Challenges and Opportunities in Applying Semantics to Improve Access Control in the Field of Internet of Things

Riste Stojanov, Vladimir Zdraveski and Dimitar Trajanov

Abstract— The increased number of IoT devices results in continuously generated massive amounts of raw data. Parts of this data are private and highly sensitive as they reflect owner's behavior, obligations, habits, and preferences. In this paper, we point out that flexible and comprehensive access control policies are "a must" in the IoT domain. The Semantic Web technologies can address many of the challenges that the IoT access control is facing with today. Therefore, we analyze the current state of the art in this area and identify the challenges and opportunities for improved access control in a semantically enriched IoT environment. Applying semantics to IoT access control opens a lot of opportunities, such as semantic inference and reasoning, easy data sharing, data trading, new approaches to authentication, security policies based on a natural language and enhances the interoperability using a common ontology.

***Index Terms* — IoT, Semantic Web, Access Control.**

Review Paper
DOI: 10.7251/ELS1721066S

I. Introduction

Internet of Things (IoT) is rapidly growing and it is expected that there will be around 30 billion devices deployed by 2020 [1]. Gartner [2] estimates that almost 60% of the available IoT devices were owned by regular people in 2014, and this percentage is expected to increase up to 65% by 2020. At this scale and impact, the need for protecting the data produced by the IoT devices is evident, since most of the IoT devices are tightly connected to their owners and can expose their privacy. The significance of the privacy is becoming crucial in the digital world, so one of the most important aspects here is the ability to control the data access, i.e. to define who can obtain the data and which part of the data is available. Indeed, this is the definition for the authorization that is well-studied discipline in the enterprise environments. However, the IoT devices and architectures impose new challenges and needs for completely new approaches in the authorization process.

In the near future, everything will be connected. Starting from our phones that access the Internet; continuing with our light bulbs, front doors, microwaves, comforters, blenders etc. One can drive some of these devices with an universal remote control, and pretty much all of them with a mobile phone or a web application. Some of the protocols overlap and support each other; whereas others are more exclusive. Currently there is no simple plug-and-play option to connect all of them and even less, to control the access to all of them and share or reuse the data they produce. The IoT expansion forecast means that there will be multiple devices that will generate a different kind of data, and owned by regular people, without technical skills [75][3]. Thus, the authorization process must provide a decentralized policy language in which each device owner can easily configure who can have access to which of his/hers devices, and what part of the data is available through the policy. The policy languages also have to overcome the heterogeneity of the devices and the data they generate, regarding precision, measurement unit and different serialization formats. It is not acceptable to have separate permissions for each device since it will be difficult to merge them for all different devices.

Unlike the traditional authorization approach, in the IoT domain, the data is not static. It is in the form of a stream that has temporal and spatial features. Therefore, the policies must support stream protection, in terms of who gets the data, as it is being generated. Moreover, the IoT has no sense without Machine-to-Machine (M2M) communication, where one device can trigger an action to another. In [5] the authors present a scenario in which an attacker causes a blackout to a smart lighting system by masquerading as a user device. Thus, it is crucial to protect the inter-device communication, so that the device corruption will be omitted.

A privacy disrupt by a "smart" baby monitor device that is controlled by an iOS application is presented in [6]. The problem appears due to the ability of each instance of the iOS application to pair with the baby monitor, even though the owner of the app is not a family member. Furthermore, once the pairing is done, the baby monitor signal can be obtained from anywhere, imposing a serious privacy leak. Thus, convenient policies should be able to limit the devices discoverability.

Manuscript received: May 18th, 2017
Received in revised form: December 26th, 2017

Riste Stojanov is with the Faculty of Computer Science and Engineering, University of Cyril and Methodius, Skopje, Macedonia (e-mail: riste.stojanov@finki.ukim.mk).

Vladimir Zdraveski is with the Faculty of Computer Science and Engineering, University of Cyril and Methodius, Skopje, Macedonia (e-mail: vladimir.zdraveski@finki.ukim.mk).

Dimitar Trajanov is with the Faculty of Computer Science and Engineering, University of Cyril and Methodius, Skopje, Macedonia (e-mail: dimitar.trajanov@finki.ukim.mk).

A smart health wearable IoT system is presented in [7] and it is pointed out that there is a need for so-called "Break the glass" or "emergency" policies so that in a case of a collapse of the wearables' owner, the private data will be available for the medical stuff. This scenario points out the indecipherable connection between the IoT systems and the surrounding context that they operate in. Thus, context- aware policies must be defined, to provide proper authorization for the IoT systems.

In this paper, we first introduce the related work in the area of IoT, in Section II, with a focus on the access control. Then, Section III provides an overview of the semantic technologies, applicable to overcome the heterogeneity problem in the IoT systems together with the state of the art approaches for Semantic Web authorization. We discuss the open challenges for access control in the IoT domain in Section IV and Section V explores the opportunities for access control implementation in the IoT domain.

II. Related Work

A. IoT Standardization

Many initiatives are focused on standardization and protocols for the IoT, including W3C, IEEE, and IETF. The authors in [8] categorize the standardization efforts in groups of application protocols, service discovery, infrastructure protocols and other influential protocols. Here we will shortly describe the most important application layer protocols.

The application protocols define the architecture and the way devices communicate with each other. The most popular protocols in this group are the IETF's Constrained Application Protocol (CoAP) [9] and the OASIS's Message Queue Telemetry Transport (MQTT) protocol [10]. The survey on application layer protocols for IoT [9] points out that REST Services and Web Sockets are commonly used protocols for consuming the data generated from the IoT devices. However, these protocols are rarely used on the devices themselves, since they use the TCP transport layer protocol and are not optimized for resource constrained environments. Furthermore, even though at the beginnings the Extensible Messaging and Presence Protocol (XMPP) was considered suitable for communication in the IoT domain due to its publish/subscribe architecture, it is now abandoned because of the overhead introduced by its XML messages. Nowadays, the most widespread protocols in the IoT domain are CoAP and MQTT because they are specially designed for resource constrained environments, and we will describe them in more details in this section.

The CoAP is is a request/response protocol based on REpresentational State Transfer (REST) architecture, which utilizes both synchronous and asynchronous responses. It reuses the HTTP methods, such as GET, POST, PUT and DELETE to define the interactions among the devices, which are identified using URIs. In order to be better suited for resource constrained sensor netwoks, this protocol removes the TCP overhead and reduces bandwidth requirements by utilizing the UDP transport layer protocol. When a secure communication is needed, the Datagram Transport Layer Security (DTLS) [11] can provide authentication, data integrity, confidentiality, automatic key management, and cryptographic algorithms.

The CoAP protocol exposes the devices as resources using the CoAP protocol URIs. The device state and observations can be accessed by using synchronous request/responses or by subscribing for asynchronous responses when new observation is available.

The device URIs are globally accessible thanks to the CoAP HTTP proxies, which handle the message translation between CoAP and HTTP.

As shown in Figure 1, the devices can communicate among each other using the CoAP protocol messages, which support the standard HTTP verbs (GET, POST, DELETE), while the communication to the outer world is translated by the CoAP proxy instance which translate the messages to HTTP and vice versa.

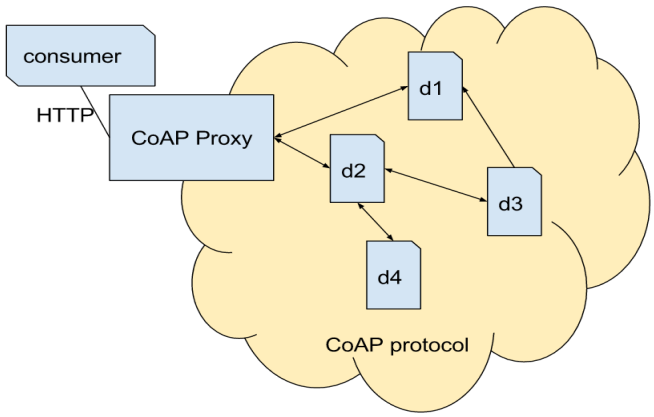

Fig. 1 CoAP Architecture

The MQTT protocol is based on the publish/subscribe mechanism, where a centralized broker distributes the messages. The broker empowers a routing in order to decide who will get the message, which makes it suitable form of M2M communication. The MQTT specification defines tree components: publisher, subscriber, and broker. The subscribers can register on the desired topics, and when the publishers send messages to those topics, the broker routes them to the subscribers. In this process, the broker is able to introduce authorization filtering, as described in [12]. Figure 2 shows the MQTT architecture where devices can publish to more than one topic on different brokers (solid lines), and subscribe for consuming the data from other topics (dashed lines). In this architecture, there is no service invocation concept, and the only way to send a command to an individual device is through a separate topic for this purpose.

The MQTT protocol is designed to use bandwidth and battery more carefully. Even though MQTT runs on TCP, it is designed to have a lower overhead compared to other TCP-based application layer protocols. MQTT does not incorporate authentication and authorization in its messages, and when a secure communication is needed, it relies on the TLS/SSL (Secure Sockets Layer), which is the same mechanism used to

ensure privacy for the HTTP protocol.

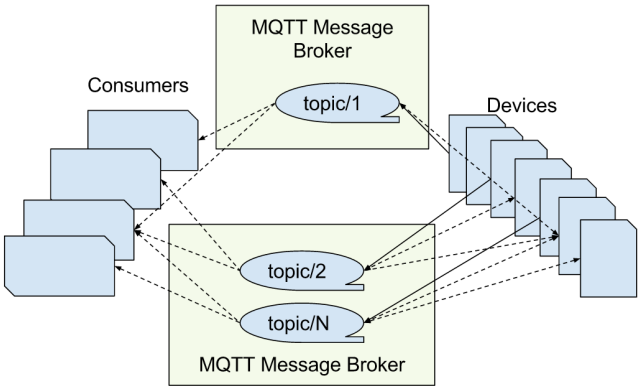

Fig. 2 MQTT Architecture

According to the study in [12], MQTT has lower delays than CoAP for low packet losses, but CoAP generates less additional traffic for reliability. However, results can vary depending on the network conditions and the QoS of the messages.

B. IoT Access Control

Most of the work in the field of IoT authorization relies on the concept of securing the communication channel with Transport Layer Security (TLS) or DTLS, through various ways of keys definition and distribution [13]-[16]. These approaches implement the authorization using shared keys for the involved devices and users in order to securely authorize their communication. They do not provide an option for filtering out the pieces of data that are prohibited in the exchange process. Thus, it can result in unwanted information sharing. This approach is partially extended by the use of OAuth protocol, where the authorization is determined based on scopes and available resources. Few implementations of this approach are available in the IoT domain, some for the MQTT protocol [15] [16], and other for the CoAP protocol [17]. The work in [18] describes how to secure the web APIs for the IoT infrastructures using the OAuth protocol for authorization.

In [19] the authors discuss that the IoT system protection should provide support for dynamic context, trust management, information flow control and actions for actuator control as well as data anonymization. In order to provide these functionalities, they implement the SecKit toolkit, which enables multiple models management, such as data, behavioral, context, and rule model, among the others. These models have a graph structure that is managed through a tree view component, which makes this process ambiguous. A model definition in their system requires a lot of technical knowledge and modeling skills. This imposes that well-trained security specialists are required to maintain the protection of the IoT systems with this toolkit. The protection is defined through an Event-Condition-Action form of rules, which provides a flexible policy definition. However, the use of the concepts from the other models makes the maintenance more difficult. This policy model is used in [12] for the MQTT application layer protocol. The authors define a policy enforcement point component, embedded in the MQTT message broker, which enforces the defined protection rules.

The IoT system authorization also depends on context in which the device operates [19]-[21]. In [21] is introduced a concept for identity-based personal location system, where the location is shared only in case of emergency. The policies proposed in this work define “a level of emergency”, which is the condition under which the policy is activated, and the location is provided. However, the authors only provide one policy example in human readable form and do not provide any further formalism. In [20] the authors describe the need for emergency policies through a use case in the health care domain, but the way of context information management and the policy format is not presented.

The data from the IoT devices can be logically represented as a stream, so there is a need for streaming data authorization. The work in [22] provides a theoretical base for streaming data protection. The authors focus on the authenticity and the completeness of the data results. In [23] and [24] the authors define a secure view, read, aggregate and join operations for stream protection with authorization filtering. The secure operations use expressions with logical and set operators, in combination with data filtering expressions, in order to define the data that should be available in the stream for the consumer. The policies are stored and processed by the Data Stream Management System. These systems require high theoretical knowledge from the system administrators in order to define the policies. There is no option for decentralized policy management, leaving the device owners without option to define how their data will be protected. This problem is solved in [25], where the data owner embeds the policies in the generated stream, and the stream processors or brokers can decide to whom to distribute the information. This work defines a policy format based on tuples and filtering, where the owner defines which roles can receive which types of the data.

The machine to machine communication protection is analyzed in [26][27] for cloud managed IoT devices with the use of an extended Information Flow Control model based on [28]. The authors point out the significance of the flow control in the IoT domain and define a formal model for their policies in the form of attaching security labels to data and processes (services or devices), and then enforce the security based on these labels. In [29] is presented a context-aware capability-based security model where the policies define a capability to each user role, and the access rights are obtained based on the available capability for the user. The context provides information that is used for capability determination.

Even though there are papers that model different aspects of the IoT authorization, such as stream protection, context awareness, information flow control and identity providing (with certificates or OAuth), there is no complete solution that provides policies that cover all these features together. None of the analyzed solutions provides overcoming the heterogeneity in the IoT domain in the process of data protection. Among these challenges, a complete policy model should also cover all the features from the traditional enterprise (API based) systems,

since the IoT devices are coordinated and consumed by this kind of applications, and the policies should provide distributed and complete protection of the whole infrastructure.

III. Semantics in IoT

One of the main challenges in the IoT domain is the heterogeneity of the devices, the way they communicate among each other and how to share their data. In the IoT field, many protocols and standards are developed, and their integration in one system requires protocol mapping, which is $O(N^2)$ problem, where N is the number of mapping protocols. The data representation format is another mapping dimension, since it depends on the implemented scenario. Additionally, the sensor observations can be expressed in different measurement units, so they need to be standardized in order to be further processed and used.

All those considerations introduce a need for unified data representation, in order to enable easier device integration. The Semantic Web technologies [30] are already well-known for providing standards for machine-readable and technology agnostic description of real world concepts, together with their relations and features. They enable knowledge modeling through the graph structure, by defining it as triples: <*Subject, Property, Object*>. The Subject is a resource that represents the concept that is being defined, the Property represents an attribute or a feature of the Subject, and the Object is the value assigned to the attribute. The properties can refer to a primitive value, such as a number, a string or a date in the case of simple features, but they can also reference another resource. The RDF standard uses Internationalized Resource Identifiers (IRI) in order to represent all concepts (resources) and their properties uniquely. There are multiple serialization formats, among which the Resource Definition Framework (RDF) [31], N3 [32], turtle [33] and jsonLD [34] are most widely used. All these standards allow knowledge representation that is self-explanatory and easy to consume.

The concepts' knowledge is usually defined in ontologies that are developed using the RDF Schema (RDFS)[35] and Ontology Web Language (OWL)[36]. The RDFS standard extends the RDF specification with the ability to assign resources into classes (rdfs:in the form ofClass), defining new properties and hierarchies of classes, whereas OWL provides a functional description of the properties and classes, such as symmetric, transitive or functional properties, disjoint classes, and many other features.

One of the downsides of Semantic Web technologies' applicability in IoT domain is that it takes more space to represent the sensory information, due to the self-explanatory form of the Semantic Web knowledge. In [37], the authors examine the impact of the different semantic formats regarding CPU cycles, power consumption, and packet size. The overall conclusion from their work is that the short form of the Entity Notation is the most optimal for semantic data representation in resource-constrained environments, while the next options are the N3 format and the jsonLD format with context references. However, even though the semantically represented data introduces some performance drawbacks, it provides an abstraction for the data being transferred and provides easier combination of the raw sensory data, which leads toward smarter and better observations.

A. IoT abstraction using semantic web

One of the most influential work in the IoT domain is the Semantic Sensor Networks (SSN) ontology [38][39]. It provides abstraction of the IoT devices, represented as *ssn:Sensor*, that observe a property (*ssn:Property*) of some feature of interest (*ssn:FeatureOfInterest*). The actual devices in this ontology are represented through *ssn:SensorDevice*, and it allows to define the platform and the deployment characteristics of the device. The measured data from the sensors are represented through the *ssn:Observation* instances. Even though this ontology is widely accepted, it does not model the different units of data representation and the domain knowledge of the device context, but it allows integration with domain ontologies for this purpose [40]-[42].

According to the survey [43], the ultimate goal of the IoT devices is to provide a perception from the raw sensory data. The raw sensory data does not have any deeper meaning for the humans, but when the abstraction is added to the sensory data, it becomes more suitable for the reasoning process that is used to produce the perceptions. The Semantic Web technologies provide a solid ground for an abstraction of real world processes and knowledge, and this is already accepted in the IoT community. The authors in [40] discuss that the IoT devices generate data streams that are time and location dependent, i.e., there is a large number of row data entities with a small size and a short lifespan. Thus, the authors propose an abstraction that extends the Observation and Measurement ontology with a connection to the Units ontology [44] for unifying the results. The meta-data they define also models the location and time of occurrence of the information and provide connection to the domain dependent ontologies. Since there are different types of devices (for example moving or static), and the stream is generated from one device, most of the data entities share the common attributes, and thus overwhelming the stream with redundant information. In [40] authors propose two stream compression techniques: (1) with grouping the entities with common attributes in a sequence, where the sequence contains the common attributes, and the elements contain the dynamic measurements; and (2) each element is using stream reference to other previous data element from which it inherits the common attributes. A discussion of the resource constrained IoT devices may not be the only place for data annotation and enrichment is presented in [45]. The authors propose that the Gateway devices, that have more resources, should be the one doing the semantic annotation, in their example with the SNN Ontology.

Unlike the previously described approaches, which represent the devices as resources that generate a stream of annotated data, the work in [41][42][46]-[48] represents the

devices as sensor services. This way of treatment of the devices is started by the definition of the SemSOS ontology [46] that enables service level integration. The authors in [48] introduce the term "sensor as a service" and extend the SSN and SemSOS ontologies for better description and abstraction of the sensory systems.

B. Integrating IoT devices and streams

The authors in [49] define that the devices are creating continuous streams, and in their follow up work [50] they use a semantic annotation to overcome the heterogeneous data and provide seamless integration. The integration of multiple annotated and integrated data streams can provide fused knowledge that is more valuable to the humans and closer to a perception [43][52].

When the devices are represented as a services, the integration process that leads toward perception extraction is implemented through service composition [42][47][41]. In [42] the authors propose semantic middleware for the IoT that provides composition of multiple services through their OWL-S definition [51]. OWL-S is an OWL extension for describing semantic web services, composed of the following tree main parts: (1) profile, (2) process and (3) grounding. The authors in [42] provide a tool for service discovery, composition, and execution in the IoT domain. A similar approach is presented in [47], where the authors provide business level integration with the help of a lightweight semantic model. In [41] the semantic model is additionally extended to define Quality of Service and Quality of Information, IoT service testing for device availability and other modules that enrich the IoT environment description. The authors in their work propose service composition based on probabilistic and logic filtering of the available devices, after which the results are ranked in order to produce results that outperform all previous work in respect to the precision at N measure.

The service composition and stream data fusion cannot occur if there is no way of device registration and discovery. The authors in [42] and [54] define middleware in which the different devices are semantically abstracted using an extension of the SSN ontology, and they register themselves to a centralized point through custom services. As [40] describes, the need for scalable solution for the IoT systems requires decentralized registration and discovery of the IoT devices. They propose to use device registries in each gateway and SPARQL[1] queries for discovering devices, with the use of the geospatial location information for narrowing down the gateways that should be queried. Simplified version of this discovery method is used in [55][41].

Autonomous perception and the actuation are the final refined products that should be provided by the sophisticated IoT systems. Furthermore, the actuation depends on perception, since when some perception occurs, some action should be taken. The process of obtaining a perception is an abductive process that produces inference to the best explanation in scenarios with incomplete information [52]. As explained in [52], efficient abstraction and semantic integration will significantly improve the perception inference through the process of reasoning. In [54] the authors propose aggregation and combination of semantically annotated data streams in each "virtual sensor"[2] in order to provide perceptions as an output.

C. Semantic security policies

In the field of Semantic Web, the problem of access control and authorization is a topic of interest of dozen research papers. The following text gives a survey of methods and techniques used for access control and authorization in Semantic Web, that we identified in our previous work [53].

According to [56] the policies for access control can be formally defined as

< Subject, Resources, AccessRight >

The *subject* represents how the policy defines the eligible users or agents, while the *resources* element defines which portion of the data is protected. Most of the current approaches define the subjects and the resources with a direct IRI referencing, or by grouping of the subjects according to their *class* or *role*. This way of policy declaring is not maintainable in large-scale scenarios, since the number of the policies will be substantially large. Thus, resource and subject grouping (using *role* properties [57]-[59], by *class* [60]-[63], or some other property [59][61]-[65]) provides more flexible way of policy definition, but it does not have the option for filtering values of the primitive properties. Data selection trough SPARQL query construct [64][66][67][58][68][69] or with rules [70] is the most flexible way for policy *subject* and *resources* definition, because they are designed for data selection with finest granularity.

Most of the related work does not consider the *context* in the policy format or only use temporal and spatial attributes for this purpose. A *context* is used in few approaches for selecting the active polices for the authorizing *subject* [71][67]. In [70] the authors define dynamic context definition with rules. As discussed in [68], a dynamic context is necessary for the protection of IoT data streams.

The access right defines whether the policy allows or denies access to the resources by the *subject* through some *action*. When only one option is available (either allow, or deny), the enforcement process is simpler, since there are no conflicts and need for their resolution. In this case, when client tries to protect the opposite scenario, the requirement must be translated with negation, which often is error prone. If both access and deny policies are available, there is no need for statement negation in the process of policy definition, but conflicts may occur, and there is a need for their resolution and detection. The access right also defines which actions will be allowed or denied by the policy. Most of the approaches available in the literature support some of the CRUD (Create, Retrieve, Update, Delete) operations for protection.

[1] SPARQL is a query language for semantic web represented resources

[2] The devices and the humans are generalized together and abstracted as virtual sensors.

The policy format is responsible for the ease of maintenance of the system security, as well as for its understandability and flexibility. The ease of maintenance is generally defined by the required time and effort for policy design and writing. Generally, the policy format and language should provide easy transformation of the user authorization requirements into policies. This means that in an ideal case there will be only one policy. Regarding understandability, the policy definition should be close to the human language, or at least managed through an interface that is intuitive. The flexibility for policy definition covers the ability to select the finest portion of the data in correlation with the context and subject, and provide them for every required action.

However, there is a trade-off between these aspects. The SPARQL and the rule-based language provide the finest granularity in terms of *resource* and *subject* selection, but they require high technical knowledge for policies definition. The flexibility, in this case, provides easier maintenance, while sacrificing the understandability.

In addition, the context definition is crucial for the policies in the IoT domain. However, for the human users, this is not very clear, since they perceive the context implicitly, and it is difficult to define it formally. For instance, it is clear for the users what is an emergency situation [75], but a formal definition for this contextual state is not a simple task.

IV. Using semantics to enhance and simplify security policies in IoT domain

The available protection systems described in Section II-B are addressing the features of the IoT devices, such as the streaming nature of their data, their inseparable connection to the context in which they operate and the need for their communication in order to provide autonomous functioning of the system. However, the work in this field does not address the heterogeneity issue introduced by the multiple device platforms, protocols and data formats. Also, the access control approaches analyzed here does not take into account that most of the users of the IoT systems will be regular people without any technical knowledge of the underlying technologies, and unaware of the security risks imposed by the smart IoT environment around them.The semantic web provides standards that overcome the heterogeneity problems in the IoT domain and enable easier integration of domain-centric abstractions, thus tracing the road toward better perceptions and more precise actuation. Even though there is a substantial work for access control in the semantic web, the aspects of managing the device discovery and information flow control are not covered. Also, even though there are work that include the streaming data in the semantic web [72][73], there are still challenges that need to be addressed regarding access control over semantic streams.

The use of semantic web can enable better IoT perception and actuation if it provides flexible, and manageable access control for the devices and their data. The people are recently becoming more aware of the value of their data and the privacy risks it imposes whether someone uses it without authorization. Thus, there is a need for unified policies that are easy to manage and understand, but flexible enough to protect the tiniest part of the data streams, together with the discoverability of their devices.

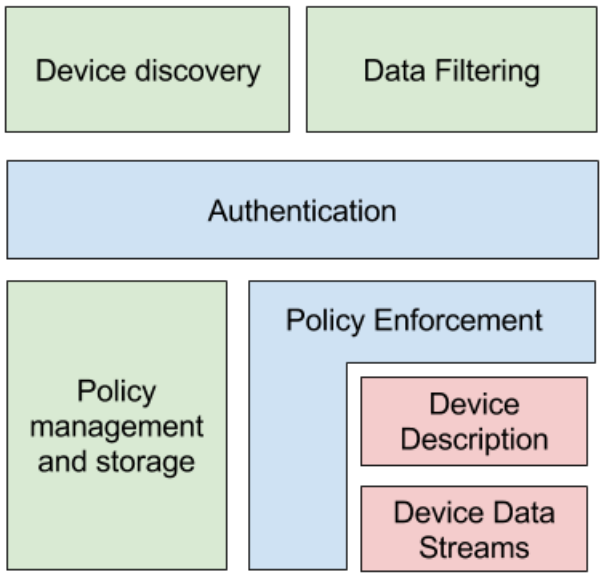

Fig. 3 Access Control modules dependency

Figure 3 provides a general overview of the components required for IoT authorization. The end-users and devices are most-often required to discover the devices and to filter the data. In order to do so, they need to authenticate their self. The policy enforcement component plays the key role in the authorization process. It uses policies, managed and stored by the policy management, and a storage module in order to define which part of the requested data should be allowed for the authenticated subject. The policy enforcement module defines the algorithms used to protect the device description in the discovery process, and to ensure that only the allowed data is returned from the devices' data streams. In this section, we will provide a further description of the components in Figure 3, with more details about the approaches that appear to be promising.

A. Device discovery

Using the semantic web abstractions and the system architecture described in [40] the data and the device discovery information are represented in the same way, enabling the same policy to protect the device discovery and data by filtering the exposure of the corresponding semantical relations. Thus, the discovery process can be achieved by adoption of the techniques for SPARQL endpoints access control. As described in Section III-C, there are multiple approaches for protection of SPARQL endpoints, but only a few of them support context awareness. The distributed nature of the IoT environment is also not considered there. The main challenges for semantically annotated device discovery protection are:

- Protection for decentralized semantics storage (multiple endpoints)
- Context support in policy definition and enforcement

B. Filtering semantic data streams

The protection of the IoT data streams defines which part of the data stream is visible to which subject. Another aspect is the protection and authorization of stream processing subscription. It can be solved by allowing subscription to everyone, but filtering everything for them when they do not have access to the data. This significantly simplifies the policies but increase the processing and complexity of the underlying system.

The main open challenges here are:

- Protecting semantically annotated data streams
- Protecting data combined from multiple streams
- Including the subject's graph in the process of policy enforcement

C. Policy storage and management

The IoT interoperability and scalability requirements impose that the access control policies should be stored in distributed manner.

A comprehensive policy model will enable easy maintenance of the policies [68]. One example is the architecture defined in [40] that allows policies to be stored and retrieved by each gateway using the SPARQL endpoints.

Since we described that there would be protected contextual data stream, the most flexible way to protect it is by defining queries that allow or deny access to some portion of the data. Since SPARQL is a formal language and most of the regular users are not familiar with it, we propose that a simple query-building interface can enable the users to use it, without sacrificing the flexibility [76]. An even better solution can be the use of guided natural language interface such as [74]. The module for Policy management from Figure 3 should provide an intuitive interface and model for policy definition. This module also provides policy storage and defines the way policies are stored and retrieved.

Additionally, incorrect policy configuration can lead to a scenario that would jeopardize the privacy and security of the IoT devices' owners. Therefore, it is important to provide design time policy validation and testing. In our previous work [67], we have designed a tool that addresses the design time policy validation through extraction of the data protected by the policy and the possible requesters that can access that data. Furthermore, we provide a design time conflict detection among the policies, together with overall unprotected data extraction. Figure 4 shows our policy management tool in action, where it extracts the protected data for a given policy and enables testing of the protected data in a given context through the generated Simulate intent form.

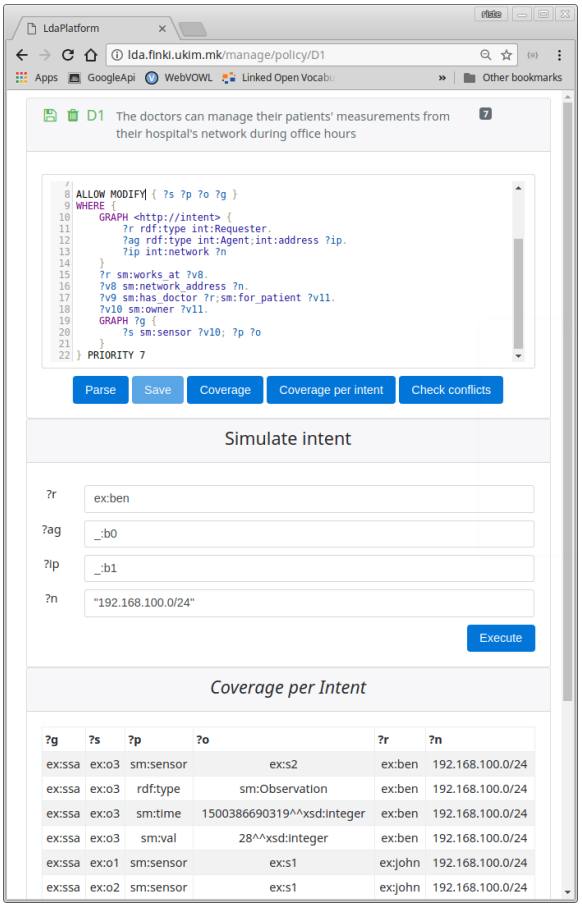

Fig. 4 Policy management console [67]

D. Authentication

In order to identify the subject that wants to consume the stream or discover the device, the WebID protocol [77][78] can be adapted for the IoT domain. This protocol transfers the subject description as a semantic graph in the headers of the HTTP message, and the protocol provides validation mechanisms using X509 public and private keys. The principle of this protocol should be reused and adapted for the IoT semantic messages. Since this protocol is based on X509 certificates for trust maintenance, the same certificates can be reused for communication protection at the transport layer, either through the TLS or DTLS protocols.

The authentication module in Figure 3 provides the information about the subject that is trying to consume the data or discover the device attributes or services. As suggested here, if an adoption of the WebID protocol is used, a semantic graph that describes the subject can be provided to the Device Discovery and Data filtering modules, which will enable them to decide about device and data availability.

E. Policy enforcement

Since the environment is provided during the device registration, data stream and the consuming subjects are represented in a semantic form, they all form the overall context

graph that should be protected. The policy engine has all the required information to decide which policies are applicable and to decide which part of this graph will be allowed for the subject. In the case of stream queries, the query engine will obtain only the portion of the stream that is allowed for the *subject*, and in the discovery case, only the allowed environment and device attributes will be exposed. The main challenge here is to provide simultaneous support on streaming data and standard queries for device and service discovery, without imposing significant performance penalties.

V. Opportunities

The semantic web technologies provide an abstraction level that opens new opportunities in the IoT technology. Their main advantage is the abstraction level they introduce, which is the main enabler for integration of multiple devices and opens the way toward better perceptions.

A. Semantic inference and reasoning

The first opportunity they open is the possibility for reasoning over the semantic data. This way, new knowledge, and perceptions can be inferred, opening opportunities for exposure to previously unknown security threats. Such example is discussed in [79], where a security threat is introduced when face recognition information is combined with the location of the person.

B. Data sharing and data trading

The unified description of sensory data with Semantic Web technologies opens an opportunity for trading with sensory information, where the device holder can "sell" the data to the consumers that can benefit from it or to expose it for the common goods. The example for the later can be publishing information such as pollution values or location for city traffic optimization (the example with Google Traffic) [80]. In these cases, it is challenging to filter or aggregate only the data that is useful for the common purpose, while hiding and protecting the personal info.

C. New approaches to authentication

In most of the current approaches, the authentication process is based on private or public keys, where their distribution is a complex process, and they do not provide any additional information about who the subject is. Thus, adoption of the webID protocol for the IoT devices can simplify the process of identification of the devices and can increase the trust among them.

D. Security policies writing using natural language

One of the biggest challenges in the access control, in general, is design and implementation of comprehensive policies. In most cases, the policy languages are hard to learn and understand, due to the use of languages that increase their flexibility and maintainability. In other cases, user interfaces are designed for convenient usage but limiting the flexibility. The natural human language is the most flexible tool through which all access control constraints can be expressed and easily understood. The semantic abstractions provide an opportunity for building guided natural language interface that will significantly simplify the process of policy design and definition.

E. Interoperability enhancing using common ontology

A standardized policy model, in the form of ontology, is also required, so that the different systems can leverage the shared domain knowledge. The schema.org is currently the most popular repository, so publication of standardized policy ontology here is a real opportunity that has potential to be widely used.

VI. Conclusion

The Semantic Web provides powerful mechanisms for knowledge representation and abstraction and this paper reviewed how the IoT systems can benefit from it. The semantic annotations can be used for device registration and discovery, whereas the semantic data streams enrich the observations and bring them closer to the desired perceptions. The interoperable nature of the semantic data, together with the reasoning techniques offer data fusion and perception inference.

The unified representation of the devices' meta-data and their observations opens new access control challenges that are not modeled by neither the IoT nor the Semantic Web research community. In this paper, we identified the potential modules that should be extended in order to solve these challenges, together with the opened opportunities for access control research. Among the most important challenges are enabling context-aware policy language that offers flexibility to protect the devices' data at a various granularity levels, and providing tools that will simplify the policy maintenance in a way that will minimize the configuration mistakes.

References

[1] A. Nordrum, "Popular internet of things forecast of 50 billion devices by 2020 is outdated," IEEE Spectrum, 18 Aug. 2016.

[2] GARTNER, "Gartner says 6.4 billion connected things will be in use in 2016, up 30 percent from 2015," Online http://www.gartner.com/newsroom/id/3165317, November 10, 2015.

[3] Stojanov R., Georgiev M., Zdraveski V., Jovanovik M., and Trajanov D.. Live objects-collaborative window in the corporate documents. In New Trends in Database and Information Systems II , pages 71–81. Cham, 2015. Springer International Publishing. AISC, volume 312

[4] Trajanov D., Stojanov R., Jovanovik M., Zdraveski V., Ristoski P., Georgiev M., and Filiposka S.. Semantic sky: a platform for cloud service integration based on semantic web technologies. In Proceedings of the 8th International Conference on Semantic Systems, ISBN: 978-1-4503-1112-0, pages 109–116. ACM, 2012. DOI: 10.1145/2362499.2362515)

[5] N. Dhanjani, "Hacking lightbulbs: Security evaluation of the philips hue personal wireless lighting system," 2013.

[6] N. Dhanjani, "Reconsidering the perimeter security argument," 2013.

[7] J. Singh and J. M. Bacon, "On middleware for emerging health services,"

Journal of Internet Services and Applications, vol. 5, no. 1, pp. 1–19, 2014.

[8] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.

[9] C. Bormann, A. P. Castellani, and Z. Shelby, "Coap: An application protocol for billions of tiny internet nodes," IEEE Internet Computing, vol. 16, no. 2, pp. 62–67, 2012.

[10] D. Locke, "Mq telemetry transport (mqtt) v3. 1 protocol specification," IBM developerWorks Technical Library, 2010.

[11] E. Rescorla and N. Modadugu, "Datagram transport layer security version 1.2," 2012.

[12] R. Neisse, G. Steri, and G. Baldini, "Enforcement of security policy rules for the internet of things," in Wireless and Mobile Computing, Networking and Communications (WiMob), 2014 IEEE 10th International Conference on, pp. 165–172, IEEE, 2014.

[13] A. Alcaide, E. Palomar, J. Montero-Castillo, and A. Ribagorda, "Anonymous authentication for privacy-preserving iot target-driven applications," Computers & Security, vol. 37, pp. 111–123, 2013.

[14] C. Hu, J. Zhang, and Q. Wen, "An identity-based personal location system with protected privacy in iot," in Broadband Network and Multimedia Technology (IC-BNMT), 2011 4th IEEE International Conference on, pp. 192–195, IEEE, 2011.

[15] A. Niruntasukrat, C. Issariyapat, P. Pongpaibool, K. Meesublak, P. Aiumsupucgul, and A. Panya, "Authorization mechanism for mqtt-based internet of things," in Communications Workshops (ICC), 2016 IEEE International Conference on, pp. 290–295, IEEE, 2016.

[16] P. Fremantle and B. Aziz, "Oauthing: privacy-enhancing federation for the internet of things," 2016.

[17] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, "Iot-oas: An oauth-based authorization service architecture for secure services in iot scenarios," IEEE sensors journal, vol. 15, no. 2, pp. 1224–1234, 2015.

[18] P. Fremantle, J. Kopecky` , and B. Aziz, "Web api management meets the internet of things," in European Semantic Web Conference, pp. 367–375, Springer, 2015.

[19] R. Neisse, G. Steri, I. N. Fovino, and G. Baldini, "Seckit: a model-based security toolkit for the internet of things," Computers & Security, vol. 54, pp. 60–76, 2015.

[20] J. Singh and J. M. Bacon, "On middleware for emerging health services," Journal of Internet Services and Applications, vol. 5, no. 1, pp. 1–19, 2014.

[21] R. M. Savola and H. Abie, "Metrics-driven security objective decomposition for an e-health application with adaptive security management," in Proceedings of the International Workshop on Adaptive Security, p. 6, ACM, 2013.

[22] S. Papadopoulos, Y. Yang, and D. Papadias, "Cads: Continuous authentication on data streams," in Proceedings of the 33rd international conference on Very large data bases, pp. 135–146, VLDB Endowment, 2007.

[23] B. Carminati, E. Ferrari, and K. L. Tan, "Enforcing access control over data streams," in Proceedings of the 12th ACM symposium on Access control models and technologies, pp. 21–30, ACM, 2007.

[24] B. Carminati, E. Ferrari, J. Cao, and K. L. Tan, "A framework to enforce access control over data streams," ACM Transactions on Information and System Security (TISSEC), vol. 13, no. 3, p. 28, 2010.

[25] R. V. Nehme, E. A. Rundensteiner, and E. Bertino, "A security punctuation framework for enforcing access control on streaming data," in Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on, pp. 406–415, IEEE, 2008.

[26] J. Bacon, D. Eyers, T. F.-M. Pasquier, J. Singh, I. Papagiannis, and P. Pietzuch, "Information flow control for secure cloud computing," IEEE Transactions on Network and Service Management, vol. 11, no. 1, pp. 76–89, 2014.

[27] J. Singh, T. F.-M. Pasquier, J. Bacon, and D. Eyers, "Integrating messaging middleware and information flow control," in Cloud Engineering (IC2E), 2015 IEEE International Conference on, pp. 54–59, IEEE, 2015.

[28] D. E. Denning, "A lattice model of secure information flow," Communications of the ACM, vol. 19, no. 5, pp. 236–243, 1976.

[29] S. Gusmeroli, S. Piccione, and D. Rotondi, "A capability-based security approach to manage access control in the internet of things," Mathematical and Computer Modelling, vol. 58, no. 5, pp. 1189–1205, 2013.

[30] T. Berners-Lee, J. Hendler, O. Lassila, et al., "The semantic web," Scientific american, vol. 284, no. 5, pp. 28–37, 2001.

[31] G. Klyne and J. J. Carroll, "Resource description framework (rdf): Concepts and abstract syntax," 2006.

[32] T. Berners-Lee and D. Connoll, "Notation3 (N3): A readable RDF syntax" W3C Team Submission, W3C, mar 2011, http://www.w3.org/TeamSubmission/2011/SUBM-n3-20110328/

[33] G. Carothers and E. Prud'hommeaux, "RDF 1.1 Turtle" W3C recommendation, W3C, feb 2014. http://www.w3.org/TR/2014/REC-turtle-20140225/

[34] M. Lanthaler, M. Sporny and G. Kellogg, "JSON-LD 1.0" W3C recommendation, W3C, jan 2014. http://www.w3.org/TR/2014/REC-json-ld-20140116/.

[35] R. Guha and D. Brickley , "RDF schema 1.1" W3C recommendation, W3C, feb 2014. http://www.w3.org/TR/2014/REC-rdf-schema-20140225/.

[36] F. Harmelen and D. McGuinness , "OWL Web Ontology Language Overview" W3C recommendation, W3C, feb 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210/.

[37] X. Su, J. Riekki, J. K. Nurminen, J. Nieminen, and M. Koskimies, "Adding semantics to internet of things," Concurrency and Computation: Practice and Experience, vol. 27, no. 8, pp. 1844–1860, 2015.

[38] L. Lefort, C. Henson, K. Taylor, P. Barnaghi, M. Compton, O. Corcho, R. Garcia-Castro, J. Graybeal, A. Herzog, K. Janowicz, et al., "Semantic sensor network xg final report," W3C Incubator Group Report, vol. 28, 2011.

[39] M. Compton, P. Barnaghi, L. Bermudez, R. Garc´ıA-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, et al., "The ssn ontology of the w3c semantic sensor network incubator group," Web semantics: science, services and agents on the World Wide Web, vol. 17, pp. 25–32, 2012.

[40] P. Barnaghi, W. Wang, L. Dong, and C. Wang, "A linked-data model for semantic sensor streams," pp. 468–475, 2013.

[41] W. Wang, S. De, G. Cassar, and K. Moessner, "Knowledge representation in the internet of things: semantic modelling and its applications," automatika, vol. 54, no. 4, pp. 388–400, 2013.

[42] Z. Song, A. A. Ca´rdenas, and R. Masuoka, "Semantic middleware for the internet of things," pp. 1–8, 2010.

[43] P. Barnaghi, W. Wang, C. Henson, and K. Taylor, "Semantics for the internet of things: early progress and back to the future," International Journal on Semantic Web and Information Systems (IJSWIS), vol. 8, no. 1, pp. 1–21, 2012.

[44] L. Lefort, "Ontology for Quantity Kinds and Units: units and quantities definitions", W3 Semantic Sensor Network Incubator Activity, 2005. http://www.w3.org/2005/Incubator/ssn/ssnx/qu/qu-rec20.html

[45] F. Ganz, P. Barnaghi, F. Carrez, and K. Moessner, "Context-aware management for sensor networks," p. 6, 2011.

[46] C. A. Henson, J. K. Pschorr, A. P. Sheth, and K. Thirunarayan, "Semsos: Semantic sensor observation service," pp. 44–53, 2009.

[47] D. Guinard, V. Trifa, S. Karnouskos, P. Spiess, and D. Savio, "Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services," IEEE transactions on Services Computing, vol. 3, no. 3, pp. 223–235, 2010.

[48] S. De, P. Barnaghi, M. Bauer, and S. Meissner, "Service modelling for the internet of things," pp. 949–955, 2011.

[49] A. Sheth, C. Thomas, and P. Mehra, "Continuous semantics to analyze real-time data," IEEE Internet Computing, vol. 14, no. 6, pp. 84–89, 2010.

[50] A. P. Sheth, "Computing for human experience: Semantics empowered cyber-physical, social and ubiquitous computing beyond the web," 2011.

[51] D. Martin, M. Burstein, J. Hobbs, O. Lassila, D. McDermott, S. McIlraith, S. Narayanan, M. Paolucci, B. Parsia, T. Payne and E. Sirin, "OWL-S: Semantic markup for web services". W3C member submission, nov 2004. http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/.

[52] C. Henson, A. Sheth, and K. Thirunarayan, "Semantic perception: Converting sensory observations to abstractions," IEEE Internet Computing, vol. 16, no. 2, pp. 26–34, 2012.

[53] Trajkova S., Stojanov R., and Trajanov D.. Semantic Web Access Control Aspects. In The 12th International Conference on Informatics and Information Technologies , pages 243-245, 2015.

[54] J.-P. Calbimonte, S. Sarni, J. Eberle, and K. Aberer, "Xgsn: An open-source semantic sensing middleware for the web of things.," pp. 51–66, 2014.

[55] S. De, T. Elsaleh, P. Barnaghi, and S. Meissner, "An internet of things platform for real-world and digital objects," Scalable Computing: Practice and Experience, vol. 13, no. 1, pp. 45–58, 2012.

[56] N. Lopes, S. Kirrane, A. Zimmermann, A. Polleres, and A. Mileo, "A Logic Programming approach for Access Control over RDF". PhD thesis, Dagstuhl, Germany, 2012.

[57] S. Kirrane, Linked data with access control. PhD thesis, 2015.

[58] S. Dietzold and S. Auer, "Access control on rdf triple stores from a semantic wiki perspective," in ESWC Workshop on Scripting for the Semantic Web, Citeseer, 2006.

[59] W. Chen and H. Stuckenschmidt, "A model-driven approach to enable access control for ontologies.," in Wirtschaftsinformatik (1), pp. 663–672, 2009.

[60] J. Hollenbach, J. Presbrey, and T. Berners-Lee, "Using rdf metadata to enable access control on the social semantic web," in Proceedings of the Workshop on Collaborative Construction, Management and Linking of Structured Knowledge (CK2009), vol. 514, 2009.

[61] S. Kirrane, A. Abdelrahman, A. Mileo and S. Decker, "Secure manipulation of linked data" in International Semantic Web Conference, pp. 248-263, 2013.

[62] H. Muhleisen, M. Kost, and J.-C. Freytag, "Swrl-based access policies for linked data," Procs of SPOT, vol. 80, 2010.

[63] S. Franzoni, P. Mazzoleni, S. Valtolina, and E. Bertino, "Towards a fine-grained access control model and mechanisms for semantic databases," in IEEE International Conference on Web Services (ICWS 2007), pp. 993–1000, IEEE, 2007.

[64] O. Sacco and J. G. Breslin, "Ppo & ppm 2.0: Extending the privacy preference framework to provide finer-grained access control for the web of data," in Proceedings of the 8th International Conference on Semantic Systems, pp. 80–87, ACM, 2012.

[65] S. Oulmakhzoune, N. Cuppens-Boulahia, F. Cuppens, and S. Morucci, "fquery: Sparql query rewriting to enforce data confidentiality," in IFIP Annual Conference on Data and Applications Security and Privacy, pp. 146–161, Springer, 2010.

[66] G. Flouris, I. Fundulaki, M. Michou, and G. Antoniou, "Controlling access to rdf graphs," in Future Internet Symposium, pp. 107–117, Springer, 2010.

[67] F. Abel, J. L. De Coi, N. Henze, A. W. Koesling, D. Krause, and D. Olmedilla, "Enabling advanced and context-dependent access control in rdf stores," in The Semantic Web, pp. 1–14, Springer, 2007.

[68] Stojanov R., Gramatikov, S., Mishkovski, I. and Trajanov, D.. Linked data authorization platform. IEEE Access, Issue 99. 2017. DOI: 10.1109/ACCESS.2017.2778029

[69] Stojanov R. and Jovanovik M.. Authorization proxy for sparql endpoints. In ICT Innovations 2017, pages 205–218, Cham, 2017. Springer International Publishing. CCIS, volume 778

[70] A. Toninelli, R. Montanari, L. Kagal, and O. Lassila, "A semantic context-aware access control framework for secure collaborations in pervasive computing environments," in International semantic web conference, pp. 473–486, Springer, 2006.

[71] L. Costabello, S. Villata, and F. Gandon, "Context-aware access control for rdf graph stores.," in ECAI, vol. 242, pp. 282–287, 2012.

[72] M. Koubarakis and K. Kyzirakos, "Modeling and querying metadata in the semantic sensor web: The model strdf and the query language stsparql," pp. 425–439, 2010.

[73] D. F. Barbieri, D. Braga, S. Ceri, E. Della Valle, and M. Grossniklaus, "C-sparql: Sparql for continuous querying," pp. 1061–1062, 2009.

[74] E. Kaufmann and A. Bernstein, "Evaluating the usability of natural language query languages and interfaces to semantic web knowledge bases," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 8, no. 4, pp. 377–393, 2010.

[75] Popovski V., Kostadinov B., Stojanov R., Mishkovski I., and Trajanov D.. Web-based disaster and crisis management system. 2013.

[76] Andreevski A., Stojanov R., Jovanovik M., and Trajanov D.. Semantic web integration with sparql autocomplete. In The 12th International Conference on Informatics and Information Technologies , pages 1–4, 2015.

[77] M. Sporny, T. Inkster, H. Story, B. Harbulot, and R. Bachmann-Gmu¨ r, "Webid 1.0: Web identification and discovery," Editor's draft, W3C, 2011.

[78] H. Story, B. Harbulot, I. Jacobi, and M. Jones, "Foaf+ ssl: Restful authentication for the social web," in Proceedings of the First Workshop on Trust and Privacy on the Social and Semantic Web (SPOT2009), 2009.

[79] J. Singh, T. Pasquier, J. Bacon, H. Ko, and D. Eyers, "Twenty security considerations for cloud-supported internet of things," IEEE Internet of Things Journal, vol. 3, no. 3, pp. 269–284, 2016.

[80] Najdenov B., Petkovski G., Jovanovik M., Stojanov R., and Trajanov D.. Automated linked data generation from the transport administration domain. In Telecommunications Forum Telfor (TELFOR), 2015, pages 827–830. IEEE, 2015.