Policy-controlled Authenticated Access to LLN-connected Healthcare Resources

Journal:	*IEEE Systems Journal*
Manuscript ID:	Draft
Manuscript Type:	Regular Papers
Date Submitted by the Author:	n/a
Complete List of Authors:	Rantos, Konstantinos; Eastern Macedonia and Thrace Institute of Technolog, Computer & Informatics Engineering Fysarakis, Konstantinos; Technical University of Crete, Electronic & Computer Engineering Manifavas, Charalampos; Technological Educational Institute of Crete, Informatics Engineering Askoxylakis, Ioannis; Foundation for Research and Technology - Hellas (FORTH), Institute of Computer Science
Keyword:	Health information management, Access control, Authorization, Authentication, Body sensor networks, Service-Oriented System Engineering

Policy-controlled Authenticated Access to LLN-connected Healthcare Resources

Konstantinos Rantos*, Konstantinos Fysarakis†, Charalampos Manifavas‡ and Ioannis G. Askoxylakis§
*Dept. of Computer & Informatics Engineering
Eastern Macedonia and Thrace Institute of Technology
Kavala, Greece
Email: krantos@teiemt.gr
†Dept. of Electronic & Computer Engineering
Technical University of Crete,
Chania, Crete, Greece
Email: kfysarakis@isc.tuc.gr
‡Dept. of Informatics Engineering
Technological Educational Institute of Crete,
Heraklion, Crete, Greece
Email: harryman@ie.teicrete.gr
§Institute of Computer Science
Foundation for Research and Technology - Hellas (FORTH),
Heraklion, Crete, Greece
Email: asko@ics.forth.gr

***Abstract*—Ubiquitous devices comprising several resource-constrained nodes with sensors, actuators and networking capabilities, are becoming part of many solutions that seek to enhance user's environment smartness and quality of living, prominently including enhanced healthcare services. In such an environment, security issues are of primary concern as a potential resource misuse can severely impact user's privacy or even become life threatening. Access to these resources should be appropriately controlled to ensure that eHealth nodes are adequately protected and the services are available to authorized entities. The intrinsic resource limitations of these nodes, however, make satisfying these requirements a great challenge. This paper proposes and analyzes a service oriented architecture that provides a policy-based, unified, cross-platform and flexible access control mechanism, allowing authorized entities to consume services provided by eHealth nodes while protecting their valuable resources. The scheme is XACML-driven although modifications to the related standardised architecture are proposed to satisfy the requirements imposed by limitations in the computational environment. A proof of concept implementation is presented, along with the associated performance evaluation, confirming the feasibility of the proposed approach.**

***Index Terms*—healthcare, authentication, authorization, body sensor networks, policy-based access control, XACML, DPWS, web services, security.**

I. Introduction

In recent years, we have experienced a lot of innovation in the Internet of Things (IoT) space. Collections of embedded and wearable nodes, typically bearing sensors and actuators, are becoming part of a networking infrastructure and gain connectivity to the Internet. The corresponding technologies are becoming mature enough to allow us to start looking into more advanced and comprehensive solutions that can enable these nodes to integrate smoothly with existing infrastructures, expanding, however, existing attack surfaces.

There are many application areas where these nodes flourish with even more being introduced to take advantage of the services that they can offer. Healthcare stands out as a key sector where these novel technologies and associated enhanced services can have a significant impact by improving the quality of life of patients, elderly people, but also the general population through real-time monitoring and intervention which enables proactive and more effective health management, justifying the intensive research efforts in the field [1], [2], [3].

These sophisticated nodes can be deployed as standalone devices serving a single purpose, or as part of an infrastructure that consists of nodes with similar characteristics comprising a so called low power and lossy network (LLN). Moreover, they can be used simply for monitoring various variables or for acting upon command issuance, be part of a closed system or provide advanced services to remote parties over public networks. The current trend for all these nodes is to adopt existing networking technologies and be reachable over the Internet, abandoning proprietary closed solutions. Moreover, existing networking mechanisms are updated and adapted to efficiently handle the vast population of the resource-constrained devices. Such examples are IETFs work on the 6LoWPAN [4] standards to enable IPv6 connectivity over IEEE 802.15.4 networks and research on improving the associated MAC protocols [5], [6], [7].

At the higher layers, sensor nodes and Service Oriented Architectures (SOAs) have become convergent technologies with several standards emerging from these efforts. SOAs

evolved from the need to have interoperable, cross-platform, cross-domain and network-agnostic access to devices and their services. This approach has already been successful in business environments, as web services allow stakeholders to focus on the services themselves, rather than the underlying hardware and network technologies. When deploying a SOA there are quite a few effective options to provide these services, but the Devices Profile for Web Services [8] specification stands out as it enables the adoption of a SOA approach on embedded and sensor devices with limited resources, allowing system owners to leverage the SOA benefits across heterogeneous systems that may be found in smart environments.

Whatever the deployment option and the mechanisms adopted, all these nodes are characterized by their limited resources in terms of computing power, memory, storage space, bandwidth and energy. These characteristics expose target devices to a variety of security issues [9], such as trivial attacks aiming for rapid resource exhaustion leading to Denial of Service (DoS). At the same time, studies [10] and published reports [11] reveal that current deployments have not adequately considered the threats that these nodes face when connected to the Internet, hence the lack of the security measures. The Open Web Application Security Project (OWASP) organization includes "Insufficient Authentication/Authorization" in the second place of its list of top ten security problems identified on IoT devices [12], preceded only by the use of "Insecure Web Interfaces". Such negligence is bound to inhibit any efforts made towards using these pervasive devices to handle our personal sensitive data. The expanded attack surface that results from the integration of LLNs with the Internet, needs new or adapted mechanisms to mitigate these new threats.

In the context of healthcare applications, the above security issues are exacerbated by the direct interaction with the human body and the associated safety and privacy concerns. In typical nodes used for eHealth purposes, environmental and physiological sensors are deployed for gathering all the required information depending on medical staffs prescribed needs, such as blood pressure and body and room temperature. On top of that, actuators controlled by authorized medical staff can also be deployed, such as an automatic insulin injection device used for remote treatment. Such sensitive actions, i.e. reading and issuing commands, need strict access control decisions before being authorized so that users privacy and even safety are not jeopardized by unauthorized actions.

Motivated from the above, this work proposes an architecture that allows authorized entities to access the services provided by resource-limited eHealth nodes. The scheme provides flexibility in terms of the authentication mechanism used, that is to say that the service requester can be authenticated using e.g. username/password, certificate, or other authentication methods. Among the main concerns of the proposed architecture are the nodes' protection from unjustifiable use of their resources and the need to be able to control access through a well-established set of policy rules that can change and adapt to new environmental parameters. The work builds upon the eXtensible Access Control Markup Language (XACML) [13] model for policy based access control infrastructures, proposing certain modifications to satisfy requirements stemming from the limited resources of nodes, and the adoption of lightweight SOA mechanisms, through the use of the DPWS, for entity interactions. Such a limited resources device is an eHealth node that a user possesses and can provide useful healthcare services to medical staff and other stakeholders. Although mainly a framework, the main components of the proposed architecture have been implemented by the authors, and results are provided here as a proof of concept.

This paper is organized as follows: Section II presents the technical background and relevant research efforts identified in the literature. Section III includes specific use cases to highlight the rationale behind this work, while Section IV lists the essential requirements identified during the design phase. Section V presents the proposed architecture in detail, while sections VI and VII detail the approach followed to implement the framework's entities and their performance evaluation respectively. A discussion on the security issues that must be considered to safeguard the framework's operation can be found in Section VIII. Finally, Section IX features concluding remarks and pointers to future work.

II. Background and related work

On the communication level, all efforts are towards the integration of low power and lossy networks (LLNs) with existing networking technologies to provide internet connectivity and realize the so called Internet of Things. Several solutions have emerged through this process with their particular advantages, disadvantages and properties. Most of them have provided their own standards and specifications and have helped formulate antagonistic technologies. They might differ on various layers of the TCP/IP stack, such as the physical and the network layer or on the upper layers, i.e. presentation and application layers. Regarding the former we have technologies like open standards 6LoWPAN and ZigBee (which is free for non-commercial purposes), proprietary provided under a license like Z-Wave, and alternatives like Bluetooth and Wifi usually met in other environments. It is not in the scope of this paper to name all these technologies and provide a comparative analysis. The proposed solution focuses on the architecture level and on the upper layers of the TCP/IP stack, thus making this solution underlying protocol independent. 6LoWPAN seems to outweigh other technologies given its Internet connectivity orientation which provides many benefits to adopting solutions.

On upper layers of the TCP/IP stack, protocols provide methods to exchange structured or unstructured messages that facilitate (secure) service access. Data are typically encapsulated in standardised protocols that allow the seamless exchange of messages between nodes and remote entities, outside the LLN boundaries, even if this is accomplished through the use of a bridge and/or router. Among the technologies being used are the service-oriented ones with several schemes being used for the way these services are provided and how a service consumer can access them. Standardisation and research efforts in the area of Service Oriented Architectures have been taking place for more than a decade and schemes have

been proposed and standardised regarding service discovery, registration, access and protection, and the corresponding communication protocols that enable the interoperable exchange of messages among remote participating entities. While in some cases efforts focus on adapting existing technologies to the constrained environment provided by such devices, other initiatives target for the introduction of new mechanisms specifically designed for such environments, without however neglecting interoperability with existing Internet technologies.

Such a scheme is the Devices Profile for Web Services (DPWS) [8], a profile of Web Services protocols that enables Web Service messaging, discovery, description, and eventing on resource-constrained devices. DPWS messages are typically encapsulated in SOAP (Simple Object Access Protocol) envelopes and transported over any transport protocol, including HTTP and UDP using the SOAP-over-HTTP and SOAP-over-UDP bindings respectively [14], or even an SMTP binding.

Another option is the use or a RESTful environment based on Constrained Application Protocol (CoAP) [15]. Both SOAP and REST are protocols used to access Web services but they are based on completely different technologies. SOAP is an XML-based standardised protocol which, although has been criticized for its complexity stemming mainly from the inherent structured information exchange, it has more power and enjoys extensibility. REST is the lighter and more flexible alternative where one can get the necessary information even by using the URL approach.

Among the above solutions, the authors chose to adopt the DPWS specification, which can enable user-to-machine and machine-to-machine interactions in a unified manner, moving on from the current state of the field, where manufacturers offer a variety of proprietary protocols which are not interoperable and essentially lock-in users, forcing them to use a specific vendor/ecosystem.

In terms of the way that access to these services is controlled, the eXtensible Access Control Markup Language (XACML) [13], provides an access control language and a model for processing requests to resources while the Security Assertion Markup Language (SAML) focuses on the way the requester is authenticated and assertions are being transferred among participating entities. WS-Trust is another web services oriented model that defines how security tokens are being issued, renewed and validated (WS-Trust).

Many access control schemes have been proposed for wireless sensor networks, yet most of them focus on authentication and authorization schemes and on enhancing basic access control models to address privacy matters. Such schemes can be found in [16], [17], [18], [19]. Little work has been carried out on policy-based access control (PBAC). The EU-funded research project Internet-of-Things Architecture (IoT-A) worked on the adoption of XACML in the Internet of Things [20] and proposed a generic model whose functional modules are mapped to a set of well-defined components that comprise the IoT-A. The authors use a logistics scenario for demonstration purposes.

In [21] the authors also utilize XACML but focus on the privacy of eHealth data within the mobile environment. In contrast to the work presented here, a complete framework is not included and, moreover, the authors choose computationally intensive security mechanisms such as XML encryption digital signatures. In [22], the authors propose a lightweight policy system for body sensors but they do so by presenting a custom API and policy definitions, thus sacrificing interoperability with existing standards and infrastructures.

Santos-Pereira et al [23] focus on enforceable security policies for systems interoperability and data exchange between healthcare entities. The authors present a Role-based Access Control mobile agent model, using public key infrastructure for authentication and access control, but the proposed scheme is presented at design-level, lacking implementation details and a performance evaluation.

This paper focuses more on the area of securing access to heterogeneous resources through policy-based access control, hence it utilises the SOA-based and XACML-related standards, while proposing certain modifications, detailed below, to better fit to the restricted environment of LLNs. This approach allows leveraging work already carried out on XACML policy definitions, but also Web Services as mentioned above.

With regard to the former, the "Cross-Enterprise Security and Privacy Authorization Profile of XACML v2.0 for Healthcare" [24] constitutes important background work, compatible and in-line with the scheme proposed in this paper. This OASIS profile specifies the use of XACML to promote interoperability within the healthcare community by providing common semantics and vocabularies for interoperable policy request/response, policy lifecycle, and policy enforcement.

The benefits of adopting a SOA-based approach come in the form of increased usability and interoperability. While typical XACML deployments require the setup of complex infrastructures to enable entities' interaction and policy retrieval (e.g. via the Lightweight Directory Access Protocol, LDAP [25]), the proposed framework leverages the benefits of DPWS. This allows the deployment of devices aligned with the Web Services technologies, thus facilitating interoperability among services provided by resource-constrained devices, facilitating seamless discovery and interactions among entities, and allowing the deployment of the framework's entities to any platform, anywhere on the hospital or home network, with minimal involvement on behalf of the user.

III. Applicable Scenario

Before moving into the presentation of the proposed architecture, it would be good to demonstrate through specific scenarios, the incentives behind this work that have also formulated the requirements defined below. The proposed scheme addresses the main need to be able to remotely access data collected by sensors and control actuators deployed in a LLN. The architecture utilizes service oriented technology to be able to provide services to remote authorized parties where access restrictions are imposed through policy rules. This typically means that access is not necessarily restricted to entities of a closed system. Such an architecture fits perfectly to a Body Sensor Network (BSN) deployment [26], [27], which actually inspired this work, and which we use here to demonstrate

the architecture's applicability and the way that policy based access control SOAs are envisaged.

Let's assume that a patient has multiple medical sensors and/or actuators deployed to monitor and/or control his/her medical condition. Sensors and actuators typically reside on nodes with very limited processing power and capabilities, namely nano nodes. These can communicate and register with a mobile device that the user has in possession, such as a mobile phone or tablet. An application running on this mobile device actively monitors sensor's readings and, if necessary or appropriately instructed, forwards these data to authorized medical staff. Alternatively, the data could be given to medical staff not as a result of an alert, but as a response to a request issued by this staff.

Now consider the case where in the context of telemedicine or in case of an accident as well as for many other medical reasons, other people not previously registered with the user's application, need to gain access to those readings and actuators. For example, a patient is involved in an accident, which is reported to the emergency services, and some readings have to be taken to validate his medical conditions while emergency services are on their way to the accident scene.

In this case, the requester, i.e. doctor, emergency services staff, will request remote access to the readings of these sensors or even issue commands to the actuators. Without loss of generality, we can claim that in the eHealth environment, as with any other environments, services might need to be accessed occasionally, depending on the patient's health condition, and on a need to know basis. Therefore, several questions arise that have to be addressed in the proposed architecture, mostly related to patient's privacy and life protection.

- Who is eligible to access this information?
- How do we authenticate a user that has not been registered with the specific service in the past?
- How is the legitimacy of his/her request evaluated?
- Who and how is going to decide about this requester's privileges?

In our scenario we consider that the medical staff can look in a central repository for the types of services provided by the patient and can request access to them. This is checked against applicable policies that the user in conjunction with medical staff and/or national insurance and/or insurance company and/or applicable law have defined. If access is granted the request is forwarded to the patient's device and the requested information is disclosed or access to the actuator is permitted. As a result, the requester will be able to have sensor readings, e.g. patient's heart rate, and/or act remotely, e.g. inject an altered dose of insulin.

IV. Requirements

Access control is very important for protecting the sensitive resources of a BSN, which can affect human lives. Among the requirements that have to be satisfied are the following [28], [29]:

- Data confidentiality: Access to medical data should only be allowed to authorized parties, such as medical staff. Note that unauthorized disclosure of medical data while in transit is also a protection requirement.
- Message authentication: Commands issued to actuators must be authenticated to avoid unauthorized execution.
- Availability: Data must remain available to authorized entities, such as medical staff, while access to them must not be denied due to wrong decisions.

IP based networking in LLNs changes the way that participating nodes can be accessed and their respective services can be consumed. For instance, there is no need for a dedicated application server that will intervene between a node and a remote party that wants to access the node's resources [30]. However, one of the problems that these nodes face in such a deployment, is that they have limited resources which do not suffice for the deployment of strong protection mechanisms. Without those mechanisms however, nodes are exposed to direct access from the Internet without having the capacity to handle unlimited requests. Therefore, several issues arise regarding the protection of nodes resources, that have to be addressed. The main aim is to protect the limited resources of a node that implements a service oriented architecture, to provide access to data and mechanisms that the node has under control.

Within this context, the proposed architecture is designed to satisfy the following requirements:

- Provide services using of Service Oriented Architecture technologies;
- Provide fine-grained access control to nodes' resources;
- Authenticate remote entities wishing to access protected nodes resources;
- Control access to nodes' resources through well-defined policies;
- Protect sensitive nodes from unauthorised access and unnecessary consumption of valuable resources including network and energy;
- Secure the channel between the participating nodes to provide message confidentiality, integrity and authentication;
- Comply with existing standards to satisfy interoperability among the participating entities, such as between the identity provider chosen by the requester and the service orchestrator, regarding the exchange of authentication messages, assertions or user metadata and attributes.

In the following section we describe the proposed architecture that satisfies the above.

V. Proposed Architecture

The architecture proposed in this paper is an enhanced policy based access control scheme that seeks to provide flexibility regarding the chosen authentication mechanism while satisfying the aforementioned requirements, typically imposed by nodes' resource limitations. For this purpose, certain modifications to the OASIS standardised policy-based access control scheme are proposed to accommodate these needs.

The scheme utilizes and seeks compliance with the following technologies:

- XACML: an XML-based OASIS standard that defines a policy and an access control decision request/response language. An XACML-based architecture typically consists of the following main components:
  - *Policy Enforcement Point (PEP):* Performs access control, by making decision requests and enforcing authorization decisions [13], [31].
  - *Policy Decision Point (PDP):* Evaluates requests against applicable policies and renders an authorization decision [13].
  - *Policy Administration Point (PAP):* Creates and manages policies or policy sets [13].
  - *Policy Information Point (PIP):* Acts as a source of attribute values [13].
- SAML 2.0 specification to protect, transport, and request XACML schema instances and other information needed by an XACML implementation [32]. Note that although SAML can be used to convey authorization decision statements, this functionality in SAML is intentionally restricted compared to the more flexible XACML solution, hence the adoption of XACML and the use of SAML for encapsulating XACML messages.

In the XACML data-flow model defined in the OASIS standard the PEP, via the context handler, is considered as the device that orchestrates the exchange of messages among the requester, the PDP, the Attribute Authority and the Attribute Repository. According to the XACML specifications the PEP is considered as "part of a remote-access gateway, part of a Web server or part of an email user-agent, etc". Therefore all initial requests, valid or not, are sent to the PEP which will act as a routing device between the requester and the back-end key entities that examine the requests and make decision based on policy rules and other parameters, such as the requester's and/or resource's attributes.

While this model is appropriate for typical application gateways, it cannot be considered as such for resource-constrained nodes that only have the capacity to accept requests from a limited number of clients. Beyond this threshold, valuable node resource consumption is not acceptable as it leads to battery drainage and service unavailability. In this context, resource-constrained devices have to participate in the decision making process only if absolutely necessary and only to authorized entities to save valuable resources. As such, they cannot assume the role of a PEP as this is defined in the XACML standard [13].

Moreover, the flow model currently defined by XACML, considers that the PIP has all the required attributes for the requester, and that the PDP gets all the information from the PIP, which might be queried twice for the required attributes, once from the PEP and once from the PDP. Use of specific PIP implies that services will only be provided to entities subscribed to the specific scheme, thus narrowing down flexibility. This is in contrast to a more flexible approach where services are offered to a broader group of users, subject to policy restrictions.

The proposed architecture is depicted in Figure 1. In this proposal we assume that nodes bearing sensor and actuators, expose their functionality as web services. This can either be done through the device that the node is attached to, e.g. a mobile device, or directly by the node, assuming that it is powerful enough to accommodate such functionality. All these nodes are part of a dispersed environment where there is not necessarily a single gateway or web server to assume the role of PEP as this is defined in the XACML standard. Besides that, the service owner might want to register these services with multiple servers. As a result, the PEP functionality cannot be assigned to a gateway but it should be on the device that exposes this functionality, i.e. the mobile device, a wearable node etc. For a given PEP, one of these web servers is assumed to play the role of the orchestrator as described below.

The core component of the proposed scheme is the Service Orchestrator (SO) which acts as a proxy for certain operations, such as relaying queries and messages exchanged among participating entities, yet not for handling the information the PEP exchanges with the requester.

Initially, the node, which assumes the role of a PEP, registers its services, defines the connection point to be the SO and sets the policy rules for its resources. This is accomplished once during the set-up phase. Following that, the data flow of the proposed architecture includes the following steps:

- A requester, who wants to access the service, formulates an appropriate request based on the advertised service rules, and sends it to the SO (step 1a). Note that this is in contrast to the XACML specifications which opted for sending the request directly to the PEP, introducing significant overhead that a limited-resources device cannot handle.
- The SO forwards the request to the PDP (step 1b) which, based on the requested target, fetches all applicable policies from the PAP (step 2) and informs the SO about the needed user attributes (step 3). As a result, the SO presents a list of approved Identity Providers (IdP) for the requester to authenticate (step 4).
- The requester chooses the appropriate IdP and the SO issues a (signed) authentication request (`<AuthnRequest>`) together with an attribute query (`<AttributeQuery>`) [32] to the chosen IdP. Upon successful authentication (step 5) the requester consents for the disclosure of certain attributes that the SO requires. Note that the IdP might be an entity that operates within the same environment as the SO. The actual authentication method used by the IdP is outside the scope of this paper.
- The IdP formulates a proper assertion for the necessary attributes and sends it to the SO via the Requester (step 6a). As a result, the SO forwards the received assertion to the PDP (Step 6b) [33].
- The forwarded assertion allows the PDP to establish a security context by combining the supplied attributes with the applicable policy rules which the PDP obtained from the PAP (step 2). Note that additional policy rules, might be obtained at this point (step 7), based on the requester's attributes. The typical XACML decision making process can take place during this step.
- The access decision is sent to the SO (step 8). If the

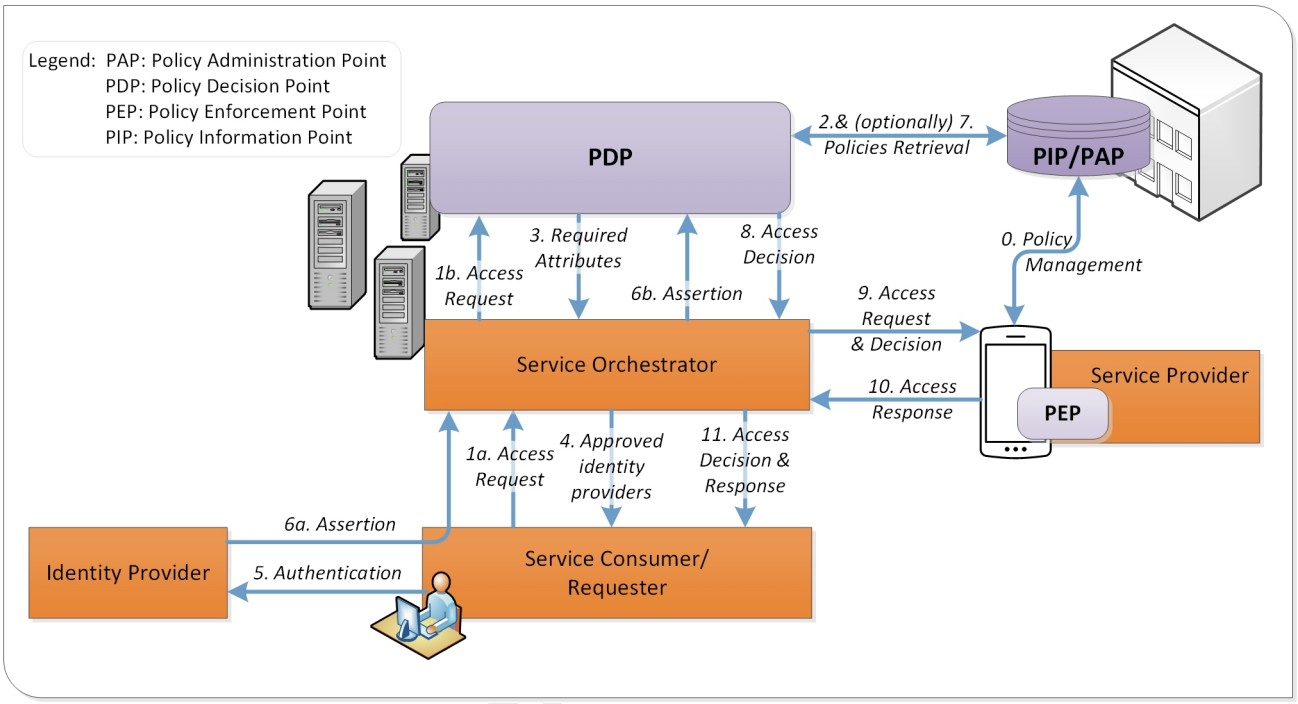

Fig. 1. Authenticated Access Control for LLNs

decision is to grant access, a signed or MAC-protected ticket is forwarded to the PEP together with details about the request (step 9). This is the first time that the node is contacted, and is only performed by an authorized party, hence not exposed to the outside world. If access is denied the decision is simply forwarded to the Requester. The Service Provider might also be informed on that based on appropriate pre-configurations.

- Now the PEP can respond to the service request through the SO (step 10). The SO can in turn send to the requester the Access Decision and the response to the Access Request. The Access Decision can be used as a token for re-accessing the same service without undergoing the authentication process.

The framework can trivially be expanded to cater for the joint operation of two or more access control infrastructures (i.e. PDPs and corresponding PIPs/PAPs). This can be used as a means to consolidate the requirements of different stake-holders and their active policy sets. In such a case, the SO can query all the different PDPs and provide or deny access based on pre-defined simple rules (e.g. only in cases where all PDPs explicitly allow such access). So, for example, someones request to access the patients blood sugar levels will only be forwarded to the pertinent medical device if both the patient and the attending doctor have authorized the specific individual to perform such an action.

VI. Implementation Approach

There are many open-source implementations of the XACML handling and decision-making process that can be utilized for the proposed architecture. The authors chose Suns XACML [34] for this implementation, as it remains popular among developers and is actually the basis of various current open source and commercial offerings.

All of the frameworks entities are implemented and their interfaces exposed using DPWS. This facilitates the discovery and description of the devices involved, also offering control and eventing mechanisms which assist in the communication of the necessary information among the entities. Web Services for Devices (WS4D) [35] is an open source initiative which provides a number of toolkits for various platforms. The authors APIs of choice is the WS4D-JMEDS (Java-based) [36] stack as it is the most advanced and active work of the WS4D initiative, supporting almost all of the existing DPWS features and providing portability to a wide range of platforms.

The approach adopted to protect the messaging of the proof of concept implementation is the use of the mechanisms detailed in the Web Services Security Specification (WS-Security or WSS, [37]). WS-Security is part of the WS-* family of specifications published by OASIS, in-line with most of the other standardised approaches adopted by the proposed framework and the one typically used alongside DPWS. The protocol specifies integrating security features in the header of SOAP messages. Working in the application layer ensures the end-to-end integrity and confidentiality of SOAP messages.

The exact implementation of the framework's entities and their communication interfaces depicted in Fig. 2 are detailed below.

Service Orchestrator to Policy Decision Point: The SO is implemented as a DPWS peer (i.e. both a client and a server). Other than the necessary mechanisms needed to interface with the approved identity providers (which will vary depending on the specific scenario/deployment examined), it also features an "Attribute_Requirements" operation. Similarly, the PDP has an

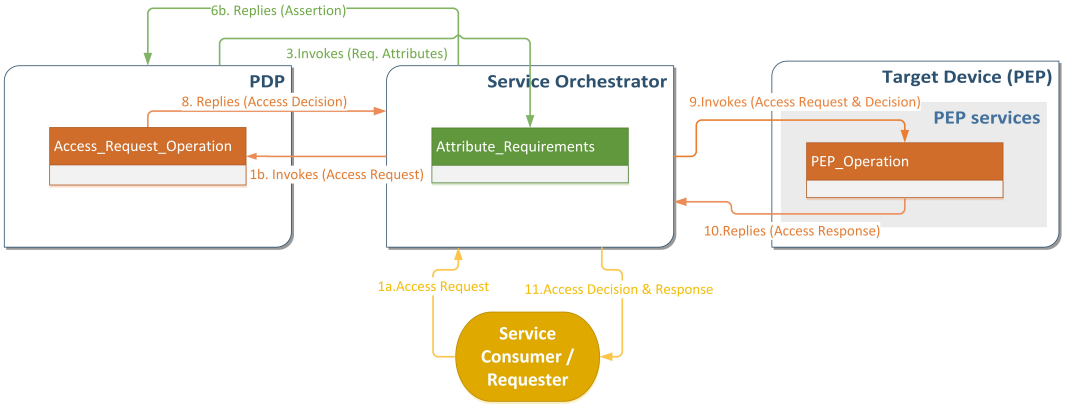

Fig. 2. DPWS-based implementation of the authentication scheme

"Access_Request_Operation". The latter is invoked by the SO as soon as an access request arrives from a service consumer, relaying the request for evaluation. As soon as the XACML decision-making process is completed, the PDP replies to the invocation with its access decision. As detailed in the information flow above, prior to providing a decision, it may need to invoke the "Attribute_Requirements" operation on the SO, in order to inform it of the needed user attributes, getting the proper assertion as an answer.

Service Orchestrator to Policy Enforcement Point: The Policy Enforcement Point must reside on every device with resources that must be protected from unauthorized access. Other than the functional elements of the devices which the framework intends to protect (e.g. access to its sensors), one extra operation must be present on each DPWS device, namely the "PEP_Operation". The SO, acting as a client, invokes this operation providing the service consumers access request along with the decision (pre-issued by the PDP) as input. If the decision accompanying the invocation is positive, the PEP replies to the SO with the resource (e.g. temperature reading) that the service consumer originally tried to access. This information is then relayed to the service consumer/requester. The above DPWS-based communication mechanisms are depicted Fig. 2.

VII. Performance Evaluation

The platform-agnostic nature of SOAs enables the proposed framework to be deployed, by design, on a variety of platforms and operating systems. However, in order to realistically assess the performance of the proposed framework, the developed entities had to be deployed on devices expected to be present in healthcare deployments. Therefore, the proposed framework was implemented and its performance was evaluated on a heterogeneous environment, featuring relatively resource-constrained embedded platforms as well as desktop computers.

The PEP-equipped target device (i.e. the device providing the actual service to be accessed) was on a Beaglebone [38], a low-cost credit-card-sized embedded device that runs a compact Linux-based operating system. It uses an ARM Cortex-A8 single core CPU running at 720MHz (throttled at 500MHz during testing) with 256MB DDR2 RAM. The testbed for the Service Orchestrator was a similar but slightly more powerful and versatile Beagleboard-xM [39] embedded platform, featuring an 1GHz ARM Cortex-A8 processor (throttled to run at 600MHz during testing) and 512MB DDR2 RAM, also running a minimal Linux-based operating system. The access control infrastructure entities, i.e. the PDP and PIP/PAP, were deployed on a desktop system (Core i5 CPU at 3.3GHz, 8GB DDR3 RAM). An identical desktop system was used to run the service consumer, a client application programmed to automatically invoke the resources exposed by the SO and record response times, for benchmarking purposes.

Tests also included a second scenario where an extra PDP and PIP/PAP were deployed on a more resource-constrained platform, namely a Beaglebone embedded device, like the one used for the target device (i.e. the PEP). The latter was used to investigate the performance impact when the SO has to query two different PDPs, each with its own policy set, to emulate the use case where e.g. the patient and the hospital each have their own access control infrastructure and policy requirements. In this scenario, the SO had to evaluate both responses and only allow the user to access the resources if both PDPs allowed such access.

The test setup described above is depicted in Fig. 3. Note that this setup is by no means the only option for the proposed framework's deployment. For instance, a Beaglebone was chosen for the SO to simply demonstrate the ability of the SO to be deployed even in a constrained environment of an embedded system. In a large-scale deployment one would expect the functionality of the SO to be deployed at an application server to ensure the system is able to serve a sufficient number of users.

Aiming to also assess the performance impact in situations where the messages exchanged would have to be secured, an alternative proof-of-concept implementation was developed adopting the security mechanisms specified in WS-Security. These mechanisms safeguard the integrity and confidentiality of the policy messaging exchanged by the frameworks entities.

The application profiling (i.e. CPU and memory utilization) was focused on the Service Orchestrator, which is the main entity of the proposed approach, and on devices which are expected to have resource limitations, i.e. the PEP-equipped target device. Moreover, the impact on user experience was

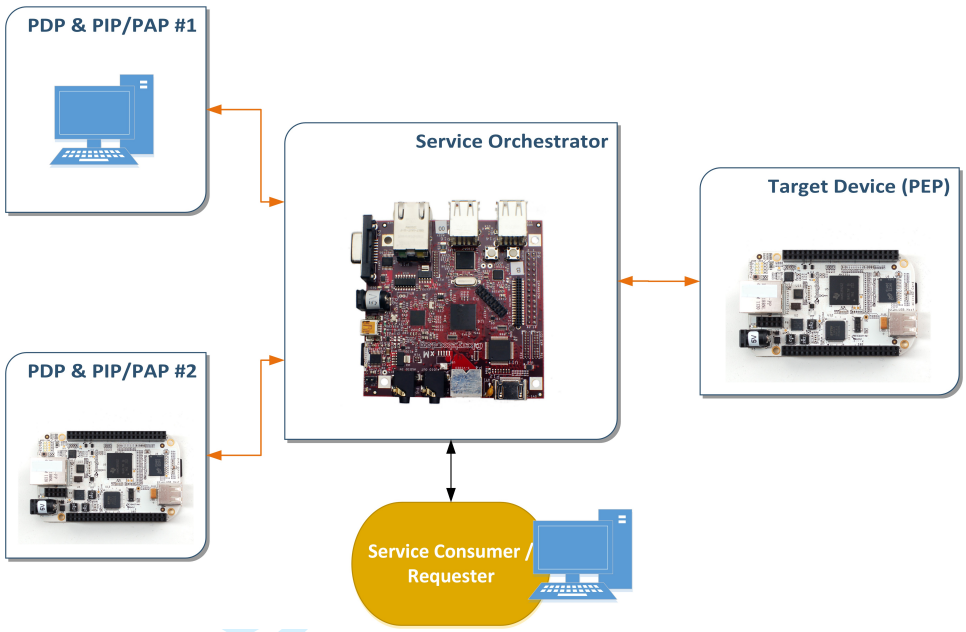

Fig. 3. The test-bed setup, featuring embedded devices and desktop PCs. Orange lines indicate communication where WS-Security is optionally enabled. Also depicts the extra PDP & PIP/PAP introduced in the second test scenario.

also assessed, by recording client-side response times in all usage scenarios.

The steps related to the Identity Provider were omitted during testing, as these will vary depending on the Identity Provider that the user will choose and are deployment-specific, thus out of the scope of the framework presented in this work.

A total of 100 consecutive requests were issued from the service consumer application to the SO residing on the Beagleboard-xM. The response time recorded by the test client trying to access the target devices resources appear in Fig. 4. The WS-Security mechanisms impose a significant overhead to the response times, which is expected given the use of asymmetric cryptographic mechanisms. In contrast, the response times for the second scenario indicate that the introduction of a second instance of the PDP and PIP/PAP is not prohibitive, while allowing to consolidate the policy requirements of different stakeholders.

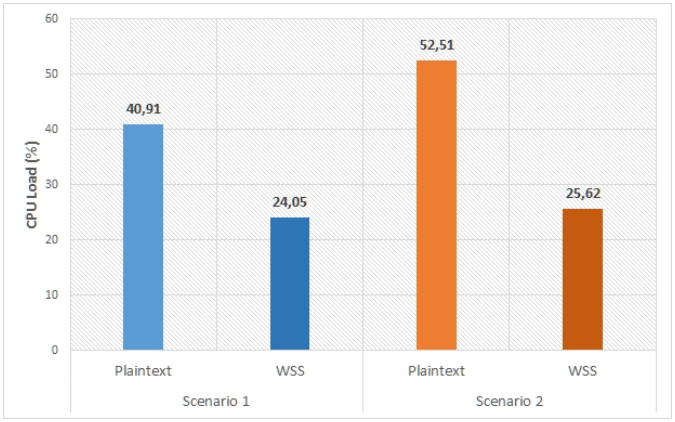

Fig. 5. Service Orchestrator's average CPU load (%).

Profiling of the SO revealed a lightweight application, even under the load of consequent requests, or in the presence of two PDP and PIP/PAP instances. The average CPU load and memory consumption appear in Fig. 5 and Fig. 7 respectively. As the occupied memory remains constant irrespectively of the presence of one or two PDPs, the numbers for the second scenario are omitted. The use of WSS imposes a relatively small memory overhead, while the average CPU load drops, as the device has to wait more between requests, due to the network and processing overhead on other framework entities.

The same behaviour with regard to CPU load was also recorded on the target device (i.e. the device featuring the PEP), as is depicted in Fig. 6. As in the case of the SO, introducing the WSS mechanisms increases the memory footprint (appearing along with SO values in Fig. 7), but the latter, along with CPU load, are not significantly affected by the presence of multiple PDPs and the corresponding PIP/PAPs, thus the numbers of the second scenario are omitted from the corresponding figures.

VIII. Security Analysis and Considerations

One of the main concerns in accessing services and issuing commands, is the protection of the data being exchanged among the participating entities. In the proposed scheme the service provider has a pre-established relationship with the SO, PDP and PAP. Note that all these three entities are only functional components and therefore the exact needs in secure channel establishment depend on the actual deployment choice and cannot be specified. In a simplified approach, the SO, PDP and PAP can be part of the same entity and therefore a secure channel establishment using pre-shared keys is a viable and efficient option.

Regarding the underlying message security mechanisms, there are a number of proposed or standardised schemes that handle the protection of messages at various layers of the

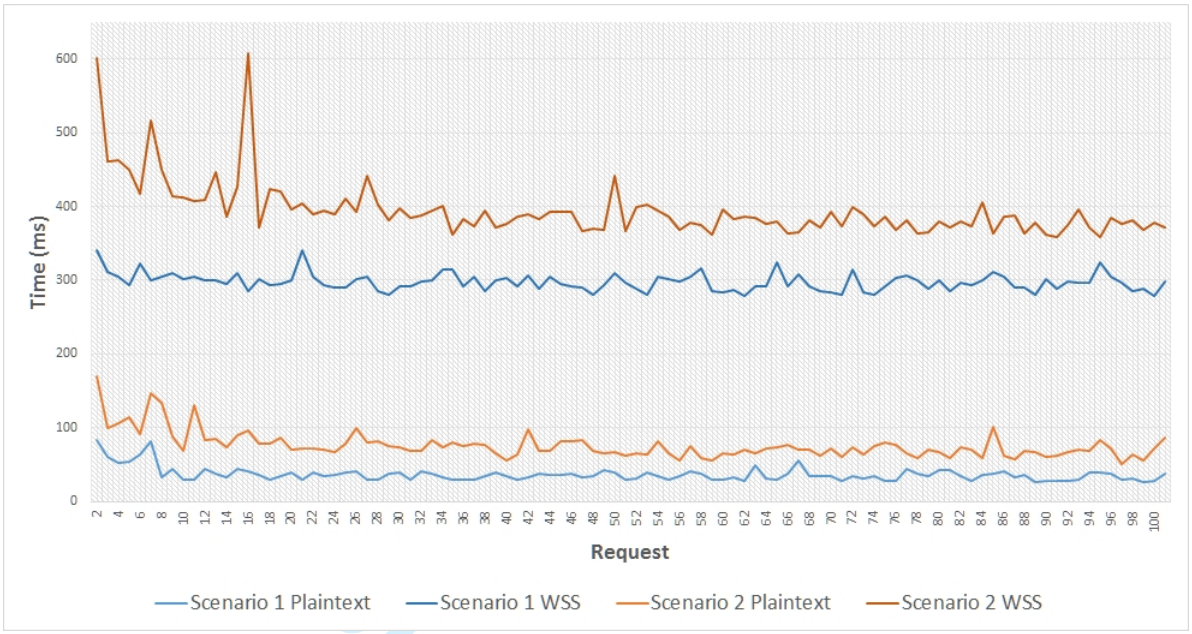

Fig. 4. Client-side response time for 100 requests to the Service Orchestrator.

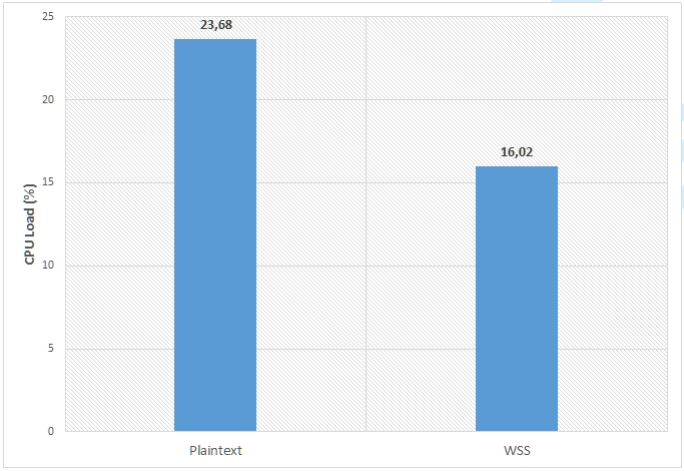

Fig. 6. Target devices CPU load (%) for both scenarios.

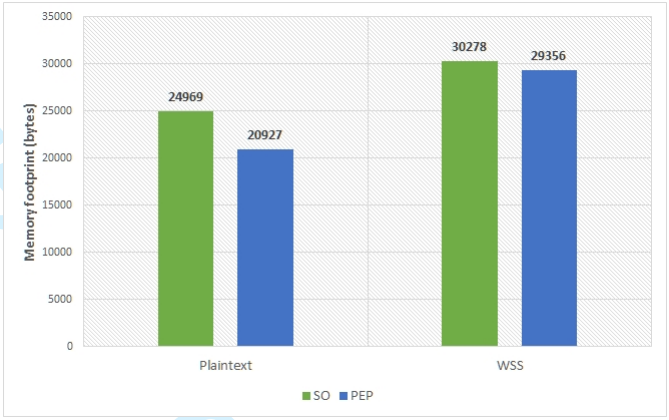

Fig. 7. Service Orchestrators and Target device's memory utilization (in bytes) for both scenarios.

network stack. The WS-Security mechanisms adopted by the authors for the proof of concept implementation is typically used alongside DPWS, but its public-key security primitives can impose a significant performance overhead, as is evident from the performance evaluation presented in the previous section. Therefore, considering the resource-constrained nature of some devices, and the need to minimize performance impact in general, alternative cryptographic primitives can also be investigated for production environments.

Some prominent alternative schemes protect messages at the application or network layer and can provide end-to-end message protection. Well-known security mechanisms for these layers are the TLS (Transport Layer Security) [40] protocol and its counterpart proposed for securing UDP messages, namely DTLS [41]. Other schemes focus on efficiently providing authenticated encryption, like the Identity-based Cryptosystem (IBC) signcryption mechanisms presented by Fagen et al. [42]. Related to the above is the relatively novel concept of security fusion, whereby weak point-to-point properties are combined in order to produce strong security properties in a resource-aware manner [43].

At the lower layers, as existing networking mechanisms are updated and adapted to efficiently handle the vast population of the resource-constrained devices (e.g. work on the 6LoWPAN), the pertinent cryptographic primitives are also adapted and improved accordingly. Such an example is the IPsec protocol and its variants that utilize header compression [44], [45], [46], which can provide similar levels of protection while preserving the valuable node resources.

It is expected that some of the framework's entities will be deployed on normal, relatively powerful nodes (personal computers or even servers). Thus, e.g. the link between the Requester and the SO could alternatively be protected

using common methods, like TLS, the same way that the communication channel between the Requester and the IdP is anticipated to be protected, although the latter is outside the scope of this paper. The cost of using TLS, however, between the Requester and the SO is that the secure channel breaks at the SO and the SO has to re-encrypt the communication using the security parameters set for the link between the SO and the service provider.

The actual authentication scenario could be further elaborated during deployment to match system owner's specific requirements and trust relationships with identity providers. Several options in such a deployment exist as they have been demonstrated in [47].

The proposed scheme provides the Service Provider the flexibility to change the orchestrator(s) it uses based on its needs. This also applies to applicable policy rules which the service provider can modify to match his/her requirements. As an example, consider the situation where the owner of the mobile device being used to offer these services, changes mobile operator. He/she simply has to change SO, to a platform operated by the new mobile operator, and register his/her policies with it. Use of the SO provides additional benefits which are related to the node's connectivity. The node can wake up occasionally to fetch any requests sent to the SO. This approach also helps save node's resources, as no requests are sent to the node unless the latter asks for it. If the service request was sent directly to the PEP, the corresponding device would have to always be online, otherwise the service would be unavailable.

IX. Conclusions

As computing becomes ubiquitous, adopters aim to exploit the potential of pervasive systems, including LLN nodes bearing sensors and actuators, in order to introduce new types of services and address inveterate and emerging problems, healthcare being one of the most prominent application. Nevertheless, a key factor in the wide adoption and success of these new technologies is the effectiveness with which the various security and privacy concerns are tackled within the resource-constrained environment.

To this end, this paper proposes an architecture for providing robust authenticated access control to heterogeneous resource-constrained devices. The scheme builds upon the standardized technologies, namely access control mechanisms based on XACML and SOA-based interfacing of its key entities. In contrast to typical XACML deployments, the core PEP functionality and the hosting resource-constrained device are efficiently relieved from the expensive computations that the XACML standard defines, without sacrificing any of the policy-based decision making process. The device is sheltered from direct user interaction, helping alleviate concerns that are typical to resource-constrained devices, like DoS attacks. Emphasis was given on the scheme's ability to serve users authorized by, typically, any authentication scheme, thus enabling the large-scale deployment of the solution to many environments.

An important parameter regarding the efficacy and applicability of such a scheme is the communication mechanism adopted to implement the interaction between the frameworks entities. The XACML architecture does not define the exact communication mechanisms to be used by its entities, but industry and researchers alike have demonstrated the potential for significant benefits from the adoption of a SOA-based approach on the various heterogeneous embedded devices that permeate smart environments. DPWS is a standardized specification that enables the bridging of SOAs with resource-constrained systems, and was thus the technology that the authors chose for interfacing the various entities that form the presented framework.

As a proof of concept, the components of the proposed scheme were developed and deployed on a heterogeneous testbed featuring desktop systems and typical embedded devices. The performance overhead imposed on the three most important endpoints, i.e. the client attempting to access the protected resources, the Service Orchestrator and the PEP, was analyzed and presented to demonstrate the feasibility of the suggested solution.

An important aspect to be investigated in future work is the on adapting and potentially extending XACML policies to consolidate the requirements introduced in the new IoT reality (where, e.g. semantics are often utilized to provide context-awareness [48] and where spatio-temporal factors have to be considered, due to the constant mobility of users and their devices [49]) with healthcare requirements, as defined in the relevant specifications (e.g. [24]). The development of more lightweight DPWS implementation should also be pursued, allowing the integration of extremely resource-constrained devices, like expendable body sensors, into the proposed framework. This work will have to be carried out concurrently with the investigation of lightweight cryptographic primitives appropriate for said devices and the communication mediums they typically use.

Acknowledgment

This work was partially supported by the Greek General Secretariat for Research and Technology (GSRT), under the ARTEMIS JU research program nSHIELD (new embedded Systems arcHItecturE for multi-Layer Dependable solutions) project. Call: ARTEMIS-2010-1, Grant Agreement No.:269317.

References

[1] C. Rcker, M. Ziefle, and A. Holzinger, "From computer innovation to human integration: Current trends and challenges for pervasive healthtechnologies," in *Pervasive Health*, ser. HumanComputer Interaction Series, A. Holzinger, M. Ziefle, and C. Rcker, Eds. Springer London, 2014, pp. 1–17. [Online]. Available: http://dx.doi.org/10.1007/978-1-4471-6413-5_1

[2] H. Kielland Aanesen and J. Borras, "ehealth: The future service model for home and community health care," in *Digital Ecosystems and Technologies (DEST), 2013 7th IEEE International Conference on*, July 2013, pp. 172–177.

[3] S. Tennina, E. Kartsakli, A. Lalos, A. Antonopoulos, V. Mekikis, M. Di Renzo, Y. Z. Lun, F. Graziosi, L. Alonso, and C. Verikoukis, "Wsn4qol: wireless sensor networks for quality of life," in *Proceedings of the IEEE 15th International Conference on e-Health Networking, Application & Services (HealthCom13)*, 2013.

[4] G. Mulligan, "The 6LoWPAN architecture," in *Proceedings of the 4th workshop on Embedded networked sensors - EmNets '07*, 2007, p. 78. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1278972.1278992

[5] B. Otal, L. Alonso, and C. Verikoukis, "Highly reliable energy-saving mac for wireless body sensor networks in healthcare systems," *Selected Areas in Communications, IEEE Journal on*, vol. 27, no. 4, pp. 553–565, May 2009.

[6] ——, "Towards energy saving wireless body sensor networks in health care systems," in *Communications Workshops (ICC), 2010 IEEE International Conference on*, May 2010, pp. 1–5.

[7] S. Ullah and K. S. Kwak, "Performance study of low-power mac protocols for wireless body area networks," in *Personal, Indoor and Mobile Radio Communications Workshops (PIMRC Workshops), 2010 IEEE 21st International Symposium on*, Sept 2010, pp. 112–116.

[8] "Devices profile for web services, version 1.1," 2009. [Online]. Available: http://docs.oasis-open.org/ws-dd/ns/dpws/2009/01

[9] S. Saleem, S. Ullah, and K. S. Kwak, "Towards security issues and solutions in wireless body area networks," in *Networked Computing (INC), 2010 6th International Conference on*, May 2010, pp. 1–4.

[10] A. Cui and S. J. Stolfo, "A quantitative analysis of the insecurity of embedded network devices: Results of a wide-area scan," in *Proceedings of the 26th Annual Computer Security Applications Conference*, ser. ACSAC '10. New York, NY, USA: ACM, 2010, pp. 97–106. [Online]. Available: http://doi.acm.org/10.1145/1920261.1920276

[11] HP, "Internet of Things Research Study," Tech. Rep., 07 2014. [Online]. Available: http://h20195.www2.hp.com/V2/GetDocument.aspx?docname=4AA5-4759ENW&cc=us&lc=en

[12] "Internet of Things Top Ten Project." [Online]. Available: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Top_Ten_Project

[13] B. Parducci, H. Lockhart, and E. Rissanen, "eXtensible Access Control Markup Language (XACML) Version 3.0," 2013. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf

[14] T. Nixon, A. Regnier, and R. Jeyaraman, "SOAP-over-UDP Version 1.1," pp. 1–20, 2009. [Online]. Available: http://docs.oasis-open.org/ws-dd/soapoverudp/1.1/

[15] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," 2014. [Online]. Available: https://datatracker.ietf.org/doc/rfc7252/

[16] D. He, J. Bu, S. Zhu, S. Chan, and C. Chen, "Distributed Access Control with Privacy Support in Wireless Sensor Networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3472–3481, Oct. 2011.

[17] Y. Faye, I. Niang, and T. Noel, "A survey of access control schemes in wireless sensor networks," *Proc. World Acad. Sci. Eng. Tech*, no. Laboratory LID, pp. 814–823, 2011.

[18] S. Yu, K. Ren, and W. Lou, "FDAC: Toward Fine-Grained Distributed Data Access Control in Wireless Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 4, pp. 352–362, 2011.

[19] J. Maerien, S. Michiels, C. Huygens, D. Hughes, and W. Joosen, "Access Control in Multi-party Wireless Sensor Networks," in *Wireless Sensor Networks SE - 3*, ser. Lecture Notes in Computer Science, P. Demeester, I. Moerman, and A. Terzis, Eds. Springer Berlin Heidelberg, 2013, vol. 7772, pp. 34–49.

[20] A. Serbanati, A. S. Segura, A. Oliverau, Y. B. Saied, N. Gruschka, D. Gessner, and F. Gomez-Marmol, "Internet of Things Architecture, Concept and Solutions for Privacy and Security in the Resolution Infrastructure. EU project IoT-A, Project report D4.2," 2012. [Online]. Available: http://www.iot-a.eu/

[21] A. El-Aziz and A. Kannan, "Access control for healthcare data using extended xacml-srbac model," in *Computer Communication and Informatics (ICCCI), 2012 International Conference on*, Jan 2012, pp. 1–4.

[22] Y. Zhu, S. Keoh, M. Sloman, and E. Lupu, "A lightweight policy system for body sensor networks," *IEEE Transactions on Network and Service Management*, vol. 6, no. 3, pp. 137–148, Sep. 2009.

[23] C. Santos-Pereira, A. Augusto, R. Cruz-Correia, and M. Correia, "A secure rbac mobile agent access control model for healthcare institutions," in *Computer-Based Medical Systems (CBMS), 2013 IEEE 26th International Symposium on*, June 2013, pp. 349–354.

[24] D. DeCouteau, M. Davis, and D. Staggs, "OASIS Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of XACML v2.0 for Healthcare Version 1.0," 2009. [Online]. Available: http://docs.oasis-open.org/xacml/xspa/v1.0/xacml-xspa-1.0.pdf

[25] J. Sermersheim, "Lightweight Directory Access Protocol (LDAP): The Protocol," pp. 1–69, 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4511.txt

[26] B. Lo and G.-z. Yang, "Body Sensor Networks: Infrastructure for Life Science Sensing Research," in *2006 IEEE/NLM Life Science Systems and Applications Workshop*. IEEE, Jul. 2006, pp. 1–2.

[27] G. Yang and M. Yacoub, *Body sensor networks*. Springer London, 2006, vol. 6, no. 3.

[28] B. Alhaqbani and C. Fidge, "Access control requirements for processing electronic health records," in *Proceedings of the 2007 international conference on Business process management*, Arthur Ter Hofstede Boualem Benatallah and H.-Y. Paik, Eds. Springer-Verlag, 2007, pp. 371–382.

[29] P. Ray and J. Wimalasiri, "The need for technical solutions for maintaining the privacy of EHR." in *Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, vol. 1, no. 1557-170X (Print) LA - eng PT - Journal Article SB - IM. IEEE, 2006, pp. 4686–4689.

[30] W. Colitti, K. Steenhaut, and N. De Caro, "Integrating wireless sensor networks with the web," in *In Proc. of Extending the Internet to Low Power and Lossy Networks , Chicago, IL, USA*, 2011.

[31] A. Westerinen, J. Schnizlein, J. Strassner, M. Scherling, B. Quinn, S. Herzog, A. Huynh, M. Carlson, J. Perry, and S. Waldbusser, "Terminology for Policy-Based Management," pp. 1–22, 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3198.txt

[32] A. A. and H. Lockhart, "SAML 2.0 Profile of XACML, Version 2.0," 2005. [Online]. Available: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-saml-profile-spec-os.pdf

[33] J. Hughes, S. Cantor, J. Hodges, F. Hirsch, P. Mishra, R. Philpott, and E. Maler, "Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0 ," 2005. [Online]. Available: http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

[34] "Sun Microsystems Laboratories, XACML." [Online]. Available: http://sunxacml.sourceforge.net

[35] "Web Services for Devices (WS4D)." [Online]. Available: http://ws4d.e-technik.uni-rostock.de

[36] "WS4D-JMEDS DPWS Stack." [Online]. Available: http://sourceforge.net/projects/ws4d-javame/

[37] K. Lawrence, C. Kaler, A. Nadalin, R. Monzilo, and P. Hallam-Baker, "Web Services Security: SOAP Message Security 1.1," pp. 1–76, 2006. [Online]. Available: http://docs.oasis-open.org/wss/v1.1/

[38] "BeagleBone System Reference Manual, RevA3_1.0." [Online]. Available: http://beagleboard.org/static/beaglebone/a3/Docs/Hardware/BONE_SRM.pdf

[39] "BeagleBoard-xM System Reference Manual, Rev. C." [Online]. Available: http://beagleboard.org/static/BBxMSRM_latest.pdf

[40] T. Dierks and E. Rescorla, "RFC 5246 - The Transport Layer Security (TLS) Protocol Version 1.2," pp. 1–104, 2008. [Online]. Available: http://tools.ietf.org/rfc/rfc5246.txt

[41] E. Rescorla and N. Modadugu, "Datagram Transport Layer Security," pp. 1–31, 2012. [Online]. Available: http://tools.ietf.org/rfc/rfc6347.txt

[42] F. Li, H. Zhang, and T. Takagi, "Efficient signcryption for heterogeneous systems," *Systems Journal, IEEE*, vol. 7, no. 3, pp. 420–429, Sept 2013.

[43] S. Nair, O. Al Ibrahim, and S. Abraham, "State machine-based security fusion for resource-constrained environments," *Systems Journal, IEEE*, vol. 7, no. 3, pp. 430–441, Sept 2013.

[44] K. Rantos, A. Papanikolaou, and C. Manifavas, "IPsec over IEEE 802.15.4 for Low Power and Lossy Networks," in *Proceedings of the 11th ACM International Symposium on Mobility Management and Wireless Access*, ser. MobiWac '13. New York, NY, USA: ACM, 2013, pp. 59–64.

[45] K. Rantos, A. Papanikolaou, C. Manifavas, and I. Papaefstathiou, "IPv6 security for low power and lossy networks," in *Wireless Days (WD), 2013 IFIP*, Nov 2013, pp. 1–8.

[46] S. Raza, S. Duquennoy, T. Chung, D. Yazar, T. Voigt, and U. Roedig, "Securing Communication in 6LoWPAN with Compressed IPsec," in *Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems (IEEE DCOSS 2011)*, Barcelona, Spain, Jun. 2011.

[47] STORK 2.0, Secure Identity Across Borders Linked 2.0, "D4.1 First version of process flows," 2012. [Online]. Available: https://www.eid-stork2.eu/

[48] A. Huertas Celdran, F. Garcia Clemente, M. Gil Perez, and G. Martinez Perez, "Secoman: A semantic-aware policy framework for developing privacy-preserving and context-aware smart applications," *Systems Journal, IEEE*, vol. PP, no. 99, pp. 1–14, 2014.

[49] R. Abdunabi, M. Al-Lail, I. Ray, and R. France, "Specification, validation, and enforcement of a generalized spatio-temporal role-based access control model," *Systems Journal, IEEE*, vol. 7, no. 3, pp. 501–515, Sept 2013.