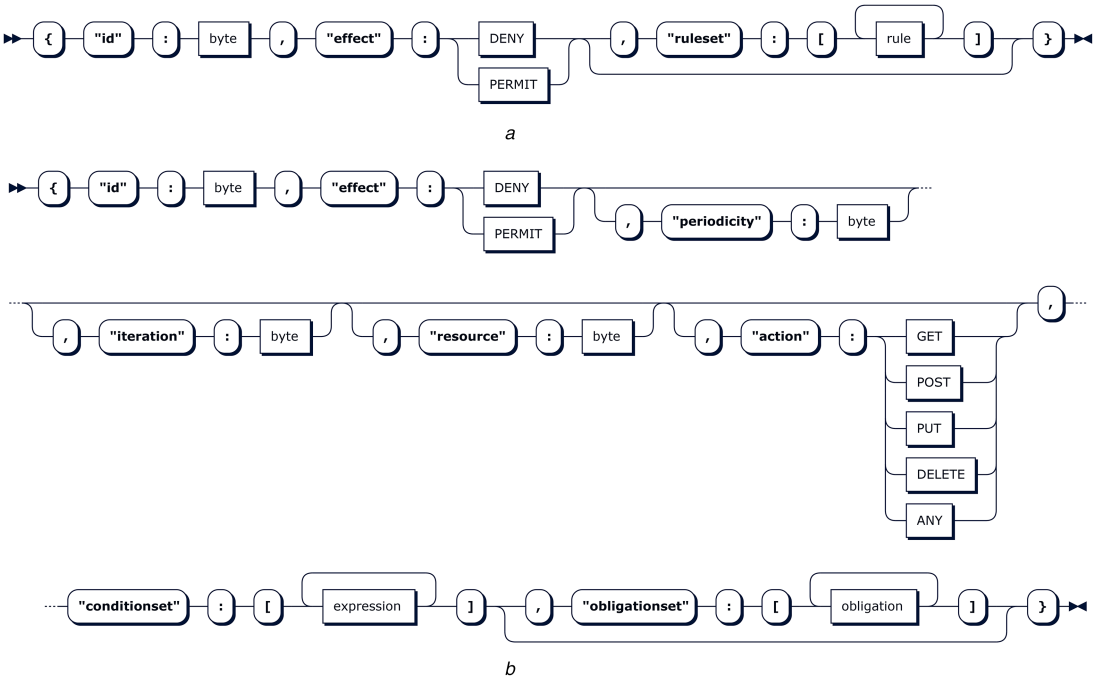

Fig. 2 *Authorisation policy language main constructs*
(a) Policy construct definition, ***(b)*** Rule construct definition

A resulting policy instance is defined, like in the general event-condition-action approaches, as an optional set of rules, which specifies both the conditions to be checked and the related reactions, in enforcement time. Specifically, this policy language stands for a sequence of constructs with a particular meaning in the decision making and enforcement time.

Some of the constructs are defined as mandatory, and some others as optional, enabling to shorten the length of the policy when a simple policy is enough. Additionally, some constructs are extended through other nested constructs, and some of them can be instantiated many times within a container construct. Related to this elasticity feature, the more constructs, the higher the expressiveness of the policy, so the more granular the policy is, and thereupon the tighter the enforcement is. Consequently, the challenge to overcome is to be feasible even in the most expressive use-case.

The policy language enables a policy instantiation through the *policy* construct, with three nested constructs as depicted in Fig. 2.

First of all, a policy instance identification, *id*, is specified for logging, tracking and auditing purposes. Then, a default policy granting *effect* is specified. This effect will prevail in the case of absence of rules, or any rule evaluation conflict. This construct is very useful in most simple policy instances, with no rules, where authentication or preliminary authorisation in the ACS is condition enough to grant access in a request. It is also useful for notifying revocation and related security association finalisation. Lastly, optionally, an array of rules may be instantiated as a *ruleset* to specify the conditions and related reactions. Each *rule* in the array is an extendible construct.

The *rule* construct is defined as a sequence of eight nested constructs, where the order is crucial. Some of them, such as *id*, *effect*, and *conditionset* are mandatory, and the rest named *periodicity*, *iteration*, *resource*, *action,* and *obligationset* are optional. In fact, *periodicity* and *iteration* are used to specify policy re-evaluation timing and repetitions, respectively. Moreover, the granularity of the policy aiming at least privilege enforcement is enabled by the optional specification of *resources* and *actions*. Concretely, they enable to trigger tighter rules to be triggered depending on the request.

The *conditionset* and *obligationset* are arrays of expressions and obligations, respectively. These repeatable and extensible *expression* and *obligation* constructs are defined in a similar way enabling the instantiation of rich expressions on attributes declared as inputs as well as reactive tasks declared as obligations.

This policy language is highly inspired in XACML as representative by its level of adoption, and it adapts the most significant constructs. The number and meaning of the constructs in the policy language result from the compromise between expressiveness and performance. In fact, as an approximation to a coarse completeness and soundness overview, the proposed constructs enable the full set of features mentioned in Section 2. Such features are (i) granular rules based on expressive functions as conditions over the subject, resource, action, and context attributes, (ii) reactive obligations and (iii) usage control oriented re-evaluation in a lightweight way, as shown in Table 1.

This policy language can be used to implement the most extended access control models, i.e. role-based access control and attribute-based access control (ABAC) models, using the same set of constructs in different domains, which leads to simpler policy instances, and more uniformity.

3.2 Policy language codification

This section summarises the codification of the policy instances that notably reduces the length of the policy file. The length of any policy instance, in a human readable format, grows proportionally with the aimed tightness, and it would impact negatively on the performance. So a policy instance codification is proposed, distinguishing from existing ones that serialise policy instances through standardised generalist solutions such as CBOR. Existing serialisation approaches do not optimise the agreed common understanding of the constructs by means of their sequence, meaning, type, scope nor elasticity. Consequently, the compression ratios of currently available approaches remaining lower than the one resulting from the policy codification proposed in this section.

The proposed policy codification serialises each construct and concatenates them in a bit stream. In fact, it takes profit of beforehand knowledge of the defined sequence of the constructs, and their format. An additional crucial factor is an injection of (i) some agreed bit masks, to specify the existence or not of optional constructs and (ii) some related array length bits. It enables to deal optimally with the elasticity defined in the policy language, avoiding unused but expected fields of expressive policies, greatly reducing the length. In the CDS side, the de-codification and enforcement are based on the same principle of beforehand construct sequence knowledge as well as the shared semantics for the injected bit masks and array lengths as detailed in [5].

Table 2 Length comparison for different representations of four instances

Representation	Nature	Length (bytes)			
		IS_1	IS_2	IS_3	IS_4
JSON	human-readable text	30	164	236	798
JSON'	pre-processed text	23	118	174	554
CBOR	binary stream	14	81	123	391
APBR	optimised binary stream	2	7	9	32

Table 3 Length comparison for different representations of four instances

Use case	Resources	Roles	Subjects	APBR		ACL	
				Policy instance	Bytes	Stored policy	Bytes
UC_1	4	3	8	IS_{4Rules}		$RWX(Rl_i, R_i)$	4
						subject-role mapping	24
						subject index	40
					8		68
UC_2	8	3	8	IS_{8Rules}	13	||	72
UC_3	8	3	32	IS_{8Rules}	13	||	264
UC_4	8	3	64	IS_{8Rules}	13	||	520
UC_5	8	3	128	IS_{8Rules}	13	||	1032
UC_6	8	3	256	IS_{8Rules}	13	||	2056

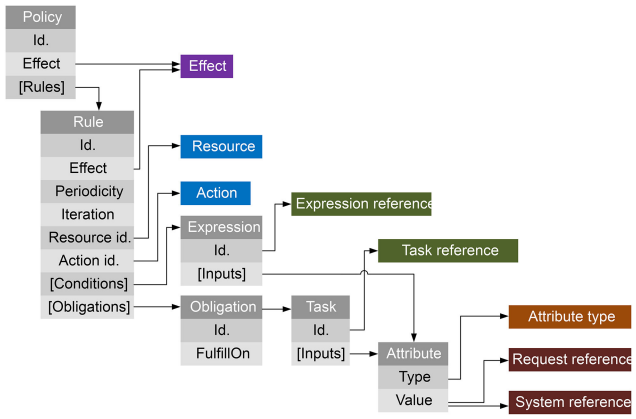

Fig. 3 *PDM conveying the constructs of policy language and the reference indexes specifying additional domain-related conventions*

With respect to covered policy formats, this authorisation policy binary representation (APBR) can easily be applied to any original policy instance format (XACML, JSON *etc.*), from textual files to structured policy instance representations. For example, four different policy instances of the sample (IS) explained after in the performance evaluation section, can be represented in JSON with lengths of 23, 118, 174, and 554 bytes. They can also be codified in CBOR with lengths of 14, 81, 123, and 391 bytes, respectively, and they are notably reduced to 2, 7, 9, and 32 bytes using the proposed codification, as shown in Table 2.

At this point, it is worthy to examine traditional ACLs so broadly adopted in closed scenarios, where each resource has defined a set of subjects and related permissions. One accepted implementation is to define such permissions per resource based on roles. This approach could be codified as the simplest read-write-execute (RWX) for each role (Rl_i) per resource (R_i), $RWX(Rl_i, R_i)$, in a byte per resource. This implies to keep also a subject index table as well as a subject-role mapping table if nominal access control is to be enforced.

So in a use case of eight subjects with three roles accessing four resources with three actions, the corresponding ACL codification would result in 68 bytes as shown in Table 3. In a similar way, doubling resources could be codified with 72 bytes. However, when the number of subjects rises to 32, 64, 128 or 256 the codification of the ACL implies 264, 520, 1032 and 2056 bytes, respectively.

On the other hand, ACLs result rather static compared with the policies instantiated and codified with the assessed policy language, which enclose as many rules as resources to cover the most stressing case. To compare, the evaluated APBR codification of the same use cases would imply policy instances of four and eight rules (IS_{4Rules}, IS_{8Rules}) codified constantly with 8 and 13 bytes whatever is the number of subjects, roles and actions over the resources.

Besides enforcement based on ACLs is far from being flexible and it does not consider local context nor enables to launch any reactive obligation. Additionally, the length of the ACLs notably grows and it scales unacceptable when the number of subjects rises moderately compared with the expected widespread increase of potential subjects in the envisioned open scenarios. Instead, the length of the policy instances with the proposed policy language and codification remains constantly low independent of the subject number, as shown in Table 3.

3.3 Policy domain model

The policy domain model (PDM) defines the assumptions that are useful to modulate the length in a trade-off with the expressiveness. These assumptions are related to the possible values in a construct, the repetitiveness of sub-constructs as the lengths of the arrays, or their reference through identifiers pointing to attributes, functions, resources etc.

This PDM consists of the detailed definition of the policy language (in grey) and the adopted conventions (in different colours), as denoted in Fig. 3.

The PDM definition enables different gradual implementations that could be patterned as it is done nowadays by most CDS implementations. When the PDM is specified through decoupled files, their edition, modification, provisioning, and activation is much more agile. Through the decoupled PDM specification file, better abstraction, higher scalability, and flexibility as well as multi-domain applicability are obtained, namely access control model manageability in the CDS.

3.4 Hidra messaging protocol

To efficiently convey the aforementioned access control policies to the CDSs, Hidra messaging protocol is defined in [5], and briefly described hereinafter. Hidra, depicted in Fig. 4, is based on three parties architecture and provides authentication, authorisation in two steps, dynamic policy provisioning, and accounting.

Hidra is based on Ladon [18], which is a validated solution for the establishment of E2E security associations, through pair-wise

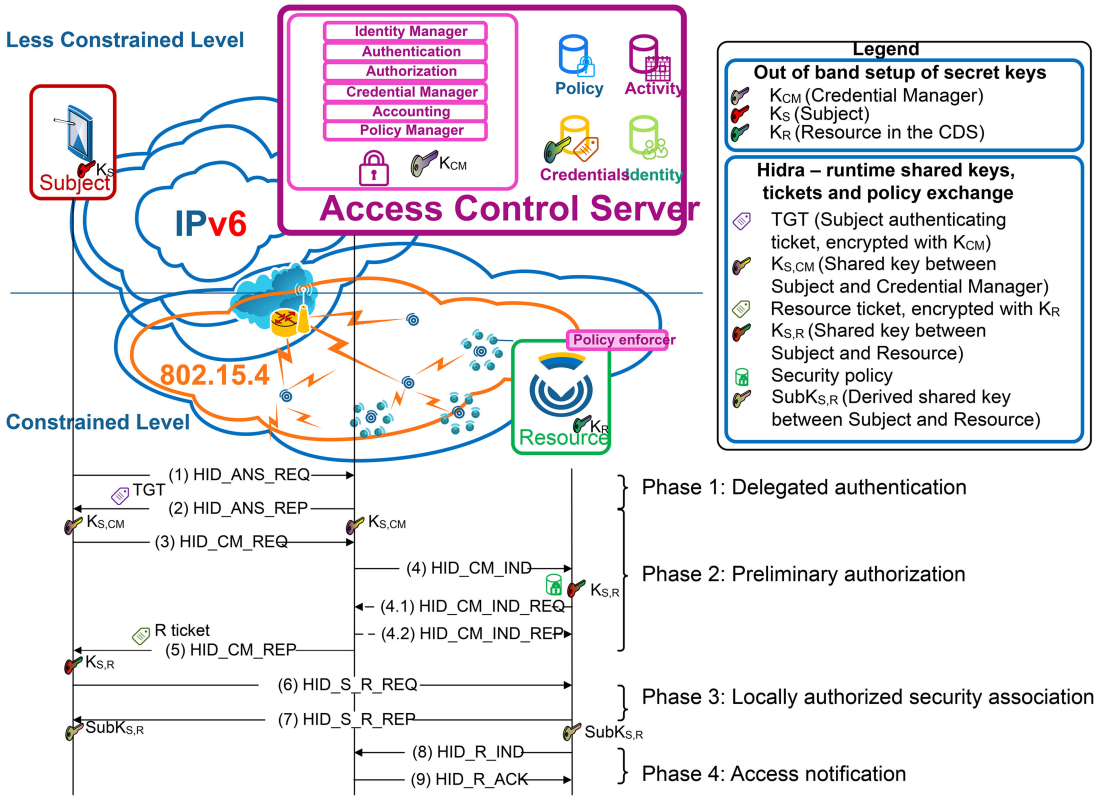

Fig. 4 *Hidra protocol messages and security association establishment related authentication, authorisation, key exchange and notifications*

keys, guaranteeing mutual authentication and authorisation in very CDSs.

Hidra is based on SKC and it assumes that each endpoint owns a secret key shared with the ACS. The operation is based on the use of tickets, a capability distributed by the ACS that contains a proof of the identity of the subject that requests it. Tickets are encrypted so that only the entities which they are intended for, are able to decrypt them.

After a successful authentication in the ACS (phase 1) the subject obtains a ticket granting ticket (TGT). This TGT is used by the subject to obtain resource tickets (phase 2) required to access any resource on the CDSs.

This approach enables the ABAC authorisation enforcement in two steps. On the first one, as a condition to release any resource ticket, fine-grained preliminary access control is performed in the ACS (phase 2), focusing on the attributes of the subject, resource, and expected actions. If this first authorisation step is successful, the ACS sends a message to the subject including a resource ticket and a message to the CDS conveying an expressive authorisation policy instance. This instantaneous custom policy provisioning is innovative and advantageous since it avoids permanent policies' storage in the CDS and reduces network overhead. In fact, typically, policies might be enclosed in the resource ticket such as in Ladon and DcapBAC. However, such an approach implies increasing the length of the ticket, which is a long structure by itself and can result in packet fragmentation, and therefore, additional network overload, due to the short available payloads of IEEE 802.15.4 frames. Therefore, Hidra takes advantage of the HID_CM_IND message to efficiently convey the access control policy to the CDS.

On the second authoritative step, once the subject has obtained a resource ticket, the local context-based access control is performed in the CDS (phase 3). First, the proper fine grained rule is evaluated to make the granting decision, and then the corresponding reactive actions are enforced. In a positive case, the result is (i) the establishment of a session key to be used on further E2E resource access exchanges as any security association protocol and (ii) a cutting-edge fresh policy provisioning that is custom and very granular, and which is enforced during the security association.

Another novelty with respect to any security protocol is the addition of a pair of messages to enable precise accounting (phase 4). By means of these messages, the CDS will notify details like who performed what, where and when in each and every access request received from the subject. These notifications are gathered, normalised, and treated properly by the ACS. Additionally, the ACS can react and send a HID_CM_IND message with a fresh policy instance, enabling the dynamic delegation, request, cancellation, and revocation of permissions.

Then, while the security association is not finalised, the access control is enforced and accounted in the CDS autonomously in each and every further request attempt, since the received expressive fresh policy (phase 2) includes also related rules. In this specific aspect, Hidra enables local context policy based access control enforcement with maximum granularity and minimum policy storage and footprint compared with static generic policy based approaches.

Consequently, besides a tight and continuous local context based enforcement, a unified, coherent and adaptive management of the policies by the ACS is achieved. These two main features are not covered this way by any other security association establishment protocol nowadays. Additionally, the Hidra protocol and the adopted architecture enable to rely the most expensive features on the ACS, which entails the usage of standard security and access control technologies in the non-constrained interactions. It also achieves that most unauthorised access attempts are refused before reaching the CDS, avoiding unsuccessful message exchanges and thus, saving energy in the CDS, which is a crucial aspect.

4 Performance evaluation

The analytical performance evaluation described here focuses on the impact of the length of the policy on two critical parameters for the envisioned scenarios: (i) the delay introduced by the access control model for an authorised E2E secure session establishment and (ii) the energy cost on the CDS's battery of such secure session establishment. The delay needs to be under an accepted value and the energy consumption cannot exceed a rationale and proportional limit.

The reference scenario for the performance evaluation is depicted in Fig. 4. In this scenario, a subject is connected to the Internet and establishes an E2E connection with a resource running on a CDS in an IEEE 802.15.4 network. A 6LoWPAN [19] router in orange, acts as the LowPAN coordinator and connects a beacon-enabled cluster-tree structure to the Internet. The IEEE 802.15.4 network is three-hops deep, which is considered significantly large for validation. The PAN router coordinator has three child coordinators, which have another three child coordinators each, controlling a cluster of six leaf nodes where each CDS exposes resources as management services. Therefore, 54 ($3 \times 3 \times 6$) CDSs are integrated into the network, but details of the branches have been omitted for the clarity of the picture.

4.1 Time computation

For the computation of the E2E secure session establishment time, four contributions are considered for each of the messages exchanged in Hidra: the time to generate the message in the origin, the network transmission time, the queue waiting time in the destination, and the time to process the received message.

In the computation of the generation and processing time, the execution of the cryptographic operations is the most significant ones, and the rest of the operations are comparatively negligible. In fact, among the cryptographic functions two different constant bit rates are considered for encryption and MAC computation, and it implies that the length of the messages impacts directly on the computation of this time.

Equation (1) shows the calculation of the service time as encryption time, related to the message i in the entity X, which is computed based on aforementioned two terms, namely $\left|\text{CRYP}(\text{Message}^i)\right|$, which represents the number of bytes of the fields in the message i that are subject to cryptographic operations and TCRYP_X, which denotes the constant bit rate as bit/s for encryption functions in the corresponding entity X

$$S_{Xi} = \frac{\left|\text{CRYP}(\text{Message}^i)\right|}{\text{TCRYP}_X}. \tag{1}$$

In the case of the computation of the transmission time, backoff delay (D_{BOT}), which depends on the network density and has been computed following the model in [20], and propagation delay (D_{Tx}), which depends on the length of each message $\left|\text{Message}^i\right|$, are considered. Equation (2) shows the calculation of the transmission time in each of the three hops of the evaluation scenario ($D1$, $D2$, $D3$), where a constant propagation rate (R) in kbps is considered for all the IEEE 802.15.4 links

$$D_l = D_{\text{BOT}}^l + D_{\text{Tx}}^l, \quad \text{for } l = 1, 2, 3, \tag{2}$$

where

$$D_{\text{Tx}}^l = \frac{\left|\text{Message}^i\right|}{R}. \tag{3}$$

So, the total transmission time is calculated as

$$E[t_i] = D_1 + D_2 + D_3. \tag{4}$$

Related to the queue waiting time, each entity is modelled as an M/G/1 queue, Hidra related messages are only considered and subjects generate requests according to a Poisson distribution. Therefore, considering the resource utilisation as $\rho = \lambda\overline{X}$, the random variable for service time of job i as X_i, and the second moment of service time as $\overline{X^2} = E[X_i^2]$, the average waiting time in the queue for each entity is computed according to Pollaczek–Khinchin mean formula [21], as

$$W = \frac{\lambda\overline{X^2}}{2(1-\rho)}. \tag{5}$$

4.2 Energy computation

For the computation of the energy cost in the CDS, the energy consumed by communications, namely transmission or reception of Hidra messages as bits over the air, as well as the computation of cryptographic operations are considered.

Equation (6) shows the calculation of the energy consumption by the communications in the reception of a message in the CDS X, based on the energy consumed to receive a message i (ε_{Rx}), where P_{Rx} denotes a constant reception power consumption and R denotes a constant wireless link data rate in the LoWPAN network

$$\varepsilon_X = \varepsilon_{\text{Rx}} = \frac{\left|\text{Message}^i\right|}{R} P_{\text{Rx}}. \tag{6}$$

Additionally, the energy consumed during the backoff process is also considered for the transmission. Consequently, (7) shows the calculation of the energy consumption by the communications in the transmission of a message in the CDS X, based on the energy consumed during the backoff processes (ε_{BOT}) computed following the model in [20] and the energy consumed to transmit a message (ε_{Tx}), where P_{Tx} denotes a constant transmission power consumption

$$\varepsilon_X = \varepsilon_{\text{BOT}} + \varepsilon_{\text{Tx}}, \tag{7}$$

where

$$\varepsilon_{\text{Tx}} = \frac{\left|\text{Message}^i\right|}{R} P_{\text{Tx}}. \tag{8}$$

Related to cryptographic operations involved in the generation and processing of the messages in the CDS, two different constant bit rates are considered for encryption and MAC computation. Equation (9) shows the calculation related to the message i in the entity X, which is computed based on the length of the fields subject to cryptographic operations ($\left|\text{CRYP}(\text{Message}^i)\right|$), and the cryptographic operation rate TCRYP_X, considering a constant instantaneous power consumption (P_{C}) for the computation

$$\varepsilon_{Xi} = \frac{\left|\text{CRYP}(\text{Message}^i)\right|}{\text{TCRYP}_X} P_{\text{C}}. \tag{9}$$

4.3 Analysis scenario

In the aforementioned reference scenario depicted in Fig. 4, a subject is connected to the Internet and establishes an E2E connection with a resource running on a CDS in an IEEE 802.15.4 network of 54 nodes, which is three hops deep. The considered effective data bit rate is of 70 kbps, so the scenario network is significantly large and stressed for validation purposes.

Regarding message exchange, queuing theory is considered, where each security association request generates a new job in the queue, and the most demanding case of a request per hour has been considered. This demanding rate could be the case of a CDS that is tuned for a better user experience with several user parameters, e.g. physiological parameters in an elite training monitoring scenario. This case can be considered more demanding compared with either management or maintenance tasks, which are much more sporadic.

The key performance factor is the length of the policy since it impacts proportionally on the delay, the energy consumption, and therefore in the feasibility.

Since the provisioned policy is elastic, four samples have been defined for the analysis of Hidra:

- Sample 1, a policy with no rules and comparable in expressiveness with Ladon, validated in C_0 CDSs [22]. That could be the case of the access granted to a subject initially authenticated and authorised in the ACS, and then just authenticated in the CDS.

- Sample 2, a policy with one rule with conditions, slightly comparable in expressiveness to existing DcapBAC approaches based on PKC and CBOR codification of the policy, which are validated in not so constrained devices (C_2 CDSs). In fact, the proposed approach enables rich expressions on attributes, instead of the simple matching of DcapBAC. For example, beyond the initial authorisation in the ACS and local authentication in the CDS, access for any maintenance action is only granted after checking that battery level is greater than a given threshold.
- Sample 3, a policy with one rule with conditions and obligations, which is beyond the other existing feasible solutions in expressiveness and tightness of the enforcement. For example, after checking local conditions such as battery status, while granting any maintenance action, the system status flag or semaphore is updated accordingly as reaction enabled by obligations.
- Sample 4, a policy with two rules with conditions, obligations and periodical re-evaluation, which is far beyond the other existing feasible solutions. This sample 4 incrementally covers the checking of different conditions related to different actions on different resources such as system status attributes checking before maintenance actions and attempts counter checking before administration granting, which both point out a bigger granularity. Additionally, each of the accesses may produce particular reactions as obligations such as system flags updating enabling transactional controls, counters updating enabling usage controls, or concrete remote notifications enabling instant awareness. Also, finally, both rules can be re-checked after a while to see on the one hand, whether system flags or activity counters remain under accepted values to preserve granted rights, or on the other hand, whether the policy has been updated, e.g. revoking any previous right due to an anomalous behaviour notification.

For such usable samples, the length of specific policy instances varies depending on specific field values and iterations of nested constructs. For each of the four samples, exemplary policy instances (IS_i) have been considered in order to calculate specific lengths as shown in Table 4. Additionally, Table 5 shows the message lengths of the Hidra protocol computed with the four policy instances enclosed in the HID_CM_IND message, as well as the lengths of the fields that are subject to cryptographic operations, encryption as well as MAC computation in the CDS. Message composition and lengths per field are detailed in [5].

4.4 Performance analysis

Currently, there is no approach validated for C_0 and C_1 CDSs in the literature which equals Hidra in expressiveness and functionality, both in enforcement and accounting. Instead, existing alternatives have been validated for C_2 CDSs, which are more powerful devices with bigger batteries. In any case, there is no comparable performance analysis based on the tightness of the enforcement if not Ladon, which also enables the establishment of an authenticated and centrally authorised security association and could be comparable to sample 1 of Hidra.

In this performance analysis, the impact of the policy length on the performance of Hidra with different samples is assessed. Table 6 shows the reference parameters including power consumptions corresponding to a MEMSIC TelosB mote (TPR2420CA) [23]. Such parameters have been considered to compute both the average delay and the energy consumption conveyed in Fig. 5 in the establishment of a security association, according to (4) and (9) computed on messages [1–9] detailed in Table 5.

Attending to the delay, the length of the four samples impacts proportionally rising up to 145 ms in the most common samples 1–4. Logically, the functionally comparable IS_1 delay is very similar to Ladon since innovative accounting messages are posterior to the session establishment response message to the user and do not affect the delay experienced by the user. In all cases, the delay is under the ITU-T Y.1541 limit of 400 ms for E2E delay of interactive transactions and the maximum allowed E2E delay noticeable by users in traditional IP applications, which is 1000 ms. Considering the high improvement in the control features, the delay related to the most usable policy instances [IS_1, IS_4] remains under 160 ms, which is acceptable according to even the maximum acceptable value of 1000 ms assessed by Stallings [24] as good quality for the response time of interactive go and back applications.

Attending to the energy consumption, measurements point out that the impact on energy consumption introduced due to a secure session establishment using Hidra with four different policy samples keeps proportional to the policy length. The measured energy consumption rises up to 3.8 mJ, which is really low compared with battery capacities around the 5940 J (the millionth part), and considering the big improvement in the control features, the impact remains more than acceptable. Logically, the addition of

Table 4 Codification lengths of four instances of samples

Sample	Instance of sample (IS) and description	Length	
		Bits	Bytes
1	IS1: no rules, just policy id. (for tracking) and granting effect	10	2
2	IS2: one rule with one condition with one input	53	7
3	IS3: one rule with one condition and one obligation	67	9
4	IS4: two rules with three conditions and one obligation, and periodical re-evaluation during access session	258	32

Table 5 Lengths of Hidra protocol messages with four different samples

Message type	Length, bytes	Subject to cryptographic operations	
		Encryption, bytes	MAC, bytes
HID_ACS.ASN_REQ	15	—	—
HID_ACS.ASN_REP	62	—	—
HID_CM_REQ	47	—	—
HID_CM_IND	35/40/42/65	2/7/9/32	29/34/36/59
HID_CM_IND REQ	14	—	10
HID_CM_IND REP	22	—	26
HID_CM_REP	62	—	—
HID_S_R_REQ	60	52	—
HID_S_R_REP	32	32	—
HID_R_IND	26	24	—
HID_R_ACK	14	—	10

Table 6 Parameters used to define the operation of the Hidra protocol

Parameter	Description	Value
TCRYP_S	subject encryption rate	50 Mbps
TCRYP_R	resource encryption rate	50 Kbps
TMAC_R	resource MAC computation rate	250 Kbps
$\mathrm{TCRYP}_{\mathrm{ACS}}$	ACS encryption rate	100 Mbps
$\mathrm{TMAC}_{\mathrm{ACS}}$	ACS MAC computation rate	200 Mbps
λ_0	mean job generation rate	1 request/h
N_{R}	number of resources	54
N_{S}	number of subjects	1
B	effective wireless link data bit rate	70 kbps
P_{RX}	power consumption in reception mode	74.4 mW
P_{TX}	power consumption in transmission mode (0 dBm)	65.7 mW
P_{C}	power consumption in cryptographic processing mode	5.4 mW

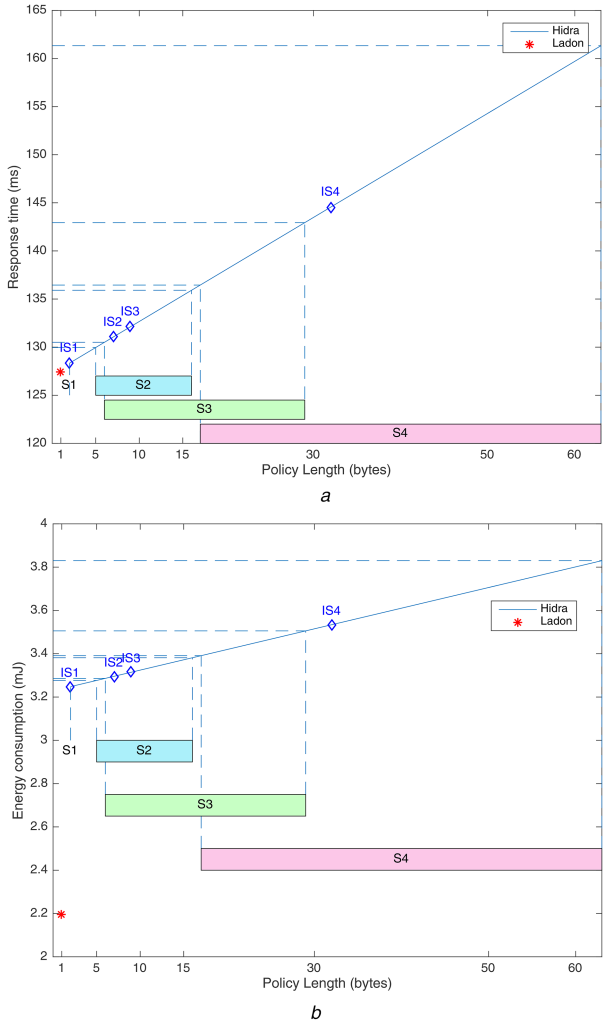

Fig. 5 *Performance analysis of a secure session establishment using Hidra with the instances related to the four policy samples*
(a) Impact of the policy length on the average delay, ***(b)*** Impact of the policy length on the energy consumption

accounting features compared with Ladon implies the exchange of two additional messages and therefore, a rise in energy consumption but always under acceptable values.

5 Conclusion

Incoming smart scenarios enabled by IoT envision smart objects exposing services to be adapted to user experience or to be managed to aim at higher productivity, often in multi-stakeholder applications. In such environments, smart things are cheap, therefore constrained devices, but critical components, so security is a must. Existent approaches do not cope with the principle of least privilege since they lack expressiveness and updating of the policy to be enforced in the CDSs, as well as additional control through obligations or accounting.

There is an innovative access control model that deals with a hybrid architecture and an expressive policy language for dynamic fine-grained policy enforcement in the sensor. This least privilege oriented enforcement is based on local context conditions and corresponding obligations, not only during secure session establishment but also afterwards while the security association is in use, in order to control the behaviour of the access. Such a dynamic policy cycle avoiding local storage requires an efficient message exchange protocol, named Hidra, in order to assure the mutual authentication, the expressive policy injection, the tight policy enforcement in the secure association establishment and the derived resource access, as well as the accounting for further tracking and auditing purposes.

Considering the challenge of the feasibility unsolved by the existing fine-grained access control models, the tightness of the enforcement is a trade-off with the expressiveness of the policy instances. The key performance factor is the length of the policy instance, since it impacts proportionally on the delay, the energy consumption, and therefore, in the feasibility. Since the analysed policy language is elastic, four samples of incremental tightness are instantiated. Such policy instances are provisioned on a CDS through the Hidra security protocol over an 802.15.4 network of three hops deep and 54 CDSs, which is parameterised with current C_0 and C_1 capabilities and empirical network data rates. In this realistic and stressing scenario, the resulting impact rises proportionally to the length of the policy but always under acceptable values.

Therefore, the assessed access control model is the first approach to bring to C_0 and C_1 CDSs a similar expressiveness level for enforcement and accounting as in the current Internet. The positive performance evaluation concludes the feasibility and suitability of this access control model, which notably rises the security features on the CDSs for the incoming smart scenarios.

Finally, there is no comparable impact assessment of policy expressiveness of any other access control model and presented the analysis model, as well as results, might be a reference for further analysis and benchmarking.

6 Acknowledgments

Part of this work is funded by the Department of Economic Development and Competitiveness of the Basque Government through the SEKUrtasun TEKnologiak SEKUTEK KK-2017/00044 collaborative research project and by the Spanish Ministry of Economy, Industry and Competitiveness through the State Secretariat for Research, Development and Innovation under the 'Adaptive Management of 5G Services to Support Critical Events in Cities (5G-City)' project TEC2016-76795-C6-5-R.

7 References

[1] Lee, J., Bagheri, B., Kao, H.-A.: 'A cyber-physical systems architecture for industry 4.0-based manufacturing systems', *Manuf. Lett.*, 2015, **3**, pp. 18–23
[2] Zhong, N., Ma, J.H., Huang, R.H., *et al.*: 'Research challenges and perspectives on wisdom web of things (W2T)', *J. Supercomput.*, 2010, **64**, (3), pp. 862–882
[3] Bormann, C., Ersue, M., Keranen, A.: '*Terminology for constrained-node networks*', RFC 7228 (RFC Editor, 2014) Available at http://www.rfceditor.org/rfc/rfc7228.txt, accessed 19 December 2016
[4] Roman, R., Zhou, J., Lopez, J.: 'On the features and challenges of security and privacy in distributed internet of things', *Comput. Netw.*, 2013, **57**, (10), pp. 2266–2279
[5] Uriarte, M., Astorga, J., Jacob, E., *et al.*: 'Expressive policy based access control for resource-constrained devices', *IEEE Access*, 2017, **PP**, (99), pp. 1–1
[6] Sicari, S., Rizzardi, A., Grieco, L., *et al.*: 'Security, privacy and trust in internet of things: the road ahead', *Comput. Netw.*, 2015, **76**, pp. 146–164
[7] Yan, Z., Zhang, P., Vasilakos, A.V.: 'A survey on trust management for internet of things', *J. Netw. Comput. Appl.*, 2014, **42**, pp. 120–134
[8] Han, W., Lei, C.: 'A survey on policy languages in network and security management', *Comput. Netw.*, 2012, **56**, (1), pp. 477–489, Available at http://www.sciencedirect.com/science/article/pii/S1389128611003562
[9] Barker, S.: 'The next 700 access control models or a unifying meta-model?'. Proc. 14th ACM Symp. on Access Control Models and Technologies (SACMAT '09), 2009, pp. 187–196, Available at http://doi.acm.org/10.1145/1542207.1542238
[10] P. L. I. G. at W3C: 'Review of policy languages and frameworks'. Available at https://www.w3.org/Policy/pling/wiki/PolicyLangReview
[11] Parducci, B.: 'Extensible access control markup language (XACML) version 3.0, standard', 2013. Available at http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html
[12] Gusmeroli, S., Piccione, S., Rotondi, D.: 'A capability-based security approach to manage access control in the internet of things', *Math. Comput. Model.*, 2013, **58**, (5/6), pp. 1189–1205
[13] Hernández-Ramos, J. L., Jara, A.J., Marín, L., *et al.*: 'Distributed capability-based access control for the internet of things', *J. Internet Serv. Inf. Secur.*, 2013, **3**, (3/4), pp. 1–16
[14] Shelby, Z., Hartke, K., Bormann, C.: '*The constrained application protocol (CoAP), RFC 7252*' (RFC Editor, 2014), Available at http://www.rfc-editor.org/rfc/rfc7252.txt, accessed 19 December 2016
[15] Gerdes, S., Bergmann, O., Bormann, D.C.: 'Delegated CoAP authentication and authorization framework (DCAF)'. Internet-Draft draft-gerdes-ace-dcafauthorize-04, Internet Engineering Task Force (October 2015), work in Progress
[16] Park, J., Sandhu, R.: 'The UCONABC usage control model', *ACM Trans. Inf. Syst. Secur.*, 2004, **7**, (1), pp. 128–174

[17] Su, Z., Biennier, F.: 'On attribute-based usage control policy ratification for cooperative computing context'. Computing Research Repository (CoRR), 2013, abs/1305.1727
[18] Astorga, J., Jacob, E., Huarte, M., *et al.*: 'Ladon: end-to-end authorisation support for resource-deprived environments', *IET Inf. Sec.*, 2012, **6**, (2), pp. 93–101
[19] Kim, E., Kaspar, D., Gomez, C., *et al.*: '*Problem statement and requirements for IPv6 over low-power wireless personal area network (6LoWPAN) routing*' rfc 6606 (RFC Editor, 2012), Available at https://www.rfceditor.org/rfc/rfc6606.txt, accessed 19 December 2016
[20] Kohvakka, M., Kuorilehto, M., Hännikäinen, M., *et al.*: 'Performance analysis of IEEE 802.15.4 and ZigBee for large-scale wireless sensor network applications'. Proc. 3rd ACM Int. Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor and Ubiquitous Networks (PE-WASUN'06), 2006, pp. 48–57
[21] 'The M/G/1 System, Pollaczek Khinchin theorem', 2005, Available at http://www.richardclegg.org/previous/networks2/Lecture9_06.pdf
[22] Astorga, J., Jacob, E., Toledo, N., *et al.*: 'Analytical evaluation of a time- and energy-efficient security protocol for IP-enabled sensors', *Comput. Electr. Eng.*, 2014, **40**, (2), pp. 539–550
[23] MEMSIC's TelosB mote (TPR2420CA) datasheet. Available at http://www.memsic.com/products/wireless-sensor-networks/wirelessmodules.html
[24] Stallings, W., Case, T.: '*Business data communications: infrastructure, networking, and security*' (Pearson Education Limited, 2013, 7th edn.), chapter 2, pp. 57–84