Future Generation Computer Systems 101 (2019) 723–736

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Securing Spatial Data Infrastructures for Distributed Smart City applications and services

Kanishk Chaturvedi [a,*], Andreas Matheus [b], Son H. Nguyen [a], Thomas H. Kolbe [a]

[a] Chair of Geoinformatics, Technical University of Munich, Germany
[b] Secure Dimensions GmbH, Germany

ARTICLE INFO

Article history:
Received 30 November 2018
Received in revised form 25 March 2019
Accepted 4 July 2019
Available online 12 July 2019

Keywords:
Smart Cities
Security
SAML
OAuth2
Single-Sign-On
CityGML

ABSTRACT

Smart Cities are complex distributed systems which may involve multiple stakeholders, applications, sensors, and IoT devices. In order to be able to link and use such heterogeneous data, spatial data infrastructures for Smart Cities can play an important role in establishing interoperability between systems and platforms. Based on the open and international standards of the Open Geospatial Consortium (OGC), the Smart District Data Infrastructure (SDDI) concept integrates different sensors, IoT devices, simulation tools, and 3D city models within a common operational framework. However, such distributed systems, if not secured, may cause a major threat by disclosing sensitive information to untrusted or unauthorized entities. Also, there are various users and applications who prefer to work with all the systems in convenient ways using Single-Sign-On. This paper presents a concept for securing distributed applications and services in such data infrastructures for Smart Cities. The concept facilitates privacy, security and controlled access to all stakeholders and the respective components by establishing proper authorization and authentication mechanisms. The approach facilitates Single-Sign-On (SSO) authentication by a novel combination in the use of the state-of-the-art security concepts such as OAuth2 access tokens, OpenID Connect user claims and Security Assertion Markup Language (SAML). An implementation of this concept for the district Queen Elizabeth Olympic Park in London is shown in this paper and is also provided as an online demonstration. Such access control and security federation based realization has not been considered in spatial data infrastructures for Smart Cities before.

1. Introduction

Smart Cities is a rapidly emerging field, which allows effective integration of human, physical and digital systems operating in the built environment, and thus, improve the support of citizens and governance of cities [1]. Many large companies and organizations are involved in building Smart City infrastructures worldwide. Commercial implementations include IBM Smarter Cities [2], Microsoft CityNext [3] and The Internet of Everything for Cities from CISCO [4]. Some of the projects are also run by consortia of universities, companies and city councils in collaborative manner such as Smart Sustainable Districts [5], CitySDK [6], and City Enabler (CEDUS) [7].

One important aspect of the different Smart City concepts is that they require linking various systems for managing a city effectively. Smart City solutions are most often complex distributed systems, involving different stakeholders (e.g. owners, operators, solution providers, citizens, and visitors), agents, communities and data sources including sensors, analytical tools etc. Every stakeholder has diverse interests, goals and tasks, as well as different roles and rights. All the data sources belong to different stakeholders and are based on various platforms and Application Programming Interfaces (APIs). Therefore, it is important to achieve a data integration strategy, which must allow linking distributed data and platforms securely within a common operational framework as shown in Fig. 1.

Spatial Data Infrastructures (SDI) play an important role in linking and integrating various distributed data and systems. SDIs facilitate the discovery, access, management, distribution and reuse of digital geospatial resources [8]. In general, SDIs establish service-oriented architectures (SOA) allowing unified access to distributed resources using well-defined web services and interfaces. Such service-oriented SDIs are essential for distributed Smart City systems. They allow the data to remain with their respective owners and stakeholders and to be accessed by applications and users via well-defined interfaces. However, this requires interoperability over the connected components and

* Corresponding author.
E-mail addresses: kanishk.chaturvedi@tum.de (K. Chaturvedi), am@secure-dimensions.de (A. Matheus), son.nguyen@tum.de (S.H. Nguyen), thomas.kolbe@tum.de (T.H. Kolbe).

https://doi.org/10.1016/j.future.2019.07.002

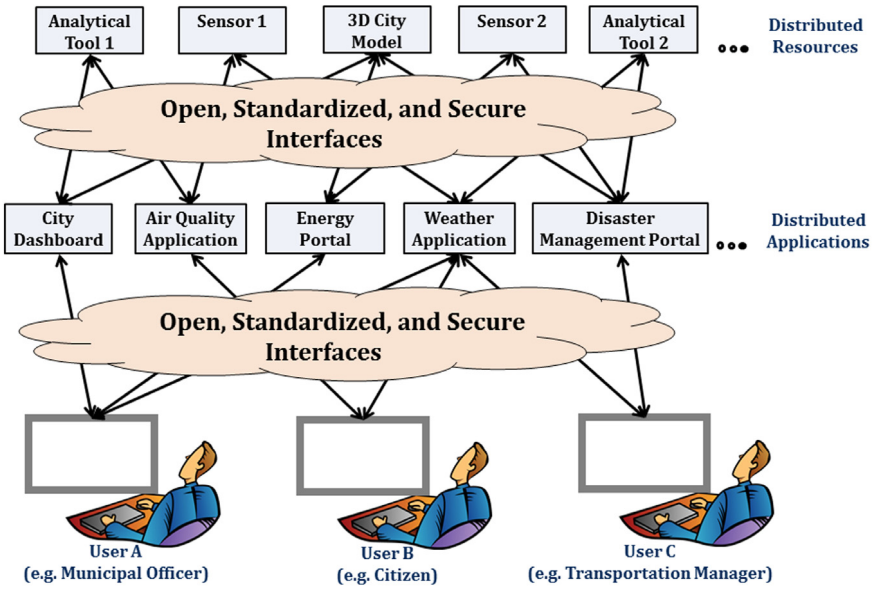

Fig. 1. Illustration of heterogeneous data sources in complex and distributed Smart City systems. In order to link and use the data within one infrastructure, the motivation is to access them using open, standardized, and secured interfaces.

systems in order to deal with the different types of data and systems. Interoperability can be achieved by using open and international standards such as provided by the Open Geospatial Consortium (OGC). These standards, on the one hand, allow modeling and representing the data sources, and on the other hand, allow interfacing the distributed systems that give access to data, applications, and analytical tools.

Towards developing SDIs for Smart City scenarios, Moshrefzadeh et al. [9] propose the Smart District Data Infrastructure (SDDI) which is focused on developing Smart City infrastructures for specific districts within selected European cities. The key aspect of the SDDI framework is that it takes into account different components of Smart City scenarios such as (i) actors and stakeholders, (ii) sensors and IoT devices, (iii) applications, (iv) simulation and analytical tools, and (v) geographic information (including physical reality of the objects like buildings, coverages and maps). SDDI provides a well-defined structure to categorize these different components and to cast them into a common operational framework. This distinguishes the SDDI from a general Spatial Data Infrastructure (SDI). Furthermore, SDDI is laid out as a service-oriented architecture and allows modeling and interfacing such distributed data and systems using well-defined information and interface models based on the Open Geospatial Consortium (OGC) standards. More details about the SDDI framework are given in Section 2.

Although such Smart City data infrastructures can increase productivity and efficiencies for citizens and governments, they may have a serious problem when they lack proper security mechanisms. Smart City solutions can facilitate access to sensitive information from different stakeholders and citizens and hence, are vulnerable to information and privacy leakage by outside attackers [10]. For example, smart meters and other types of ubiquitous sensors, pedestrian/traffic movement, simulation databases must be considered confidential data. It would cause a major threat to disclose such information to untrusted or unauthorized entities in both the physical and communication worlds. Another challenging issue is data sharing and access control. For example, within a common infrastructure with various stakeholders, it is important to establish appropriate access policies and enable privacy-preserving data sharing among the collaborators. It also requires proper identity and privacy management in order to authorize only trusted users to access the system [11]. As local governments pursue Smart City initiatives realizing the full potential of these digitally connected communities, it is key to implement security best practices by extending existing systems. Partners and stakeholders will only conduct business if their rights, trust and security requirements are met. There are also a number of other studies such as proposed by Cui et al. [12], Sookhak et al. [13], Gharaibeh et al. [14], and Biswas and Muthukkumarasamy [15], which highlight various security and privacy related issues in the context of Smart Cities.

1.1. Objectives of our contribution

This paper is a substantially extended version of the earlier work presented at the International Conference on CYBERWORLDS 2018 [16]. The paper identifies key requirements of developing and securing Spatial Data Infrastructures (SDI) for Smart City scenarios based on the proposed Smart District Data Infrastructure (SDDI) framework. Furthermore, it presents a novel concept for securing the data access and integration of distributed Smart City applications, services, simulation and analytical tools, sensors and IoT devices, and geographic information in order to meet the identified key requirements. The concept facilitates privacy, security and controlled access and provides ways to authorize and authenticate these distributed components without the need of repetitive logins. At the highest level, the approach combines the use of modern standards such as OAuth2 [17] access tokens, OpenID Connect user claims [18] and Security Assertion Markup Language (SAML) [19] based Single-Sign-On (SSO) authentication. The combination of such best practice security standards also enable easy integration with external authentication services such as public providers like Google and Facebook as well as with Academic Federations (allowing the solutions to be used by academic users). For modern, security-aware spatial data infrastructures, this is a state-of-the-art concept. To the best of our knowledge, such access control and security federation based implementations have not been considered in any spatial data infrastructures for Smart Cities before.

This paper also demonstrates an implementation of the concept for a specific scenario carried out within the district Queen

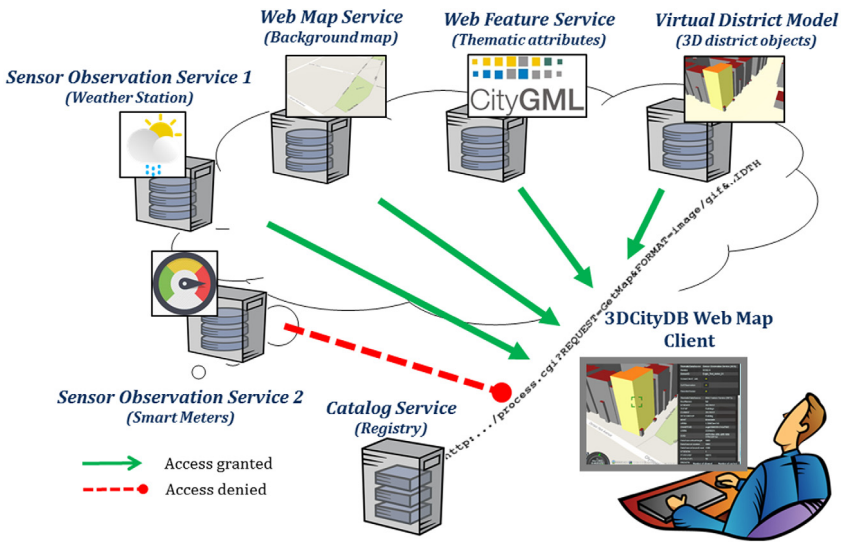

Fig. 2. Illustration of secure and controlled access to the distributed applications and services within the SDDI framework.

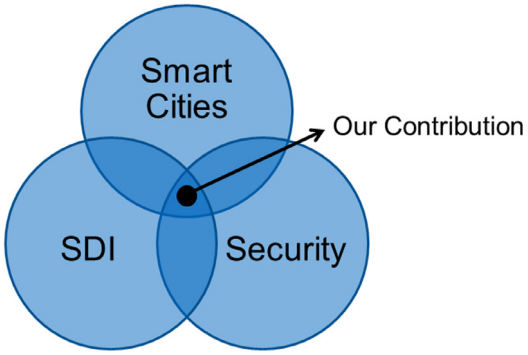

Fig. 3. Venn diagram illustrating the key focus of our contribution described in this paper.

Elizabeth Olympic Park in London. The demonstrator application is conformant to the EU General Data Protection Regulation (GDPR)[1] and allows to link different components such as 3D buildings with semantic information, weather stations, and Smart Meters in open, standardized and secure ways. In order to demonstrate handling of different identity providers, our prototype application supports individual access rights for different types of user groups including (i) public Google account, and (ii) academic users from universities and research institutes.

The rest of the paper is structured as follows: Section 2 provides a brief description of the Smart District Data Infrastructure (SDDI) framework and identifies key requirements for securing the framework. Section 3 gives a comprehensive comparative analysis on existing projects and implementations against the key requirements listed in Section 2. The scientific and technical details of the proposed methodology and implementation are given in Section 4. The implementation is demonstrated in Section 5. The last section draws conclusions about the presented work and outlines future research and development tasks.

2. Smart District Data Infrastructure (SDDI)

The Smart District Data Infrastructure (SDDI) framework proposed by Moshrefzadeh et al. [9] focuses on district level solutions within a city. The framework provides a way to integrate heterogeneous resources such as actors and stakeholders, applications, urban analytic toolkits, sensors and IoT devices with a Virtual District Model (VDM). The VDM is a 3D spatial and semantic representation of the physical reality of the district and consists of relevant objects like buildings, streets, vegetations, water bodies and networks based on the CityGML standard [20]. The VDM is also used to visualize and analyze the current situation as well as planned changes within the city district. The SDDI framework has been developed as a part of the project Smart Sustainable District under Climate-KIC of the European Institute of Innovation and Technology, and has been implemented in different European districts such as (i), Queen Elizabeth Olympic Park (London) (ii) Docks de Saint Ouen (Paris), (iii) Moabit West (Berlin), and (iv) The New Centre (Utrecht).

2.1. SDDI framework

The SDDI framework is designed based on open and international OGC standards which comprise well defined information models such as CityGML [20] for semantic 3D city models and SensorML [21] for defining sensor and IoT devices. OGC also provides a mature and well supported framework for a family of different web services such as the Web Feature Service (WFS) [22] to retrieve CityGML objects and other object-based datasets, the Sensor Observation Service (SOS) [23] and SensorThings API [24] to retrieve real-time sensor observations, and (iv) the Catalogue Service for the Web [25] allowing registering and discovering registered resources using standardized metadata. The use of international standards allow linking and managing different components in a unified and stable way and across manufacturers.

The work in this paper is demonstrated for the SDDI implementation for the district Queen Elizabeth Olympic Park in London that involves a variety of resources, of which, for illustration purposes in a simplified scenario, only a small subset are shown as follows:

- **Virtual District Model** based on the CityGML standard. It comprises of semantic 3D building and street models with spatial and thematic information stored in a 3D geodatabase.
- **Web Feature Service** allowing users to retrieve as well as modify objects from the Virtual District Model stored in the 3D geodatabases using interoperable interfaces.

[1] https://eur-lex.europa.eu/eli/reg/2016/679/oj.

- **Sensor Observation Service 1** retrieving real-time observations from a weather station installed in the park. The weather station records properties such as temperature, humidity, wind speed etc.
- **Sensor Observation Service 2** retrieving real-time observations from Smart Meters installed in important buildings such as stadium and aquatic center. The Smart Meters are managed within a proprietary platform of the company Engie and record electricity and gas consumptions for the buildings. In another recent relevant project, we have created an OGC-compliant SOS/SensorThings API interface to proprietary IoT services using our Open Source software InterSensor Service [26,27].
- **3DCityDB Web Map Client** [28] is a web-based front-end for the 3D City Database for 3D visualization and interactive exploration of large semantic 3D city models in CityGML.

Since the SDDI is a complex distributed system involving heterogeneous resources, the aim of this work is to establish a proper security layer for all the components to ensure authorization, authentication and Single-Sign-On capabilities. Such security layer enables secure and controlled access to the distributed applications and services as shown in Fig. 2. However, before establishing such security layers, it is important to identify the key requirements which should be considered for securing the SDIs for Smart Cities.

2.2. Requirements for securing SDIs for Smart Cities

This research lies in the intersection of the fields Smart Cities, Spatial Data Infrastructures (SDI) and Security (see Fig. 3).

Based on the SDDI framework (c.f. Section 2.1), the paper identifies a specific set of requirements of developing SDIs for Smart City scenarios and further securing them. The main requirements are listed as follows:

2.2.1. Smart Cities

Requirement 1: Different stakeholders. Typically, Smart City infrastructures involve distributed systems which may have different stakeholders or end users such as citizens, municipalities, utility and transportation service providers, real estate firms etc. These stakeholders are usually the group of people and organizations for which the infrastructure offers services and applications. It is important that the infrastructure considers the needs and requirements of these different stakeholders and as a consequence, not all data can and will be stored in a single system/platform.

Requirement 2: Distributed applications. It should be possible for stakeholders to register and interact with distributed applications. These applications usually implement the logics according to specific tasks and make use of different sets of data, sensor observations or simulation results involved. For example, City Dashboards, Energy Portals, Mobility Applications, and Disaster Management Portals.

Requirement 3: Simulation/Analytical Tools. There may be simulation tools or analytical toolkits, which are software components developed for specific scenarios, for example, estimating the energy demands or potentials of solar energy production for all buildings, simulating road traffic and pedestrian flows, or performing noise propagation or flooding simulations. The results of these simulations can not only be provided to the applications, but also be used for planning and forecasting. Also, results of one simulation can be used by multiple applications or one application can use results from multiple simulations. Hence, such simulation tools should be registered and operated separately from the applications.

Requirement 4: Sensors and IoT. Ubiquitous sensors and IoT devices are essential parts of several Smart City infrastructures providing detailed information by (real-time or near real-time) sensing the environment. These sensors can be stationary such as Smart Meters and weather stations. Some of the sensors can also be mobile such as moving sensors for measuring air quality. It is important to register such sensors and IoT devices in the infrastructure enabling their observations to be integrated with applications or analytical tools.

Requirement 5: Inclusion of geographic information. Nearly all Smart City concepts focus on mainstream Information and Communication Technologies (ICT) such as Internet of Things (IoT), Big Data, Cloud Computing and so on. However, it is also important to consider geographic information as a key element. Many of the simulations or planning scenarios for the cities need to work with models of the physical reality. Hence, we see semantic 3D city models [29] as an important complementary asset. These 3D city models represent both spatial and semantic information of physical objects such as buildings, roads, water bodies etc. Furthermore, semantic 3D city models provide a means for interactive and spatio-semantic queries and aggregations. It is important to consider other geographic data such as maps and coverages too, but also Building Information Models (BIM) [30].

2.2.2. Spatial Data Infrastructures (SDI)

Requirement 6: Interoperability. In order to deal with the different and heterogeneous data, applications, sensor and IoT devices, and simulation tools within a common operational framework, interoperability over all different connected components and systems is essential. Interoperability facilitates accessing and retrieving data, services, and applications by using standardized and, therefore, stable interfaces.

Requirement 7: Open International Standards. It is important that the information models and interface models are based on released and published Open Standards adopted internationally, for example, standards issued by the Open Geospatial Consortium (OGC). In the case of non-standardized Open APIs and models, there is a high risk that the encodings/APIs will be abandoned, replaced by e.g. big Internet Companies, or vanish after the project that suggested them is over.

Requirement 8: Linked Components. The use of standardized interfaces such as OGC also allows managing and accessing different components linked to each other. Dealing with such linked components is also an essential requirement for a distributed system. For example, if a Smart Meter is installed in a building, a web service (such as the Web Feature Service) can retrieve the building's semantic information, which further includes a link to the running web service (such as the Sensor Observation Service) of a Smart Meter for measuring real-time gas consumption.

2.2.3. Security

Requirement 9: Authentication and Authorization. In a complex distributed infrastructure, the most basic requirement is to protect the access to the data and functionalities. Thus, authentication and authorization of users play an important role. The term authentication means that an individual identifies himself/herself unambiguously. Typically, a username and password are used for authentication. Authorization describes the process of checking whether a user has access rights to a specific resource. However, it is not practical to use different login credentials for different resources.

Hence, modern standards such as OAuth2 [17] are used to secure applications. OAuth2 allows enabling access delegation from the resource owner (i.e. user) for a trusted application to access the protected user resources without disclosing the master credentials. It leverages access tokens for the actual access delegation aspect. OAuth2 is considered to be state-of-the-art for web and mobile applications and is supported by numerous big

players of Web 2.0 (e.g. Twitter, Google, and Facebook). However, authentication and exchange of user assertions is out of scope of the OAuth2 framework.

Requirement 10: User Information. Another important aspect is user privacy. The OpenID Connect community standard [18] has been designed as an extension to the OAuth2 framework to be able to link user assertions (user claims) with access tokens. The granularity of personal information included in the claims, linked to an access token, depends upon the user's approval, which is a very important aspect to be compliant with user privacy. Moreover, the implementation using OpenID Connect allows easy integration with external authentication services such as Google and Facebook making the application suitable for usage involving plenty of users worldwide.

Requirement 11: Single-Sign-On. In the cases of distributed systems where resources are linked, setting up a security facade for each component is cumbersome. It is not user friendly to authenticate every interface separately. Thus, it is a very important requirement to have Single-Sign-On (SSO) functionality, which should allow a user to access different applications and services without the need of repetitive logins.

An example of achieving Single-Sign-On is by the unique identification of users in a distributed system, for example, as implemented in Academic Federations like eduGAIN.[2] The eduGAIN federation is based on the international standard Security Assertion Markup Language version 2 (SAML2) [19], which is an OASIS standard to define assertion structures and protocols for exchanging assertions about users between trusted entities in a distributed system. The asserting party is the Identity Provider (IdP) and the relying party is the Service Provider (SP). Attribute assertions allow exchanging personal information about a user and authorization assertions can describe the access rights of a user on a given resource.

Requirement 12: Single-Sign-On with delegated authorization. Modern standards such as OAuth2 already support delegated authorization allowing a trusted application to access a protected resource without disclosing the master credentials. However, in order to achieve Single-Sign-On, federated authentication is required, which is not supported by the OAuth2 framework. Hence, it is essential to integrate OAuth2 with the popular standards for federated authentication such as SAML2 and OpenID. An extension to the OAuth2 framework is already available as OpenID Connect user claims allowing easy integration with authentication services like Facebook and Google. However, large Academic federations such as eduGAIN are based on SAML2. Hence, it is necessary to combine SAML2 authentication with the OAuth2 Authorization Server.

This powerful combination enables to operate an OpenID Connect compliant Authorization Server to honor the needs for modern security and web applications but also create and maintain an identity federation as operated in Academic Federations worldwide each day with hundred of millions of users.

2.3. Scenario for securing the SDDI framework

Based on the requirements listed in the previous section, this paper focuses on securing the Smart District Data Infrastructure (SDDI) framework implemented in the Queen Elizabeth Olympic Park, London. In this scenario (as shown in Fig. 4), all the services and resources are combined in an integrated application within the 3DCityDB Web Map Client [28]. The building objects of the Olympic Park are represented according to the CityGML standard. When the user clicks on a building, its thematic data such as the building name and address are retrieved from the Web Feature Service (WFS). The WFS response also includes direct links to the Sensor Observation Services giving access to Smart Meters and weather stations. For accommodating the security demonstration scenario, the infrastructure involves multiple distributed resources which are linked together in different ways.

This concept aims to fulfill the Requirements [9–12] (c.f. Section 2.2) by providing

- security layers to all of the resources, so that no resource can be accessed without proper authentication and authorization,
- federated login and Single-Sign-On access using different Identity Providers. This is demonstrated by showing that users can login using academic identity federations (such as eduGAIN service supporting approximately 2758 university identity providers worldwide) and public accounts (such as Google accounts) to all the secured resources hosted on distributed systems without repetitive logins, and
- access control to all the secured resources. Users can login via two different classes of identity providers: (i) a valid public account (e.g. Google) and (ii) a valid academic organization account linked to eduGAIN. In the scenario, if a user is not logged in, he/she can browse/view the 3D models but cannot connect to any further resource. Users logged in using the Google Identity Provider can access all resources except Sensor Observation Service 2 for Smart Meters, while users logged in using an eduGAIN based research organization's Identity Provider will be able to access all resources. In the illustrations, users from Technical University of Munich (TUM), also linked to eduGAIN, can access all the resources (see Fig. 5).

3. Literature review

There are several research studies recognizing the importance of Spatial Data Infrastructures in the context of Smart Cities. However, most of them do not define clear concepts and implementations for securing the components. Similarly, there are already frameworks proposed for Smart Cities highlighting the importance of security best practices. However, most of these frameworks do not consider distributed and heterogeneous resources. In a nutshell, the existing frameworks focus on one or two aspects from Fig. 3, but do not cover all the three aspects. As summarized in Table 1, this section gives a comprehensive literature review in the directions of Smart Cities, SDIs, and Security against the requirements listed in Section 2.2.

3.1. Smart Cities And SDIs

Recognizing the importance of open, international, and interoperable standards, several research groups have already proposed ideas and implementations to develop Smart City frameworks. The "Smart Cities Spatial Information Framework" [31] provides an architectural approach for defining information systems in Smart Cities by categorizing them according to different layers such as Application Layer, Business Layer, Data Layer, and Sensing Layer. Furthermore, the framework emphasizes on the integration of OGC open standards and geospatial technology in order to model as well as access the information systems. The OGC Smart City Interoperability Initiatives [32] include testbeds and pilots for Smart City infrastructures. One of the first initiatives recently completed is the OGC Future City Pilot Phase 1 [33]. The pilot aimed at demonstrating and enhancing the ability of SDIs to support quality of life, civic initiatives, and urban resilience. One of the objectives of the pilot was to demonstrate "how dynamic city models can provide better services to the citizens as well as can help to perform the better analysis?".

[2] https://edugain.org/.

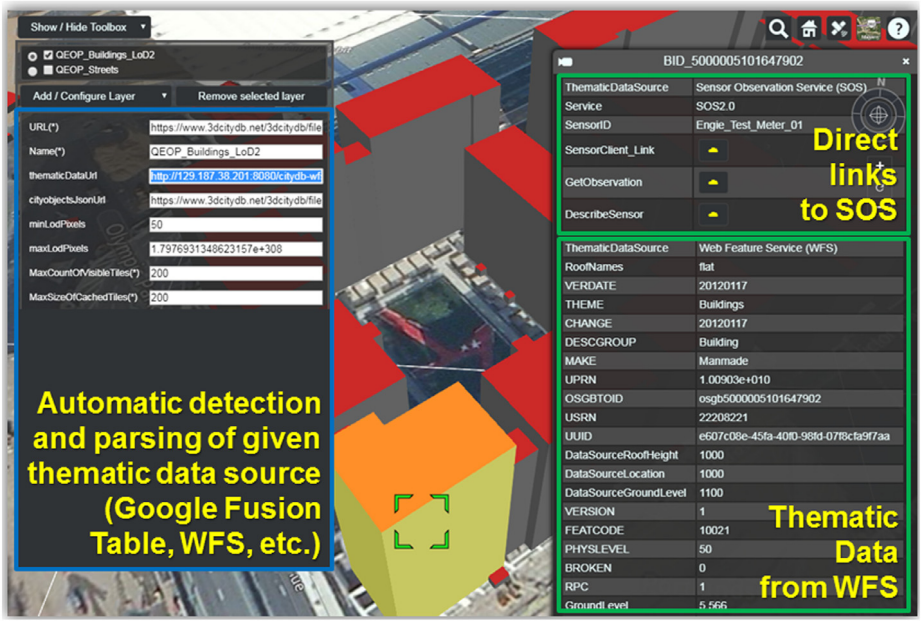

Fig. 4. Representation of chaining of distributed resources in the SDDI framework. Screenshot taken from 3DCityDB Web Map Client [28] developed by Technical University of Munich.

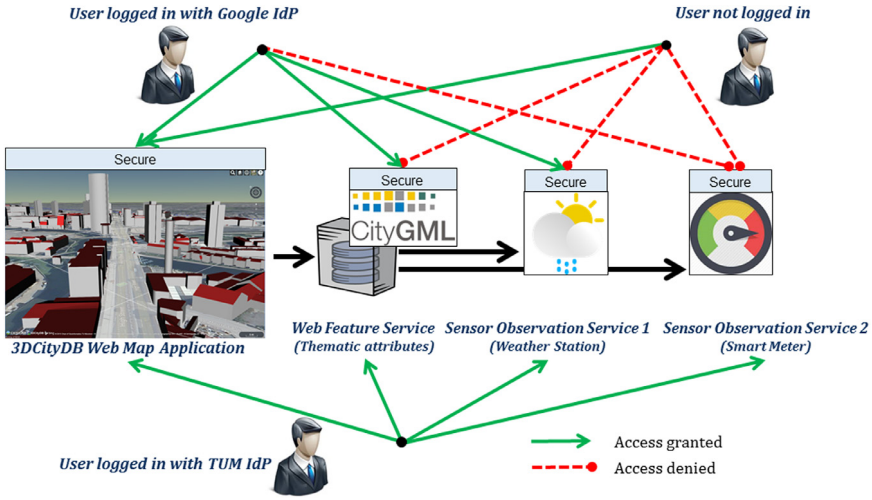

Fig. 5. Illustration of the security demonstration scenario showing that users identified by different identity providers can access the distributed components. Green arrows mean 'Access Granted' and red dashed arrows mean "Access Denied" to specific components. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Within this use case, the city's static data such as buildings with elderly citizens having special needs could be integrated with dynamic data such as outside temperature or air humidity using interoperable OGC standards such as CityGML [20] and the Sensor Observation Service [23]. Such potential integration within council owned assets could lead to better decision making in case of extreme weather or other emergency scenarios matching human needs to the right housing or resources.

Similarly, the ESPRESSO project [34] aims to provide cities and communities the ways for implementing enhanced interoperable and standards-based architecture for their specific city contexts. This project defines key elements and concepts required to be addressed to achieve interoperability between various services within a city and also to increase the interoperability between different cities. One of the key objectives of ESPRESSO is to identify a collection of open standards that work well together ("conceptual standards framework"), having been proven to help Smart Cities, and of course to identify gaps and weaknesses in the framework of available standards. The concepts developed under this project have already been tested and proven in Rotterdam (the Netherlands) and Tartu (Estonia). The importance of open and interoperable solutions for Smart Cities is also being recognized in the form of developing user guides for cities and stakeholders and by organizing hackathons and webinars for encouraging innovative application ideas. The Smart City Interoperability Reference Architecture (SCIRA) [35] is an initiative by the OGC Innovation Program. The purpose of this project is to advance standards for Smart Safe Cities and develop open and interoperable designs for incorporating IoT sensors into city services. As part of SCIRA, a hackathon "Hacks and the City"[3] was recently organized which encouraged participants to design and implement new application ideas that use a variety of city datasets and data sources

[3] https://scira.ogc.org/hack.

Table 1
Summary of reviewed literature against the requirements listed in Section 1.1.

Research work	Requirements listing											
	Stakeholders 1	Applications 2	Simulations 3	Sensors & IoT 4	3D City Models 5	Interoperability 6	Open standards 7	Linked access 8	Security 9	Privacy 10	Single-Sign-On 11	SAML+OAuth 12
OGC Smart Cities Spatial Information Framework [31]	✔	✔		✔		✔	✔	✔				
OGC Future City Pilot Phase 1 [33]	✔	✔	✔	✔	✔	✔	✔	✔				
ESPRESSO [34]	✔	✔		✔		✔	✔	✔				
OGC SCIRA [35]	✔	✔		✔		✔	✔	✔				
SMACiSYS [36]	✔	✔		✔		✔	✔	✔				
MONICA in Hamburg [37]	✔	✔		✔		✔	✔	✔				
Smart Emission Project [38]	✔	✔		✔		✔	✔	✔				
SEnviro [39]	✔	✔	✔	✔		✔	✔	✔				
Smart City Security Layer [40]	✔	✔		✔					✔			
Integrated Component for Cloud Services [41]	✔	✔		✔					✔			
Integrated Access Control Service Enabler [42]	✔	✔		✔					✔	✔		
BIG IoT [43]	✔	✔		✔		✔	✔		✔	✔		
FIWARE [44]	✔	✔		✔		✔	✔		✔	✔		
Integrated Cloud Service Control [45]	✔	✔		✔		✔			✔	✔		
Smart City Security Framework [15]	✔	✔		✔		✔			✔	✔		
OGC Authentication Interoperability Experiment [46]	✔	✔					✔		✔			
GEO AIP 6 [47]	✔	✔					✔	✔		✔	✔	
COBWEB [48]	✔	✔					✔	✔	✔	✔	✔	
Triple-A Approach [49]	✔	✔					✔	✔	✔	✔		
Our contribution	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

to improve public safety, responder awareness, and community resilience.

There are also several other projects and frameworks using OGC-based standards in Smart City contexts such as Smart Cities Intelligence System (SMACiSYS) [36], MONICA in Hamburg project [37], Smart Emission project [38], and SEnviro [39]. Although they emphasize on the importance of securing Smart City implementations, however, none of the initiatives define clear concepts and implementations for securing the components.

3.2. Smart Cities and Security

Modern standards such as OpenID [18], OAuth [17] and Security Assertion Markup Language (SAML) [19] are being considered as best practices as they allow integrating distributed services with the proper authorization, authentication, and Single-Sign-On capabilities. Large service providers such as Facebook, Google, and Microsoft are developing their commercial applications utilizing these state-of-art security best practices [50]. Likewise, the importance of such modern standards have also been mentioned in various Smart City frameworks. For example, Ferraz et al. [40] emphasize on a set of information security issues in the environment of a Smart City and propose a new approach called "Smart City Security Layer" which aims to increase security by providing entities (such as citizens, services, and sensors) with the mechanics to interact with systems using unique IDs for each system. The Security Layer is based on cryptography that generates different and unique IDs for each system relating to each citizen facilitating Single-Sign-On. However, the approach does not mention the ways for handling geographic information and only deals with sensor information. Lämmel et al. [41] propose an Integrated Component for Cloud Services (ISCS), which combines the OAuth and OpenID standards to permit secure access on cloud based services for different applications and users. The concept allows to enhance a cloud based data platform for Smart Cities with authentication and authorization features. However, there is no mention of providing Single-Sign-On capabilities to users. Likewise, Thanh et al. [42] propose an integrated access control service enabler which is a RESTful security service leveraging the above-mentioned security standards in order to provide authentication, authorization, and audit logging services for cloud applications. The framework supports OAuth based federated authentication to utilize federated identities by other truster identity providers. However, there is no mention on dealing with distributed web services and applications.

The project BIG IoT [51] deals with distributed data within Smart City applications, however focusing only on IoT ecosystems. The approach allows to register an individual IoT platform to their so-called "BIG-IoT Marketplace", which acts as a catalog. Using the Marketplace, the BIG-IoT API allows discovering, authenticating/authorizing multiple IoT resources and using them in a single application. Hernández-Serrano et al. [43] propose ways to secure IoT Ecosystems followed in the BIG IoT project for the use cases of private, public transportation and smart parking. However, the project only deals with IoT platforms. Handling other data such as geospatial data is out of scope for the project. Similar to BIG IoT, FIWARE [52] is also an open-source platform that aims to make interoperable city services, to provide access to real-time context information, and to implement Smart City applications. The platform enables developers and communities to create their services based on commonly defined APIs and data models. FIWARE is already being used in several Smart City initiatives such as "City Enabler" [7]. FIWARE has a security architecture [44] dealing with key security features such as identity management and access control. It also provides a basic support to handle geographic information (for example, the point location of a sensor). However, there is no mention on dealing with physical reality of objects such as spatial and semantic information of a building. Focusing only on cloud computing and IoT, Sciu et al. [45] propose an Integrated Cloud Service Control for using cloud computing capacities for provision and support of ubiquitous connectivity and real-time applications and services for Smart Cities' needs. The framework allows data to be procured from highly distributed, heterogeneous, decentralized, real and virtual devices (sensors, actuators, smart devices) and can be automatically managed, analyzed and controlled by distributed cloud-based services. Furthermore, Biswas and Muthukkumarasamy [15] propose a Smart City Security Framework that integrates the Blockchain technology with

smart devices to provide a secure communication platform in a Smart City. However, the framework deals with homogeneous IoT devices. Handling interoperability of different and heterogeneous IoT platforms is mentioned as a future work by the authors.

3.3. SDI and Security

Several researchers have contributed to securing SDIs for accessing and retrieving distributed data. However, most of the infrastructures are either commercial/proprietary solutions or do not consider Smart City application scenarios (e.g. lacking support of sensor and IoT platforms). One of the initial works in this direction is the OGC Authentication Interoperability Experiment [46] which focused on developing ways of transferring basic authentication information between OGC clients and OGC web services. However, the solution is developed leveraging mechanisms in existing basic protocols (e.g. HTTP Authentication). The experiment did not consider more sophisticated standards such as OAuth2. Furthermore, Matheus [47] highlights the importance of Single-Sign-On for improving the usability of open and protected geospatial services. The solutions are developed under the Group on Earth Observations (GEO) Architecture Implementation Pilot (AIP) no. 6 for protecting the OGC web services. The protected web services are made available to different kinds of clients using the SAML standard. However, this study is not oriented towards Smart City applications and also lacks supporting OAuth based federated authentication.

Another project named Citizen OBservatory WEB (COBWEB) [48] developed a generic infrastructure platform to facilitate the collection of citizen science data for the purpose of environmental monitoring. The project also demonstrates the Single-Sign-On functionality to protected web resources across administrative domains. It allows users to login using their organizational credentials (presented via a SAML Identity Provider) and then access protected resources (typically OGC web services) presented as SAML Service Providers. In this way, users only need to login once and can then access multiple protected resources in the federation, providing they are authorized properly. However, a full integration of distributed web services is not achieved in the project. Resch et al. [49] also propose a lightweight "Triple-A Approach" for securing distributed geo-service infrastructures. The framework comprises methods for authentication (confirming the user's identity), geo-authorization (defining and enforcing spatial access rights) and optimized storage and administration of access rules. The implementation uses OAuth for authorization, OGC Filter Encoding for storing authorization rules and OpenID for authentication and allowed different types of users to access distributed geo services in a secure way. However, the solution does not provide Single-Sign-On functionalities.

To summarize, there is no research work that fulfills all the requirements listed in Section 2.2. Although modern standards such as SAML, OAuth, and OpenID are being used in different studies for authorization, authentication, and Single-Sign-On capabilities, it is equally important to ensure that such security mechanisms can be established for distributed and heterogeneous resources in a unified and standardized way. This research attempts to secure the Smart District Data Infrastructure (SDDI) framework, which already provides sophisticated ways to categorize different components such as stakeholders, applications, analytical tools, IoT devices, and a semantic 3D city model and integrate them within one common operational framework utilizing open and international OGC based standards.

4. Securing the SDDI - Methodology and Implementation

Fig. 6 gives an overview of the security demonstrator architecture. For simplicity, the figure illustrates only the relevant components including the 3DCityDB Web Map Client, Web Feature Service, Sensor Observation Service 1 for a weather station, and Sensor Observation Service 2 for a proprietary Smart Meter platform. However, other applications and web services can also be secured by using the steps mentioned in the following subsections. As stated in Section 2.1, the aspect of a single online identifier for the user across any service of a distributed system is key in the same way as Single-Sign-On (SSO) is for ensuring usability. It is also important to verify which session creation and management is required based on the overall service architecture and the technical implementation of the client application.

4.1. Implementing Single-Sign-On

SAML2 specifies a Web Browser SSO Profile which involves an Identity Provider (IdP) and a Service Provider (SP). As illustrated in [19], the session initiation is triggered by the Service Provider (SP) based on two HTTP redirects. This way of initiating a session is limited to native Web Browser interactions, but that is already implied from the profile's name as well. It is important to note that redirects require that a session cookie is transported with the interactions: the first request from the client to the SP creates a temporary session which gets referred to in a cookie. It is also important that this cookie is sent on the second redirect to ensure that the SP can create a real session and issue another cookie, referencing the full session. From then on, the session is referred to by all requests initiated by the Web Browser that contain the session cookie.

Trying to adopt this SAML2 protocol behavior for a JavaScript based Web-application may result in a conflict with the Same Origin Policy. This policy safeguards the content loading in a Web Browser by intercepting network requests initiated by JavaScript and XMLHTTPRequest object or Asynchronous JavaScript and XML (AJAX). Regardless of the technology, the Web Browser verifies the conditions under which the network request is initiated based on the W3C Cross-Origin Resource Sharing (CORS) [53] recommendation. Without going into details, any JavaScript application gets loaded from a Web Server of which its hostname is considered the origin of the code. If a network request gets initiated to another hostname which is not in the same domain or sub-domain, the receiving Web Server must reply with particular HTTP headers. According to the W3C CORS recommendation, the Origin changes to the literal 'null' after any HTTP redirect. This means that for the SAML2 session initiation via the Web Browser SSO profile, the redirect ending at the IdP will carry origin 'null'. This disables the intended use of that HTTP header, which is to determine the trust of the JavaScript code based on the hostname from which the code was loaded. Based on 'null', the typical whitelisting can no longer be applied. Therefore, the IdP must blindly trust the redirected request, which it should not. At this point, the interaction to initiate a new session with the SAML2 SP fails assuming a proper validation of the origin.

Studying the OAuth2 framework general protocol [17], it is found that the session initiation is different compared to SAML2. In particular, no two-way HTTP round-trip is required to instantiate a new session. A session is referred to via an access token. The application does interact with the Authorization Server to obtain an access token leveraging one of the different protocols (grant types) that are designed to work well with the Web Browser Same Origin Policy and Web-Applications. Once the application has received an access token, it can be used for any calls to the protected resources hosted at the Resource Servers.

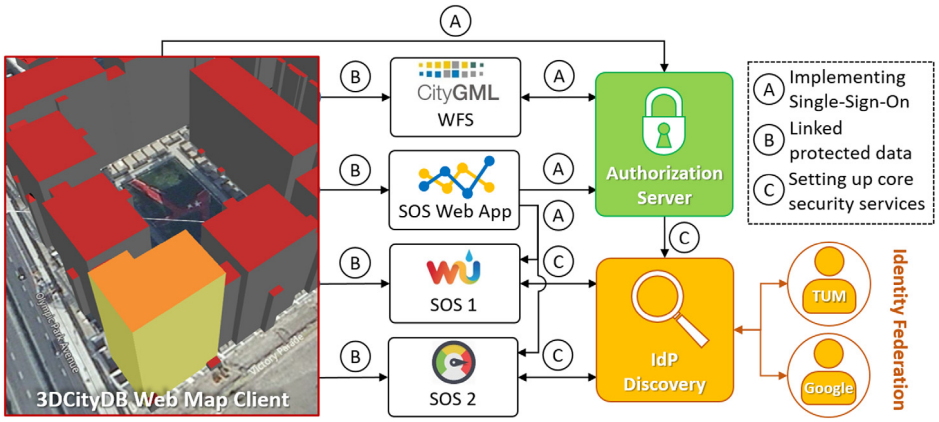

Fig. 6. An overview of the security demonstrator architecture. The notations "A", "B" and "C" in the figure refer to Sections 4.1, 4.2 and 4.3 respectively.

4.2. Linked protected data

Before introducing the details of the implementation and the securing of the service interfaces, we review the user interactions regarding the identification of the relevant and suitable standard to implement Single-Sign-On. For this approach it is essential to study the general usage of the user application, the interactions of the user, and the concrete implementation of the actions triggered by the user.

As shown in Fig. 5, majority of interactions take place between the user and the 3DCityDB Web Map Client. The client loads the CityGML based 3D city model and renders the information. Also, the application extracts URLs to additional resources (such as the Web Feature Service and Sensor Observation Services) linked from the CityGML response. For example, the 3D building model data contains links to those sensor services giving access to the sensors operated within the building. However, these links to other resources are protected. Such a link cannot contain any security context as that would be insecure. Therefore, it must either be the application or the Web Browser that can add information to the link when it is followed. At this point, it is important to know whether the Web Browser or the Web Application is going to follow the link. The first link is from the rendered 3D model to additional information about each building. This information is available to the user by clicking on each building. The 3DCityDB Web Map Client initiates the network call which causes the Web Browser to inspect the call towards CORS. This is the first fact to note when implementing the security to the Web Feature Service (WFS).

4.2.1. Securing the WFS interface

The securing of the WFS must leverage OAuth2 access tokens as the 3DCityDB Web Map Client follows the link to fetch the information from the WFS. Therefore, the WFS must be protected as an OAuth2 Resource Server (RS) accepting OAuth2 access tokens. The interface behavior for a RS is defined in the OAuth2 Bearer Token Usage [54]. According to that specification, the RS must accept the access token either as part of the URL (parameter access_token) or as part of the HTTP header named Authorization using the scheme 'Bearer'. After the access token is isolated from the incoming request, the RS must validate the access token. Because access tokens are of type bearer, the RS must request validation by the Authorization Server (AS) that issued the token. For supporting this interaction in an interoperable fashion, the AS for this prototype implemented the OAuth2 Token Introspection [55].

Assuming the RS has successfully verified the access token, it could undertake access control based on the token metadata received from the introspection endpoint or based on the user information that the RS can request from the OAuth2 assuming it is OpenID compliant. For the implementation of this protoype, no further access control is implemented. This means that any authenticated user can obtain detailed building information from the WFS.

4.2.2. Securing the SOS interface

The next level of linking is based on the links included in the WFS response: the FeatureCollection. Each geographic feature contains detailed information about a building including different types of links, but all pointing to protected endpoints. For this level of linked data, the URIs can resolve to different resource types: (i) the first kind of link would return a Web Application which is used to visualize sensor readings that are the second resource type: (ii) the second kind of link would return the responses of sensor description and sensor observations in pure XML format.

The link that refers to the Sensor Visualization Application must be resolved directly by the Web Browser. Therefore, the SAML2 session initiation must be implemented on this endpoint. The link that refers to sensor observations connects to an OGC SOS initiating the GetObservation operation. As it is a requirement for our demonstrator to show the native SensorML result in a Web Browser, the SAML2 session instantiation is sufficient. The SAML2 session management is also sufficient for this sensor visualization application, as it is loaded from the same hostname and path as the actual sensor readings. In the general case, where the sensor visualization application and the sensor readings are not hosted on the same machine, the session and access management can be based on OAuth2 access tokens.

The access controls were implemented based on the login origin of the user. As described in Section 3, when a user is logged in using a Google account, he/she can access only the sensor reading from Sensor Observation Service 1. When the user is logged in using an arbitrary eduGAIN account (like the user account from TU Munich), he/she can access sensor readings from both the Sensor Observation Services.

4.2.3. Modifying the Web 3D Application

Based on the chosen security for the WFS and SOS interfaces, the 3DCityDB Web Map Client now only needs to be enabled to use OAuth2. This can be achieved by integrating any open source library that supports OAuth2. In case that the application likes to salute the user, the library must also have implemented OpenID Connect support. The library chosen in this work is HelloJS [56]. In order to enable the 3DCityDB Web Map Client to obtain access tokens from the Authorization Server, the application must be registered. Because the application is considered 'non-confidential', it must leverage the OAuth2 Implicit Grant.

4.2.4. Modifying the sensor visualization application

This application was not modified, as SAML2 session instantiation is implemented with the SOS interfaces. As discussed, the session initialization is done by the Web Browser itself following the SAML2 SSO Web Browser profile when loading the application. Once the session is established, the application is loaded and can then fetch sensor readings from the same SOS leveraging the existing session referenced by the HTTP cookie.

4.3. Setting up the core security services

In addition to adapting the web application and the services to support the required security interfaces, there needs to be 'core' services as illustrated in Fig. 6. First of all, there needs to be Identity Providers to allow the user to login with e.g. Google and TUM. For the latter one, the TUM Identity Provider registered with eduGAIN is used. The IdP for supporting Google login is a SAML2 gateway that is based on a standard SimpleSAMLPHP [57] deployment which is open source. In case there is more than one IdP, an IdP Discovery Service must implement the SAML2 IdP Discovery Profile. To support Single-Sign-On, we have decided to use the central Discovery Service [58], implemented as open source by the Swiss academic federation, SWITCH. We have also created a coordination center that maintains and signs the SAML2 metadata that represents this mini federation. In order to make the Google IdP bridge work, we have registered the IdP bridge as an application with Google.

The protection of the SOS is based on the Shibboleth Service Provider implementation that is open source and also commonly used in the Academic Federations around the world.

Even though all deployments for creating the federation are based on open source, the Authorization Server is implemented as an extension to the Open Source implementation from Brent Shaffer. The basic OAuth2/OpenID Connect library, available from Github [59], was extended to support the SAML2 federation login. This specific extension unfortunately is not open source at the moment, but could be implemented when following the explanations of this paper and the referenced standards.

Setting up the Resource Server to protect the access to the WFS, we used a typical web server stack: The Internet facing web server is an Apache that is also configured to support HTTPS. It is important to note that all communication is via HTTPS. The actual services were deployed on Tomcat, the defacto default hosting for Java based services beside Jetty and JBoss. In order to protect the service endpoints, we created a simple PERL handler that implemented the OAuth2 Bearer Token Usage [54]. The handler, loaded as an Apache module, intercepts service requests and interacts with the Authorization Server to validate the received access token.

Finally, the support for the W3C CORS recommendation was implemented as another module inside the Apache 2.4 deployment. For returning HTTP headers to support CORS, we do not use a whitelisting for the JavaScript code as the service endpoints are protected as OAuth2 Resource Servers and can only be accessed with a valid access token. However, one must keep in mind that a HTTP request with submitting the access token as an HTTP Bearer header causes the Web Browser to execute a so called pre-flight request to check the response headers before actually submitting the intercepted request. The pre-flight request is an HTTP OPTIONS request and even though a GET and POST request require an access token, the Options request does not. This specific CORS behavior was configured into the Apache web server.

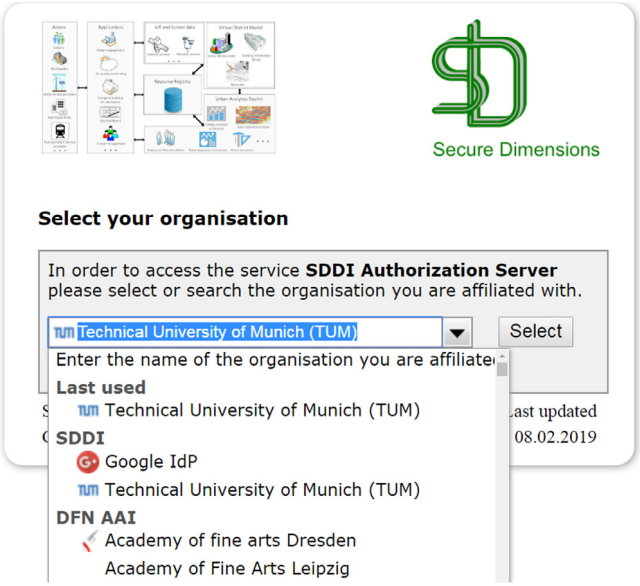

Fig. 7. Selection of appropriate Identity Provider to access the resources.

5. Using the Secured SDDI - Demonstrations

Based on the methodology and implementations as described in the previous section, the security and access control layers were successfully set up on all implemented resources including the 3DCityDB Web Map application, the Web Feature Service (WFS), Sensor Observation Service (SOS1) for weather station, and Sensor Observation Service (SOS2) for a proprietary Smart Meter platform. The additional security facades allow ensuring that (i) no resource can be accessed without proper authentication, (ii) federated login and Single-Sign-On access to all the secured resources hosted on distributed systems with one login, and (iii) access control with proper rights, roles, and grants to all the secured resources.

The Security demonstrator described in this paper is publicly available[4] and readers are encouraged to try it. The application utilizes the powerful combination of SAML2 and OAuth2, which not only enables to operate an OpenID Connect compliant Authorization Server to login using valid public accounts (in this case Google), but also creates and maintains an identity federation as practiced in Academic Federations worldwide (e.g. organizational accounts through eduGAIN service) as shown in Fig. 7.

The login access can then be provided on the basis of required authorization. As shown in Fig. 8, when a user logs in using a specific credential, based on the valid authentication and authorization rules, access tokens are generated for the protected services. The Resource Server requests validation by the Authorization Server that issued the token. Upon successful validation of the access token, appropriate access control is given based on user information that the Resource Server requests from the OpenID Connect User Information endpoint which is a part of the OAuth2 Authorization Server operated for the demonstration. In the case of the WFS, there is no further access control, which means any authenticated user can obtain building information from the WFS. In the cases of SOS1 and SOS2, access controls were implemented based on the login origin of the user.

According to the scenarios mentioned in Section 2.3, the demonstrator application showcases three scenarios:

[4] www.gis.bgu.tum.de/en/projects/smart-district-data-infrastructure/#c2414.

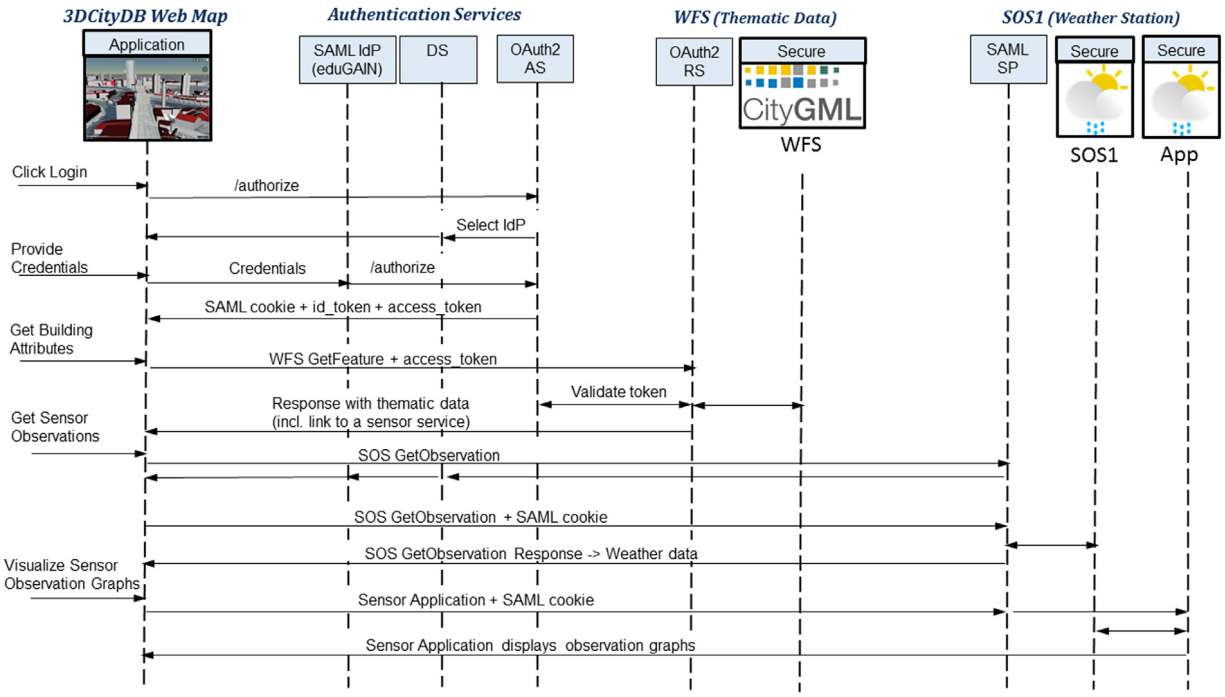

Fig. 8. Illustration of steps performed for authorization, authentication, and Single-Sign-On with the help of a sequence diagram. For simplicity, the diagram only shows the cases for authenticating and retrieving a building's attributes from WFS and sensor observations from SOS1 (weather station). In the diagram, the following abbreviations are used: (i) IdP for Identity Provider, (ii) SP for Service Provider (iii) DS for Discovery Service, (iv) AS for Authorization Server, and (v) RS for Resource Server.

1. When a user is not logged in, he/she can view the 3D city model, but cannot connect to the protected WFS and further SOS1 and SOS2 in order to retrieve thematic and sensor data.
2. When a user is logged in using a valid Google account, he/she can access the WFS and SOS1 (which is a service running on a public weather station) as shown in Fig. 9. However, SOS2 (which is a service running on a proprietary Smart Meter) is not accessible to this user group.
3. When a user is logged in using an organizational account (in this case any account supported through eduGAIN service), he/she can access the WFS as well as both SOS1 and SOS2.

Upon a successful validation of credentials and access control roles, the user is able to connect to the respective resource and retrieve the information. In this way, the application fulfills the Requirements [9–12] (c.f. Section 1.1) and ensures (i) that no resource can be accessed without proper authentication, (ii) federated login and Single-Sign-On access to all the secured resources hosted on distributed systems with one login, and (iii) access control with proper rights, roles, and grants to all the secured resources can be performed.

In addition, the application is also compliant to the new EU General Data Protection Rights (GDPR) in order to regulate the processing of personal data. The amount of personal data that can be collected by the application can be configured while registering the application at the Authorization Server. It is possible to display the amount of personal data collected by the application by (i) registering with a particular level, and (ii) by choosing a login for a particular level. Personal data is only processed after the user's approval.

5.1. Performance evaluation

This sub-section shows the performance of the application by a comparison between the total response times when querying the different services with and without security layers in place (as shown in Table 2). Without security in this context means that no authentication and authorization mechanisms are deployed for the running web services. For both cases, the mentioned values are the average of 100 measurements for each service. However, the performance evaluation for the application used by multiple concurrent users is out of scope of this paper and load tests for this purpose will be performed in the future.

As shown in the table, multiple requests were made to the running services SOS1 (Outside Temperature being measured by a weather station), SOS2 (Electricity Consumption readings by a Smart Meter), and WFS (Thematic attributes of the building in consideration). The observation frequencies by SOS1 and SOS2 are 10 and 30 min respectively. The types of requests with security layers that were performed for each web service and the payload sizes of the responses are mentioned in columns 4 and 5 respectively. The performances were evaluated for the request SOS GetObservation (queried for the duration of 15 days) by using SAML Session Cookie and Access Tokens. For the WFS, the GetFeature request is performed with Access Token in order to get the thematic attributes of the clicked building.

Furthermore, column 5 shows the average response times for each request performed (a) with respective security layer, and (b) without security layer implementations. The last column gives the latency added due to the usage of secured interfaces. The results show that the additional time (latency) caused by the secured interfaces using SAML session cookies is at most a few milliseconds, which shows a minimal impact on the overall performance of the application. However, the use of access tokens introduces a latency for token validation of approx. 20–40 ms. Both SOS are hosted on virtual machines based on Linux OS with 4 virtual CPUs, 8 GB RAM hosted on a VMWare ESXI 5.1.0 Server with 8 CPUs, 2 Processor Sockets, 4 Cores per Socket running with 2.27 GHz. Please note that this paper is about security and not performance and we have included this section only to show

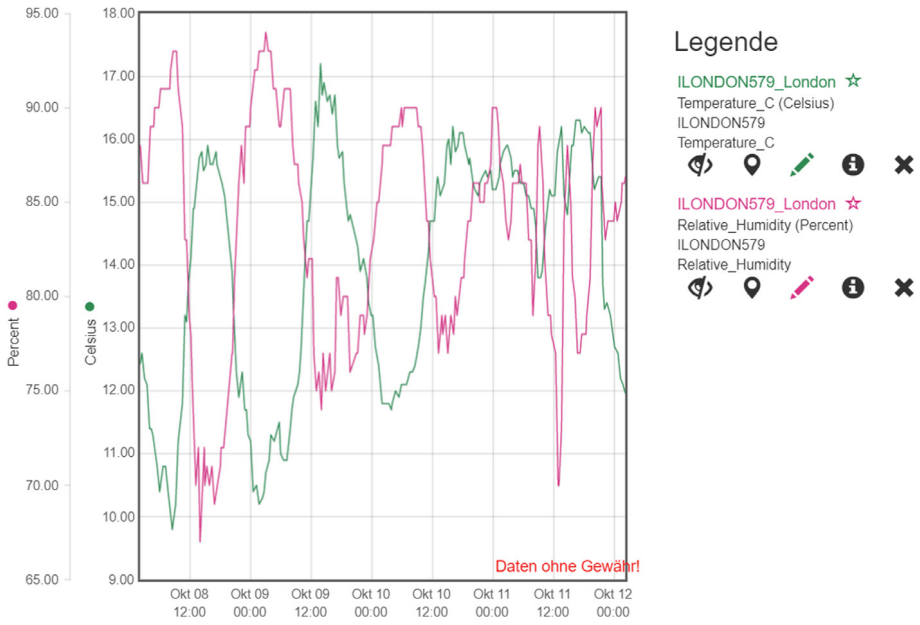

Fig. 9. SOS1 can be accessed by the user with a valid Google and eduGAIN login.

Table 2
Comparison between total response times when querying the different services with and without security layers in place. The total response times shown are the average of 100 requests that were made against each service. The SOS GetObservation is queried for the duration of 15 days. The additional time (latency) by the secured interfaces using SAML session cookies is at most a few milliseconds, which shows a minimal impact on the overall performance of the application. However, the use of access tokens introduces a latency for token validation of approx. 20–40 ms.

Service	Request	Total payload (kB)	Response time (ms)		Latency added (ms)
			With security layer	Without security layer	
SOS1	GetObservation + Access Token	36.8	728	704	24
SOS1	GetObservation + SAML Session Cookie	36.8	705	704	1
SOS2	GetObservation + Access Token	18.6	512	498	14
SOS2	GetObservation + SAML Session Cookie	18.6	501	498	3
WFS	GetFeature + Access Token	4.2	268	233	35

the effects of the additional security checks on the interactive performance.

6. Conclusion and future work

This paper identifies key requirements of developing and securing Spatial Data Infrastructures (SDI) for Smart City scenarios. Based on the proposed Smart District Data Infrastructure (SDDI) framework, the paper presents a new concept for securing the data access and integration of distributed Smart City applications, services, simulation and analytical tools, sensors and IoT devices, and geographic information. The approach meets the identified key requirements and utilizes a novel integration of modern standards such as OAuth, SAML, and OpenID Connect in facilitating ways to authorize and authenticate these distributed components without the need of repetitive logins. To the best of our knowledge, none of the research projects and implementations fulfill all the requirements that we listed for securing SDIs for Smart Cities.

This paper also demonstrates an implementation of the concept for a specific scenario carried out within the district Queen Elizabeth Olympic Park in London and allows to link different components in the project such as 3D buildings with semantic information, weather stations, and Smart Meters in open, standardized and secure ways. The application supports individual access rights for different types of user groups including (i) public Google account, and (ii) academic users from universities and research institutes using eduGAIN accounts. Please note that these IDPs are selected for demonstration purposes only. In the production environment, IdPs of the involved stakeholder organizations will be utilized.

An important aspect is the GDPR (General Data Protection Regulation) which came into force in Europe in May 2018. The prototype implementation demonstrates support of compliant processing of personal data and in particular ensures the data minimization requirement defined by the GDPR.

A next step could be the deployment of the developed concept to production services operated by different organizations involved in the project such as London Legacy Development Corporation in Queen Elizabeth Olympic Park, London.

One important aspect to implement in the future would be the fine-grained access control to geo-located resources provided by secured services similar to the implementation of the Geospatial eXtensible Access Control Markup Language (GeoXACML).[5] Such security features will be based on requirements of the service providers and application developers and users in SSD deep dive districts.

[5] www.opengeospatial.org/standards/geoxacml.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.future.2019.07.002.

Acknowledgments

This work has been carried out within the project Smart Sustainable Districts funded by the Climate-KIC of the European Institute of Innovation and Technology (EIT). We acknowledge Climate-KIC and the EIT for supporting this work. We thank our project partners London Legacy Development Corporation, Imperial College London, and Engie for providing access to sensors installed in the Queen Elizabeth Olympic Park, London. We also thank the reviewers for their critical remarks and constructive suggestions helping in the further improvement of the paper.

References

[1] A. Degbelo, C. Granell, S. Trilles, D. Bhattacharya, S. Casteleyn, C. Kray, Opening up smart cities: Citizen-centric challenges and opportunities from GIScience, ISPRS Int. J. Geo-Inf. 5 (2) (2016) http://dx.doi.org/10.3390/ijgi5020016.

[2] IBM. Smarter Cities - Future cities. 2017. URL: https://www.ibm.com/smarterplanet/us/en/.

[3] Microsoft. CityNext. 2017. URL: https://partner.microsoft.com/en-US/Solutions/CityNext.

[4] CISCO. The Internet of Everything for Cities. 2017. URL: https://partner.microsoft.com/en-US/Solutions/CityNext.

[5] Climate-KIC. Smart Sustainable Districts. 2017. URL: http://www.climate-kic.org/areas-of-focus/urban-transitions/our-initiatives/smart-sustainable-districts/.

[6] CitySDK. City Service Development Kit. 2017. URL: http://www.citysdk.eu/.

[7] CEDUS. City Enabler for Digital Urban Services. 2018. URL: www.cedus.eu.

[8] H.J. Aalders, H. Moellering, Spatial data infrastructure, in: Proceedings of the 20th International Cartographic Conference, Beijing, China, 2001, pp. 2234–2244, URL https://icaci.org/files/documents/ICC_proceedings/ICC2001/icc2001/file/f14005.pdf.

[9] M. Moshrefzadeh, K. Chaturvedi, I. Hijazi, A. Donaubauer, T.H. Kolbe, Integrating and managing the information for smart sustainable districts - the smart district data infrastructure (SDDI), in: T.H. Kolbe, R. Bill, A. Donaubauer (Eds.), Geoinformationssysteme 2017 – Beiträge Zur 4. Münchner GI-Runde, Wichmann Verlag, 2017, pp. 1–19.

[10] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, X.S. Shen, Security and privacy in smart city applications: Challenges and solutions, IEEE Commun. Mag. 55 (1) (2017) 122–129, http://dx.doi.org/10.1109/MCOM.2017.1600267CM.

[11] A. Bartoli, M. Hernández-Serrano J. Soriano, M. Dohler, A. Kountouris, D. Barthel, Security and privacy in your smart city, in: Proceedings of the Barcelona Smart Cities Congress, Vol. 292, 2011, pp. 1–6.

[12] L. Cui, G. Xie, Y. Qu, L. Gao, Y. Yang, Security and privacy in smart cities: Challenges and opportunities, IEEE Access 6 (2018) 46134–46145, http://dx.doi.org/10.1109/ACCESS.2018.2853985.

[13] M. Sookhak, H. Tang, Y. He, F.R. Yu, Security and privacy of smart cities: A survey, research issues and challenges, IEEE Commun. Surv. Tutor. (2018) 1, http://dx.doi.org/10.1109/COMST.2018.2867288.

[14] A. Gharaibeh, M.A. Salahuddin, S.J. Hussini, A. Khreishah, I. Khalil, M. Guizani, et al., Smart cities: A survey on data management, security, and enabling technologies, IEEE Commun. Surv. Tutor. 19 (4) (2017) 2456–2501, http://dx.doi.org/10.1109/COMST.2017.2736886.

[15] K. Biswas, V. Muthukkumarasamy, Securing smart cities using blockchain technology, in: IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016, pp. 1392–1393, http://dx.doi.org/10.1109/HPCC-SmartCity-DSS.2016.0198.

[16] K. Chaturvedi, A. Matheus, S.H. Nguyen, T.H. Kolbe, Securing spatial data infrastructures in the context of smart cities, in: International Conference on Cyberworlds (CW), 2018, pp. 403–408, http://dx.doi.org/10.1109/CW.2018.00078.

[17] D. Hardt, The OAuth 2.0 Authorization Framework. 2012. URL: https://www.ietf.org/rfc/rfc6749.txt.

[18] N. Sakimura, J. Bradley, M. Jones, B. de Medeiros, Mortimore C., OpenID Connect Core 1.0 incorporating errata set 1. 2014. URL: https://openid.net/specs/openid-connect-core-1_0.html.

[19] S. Cantor, J. Kemp, R. Philpott, E. Maler, Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0. 2005. URL: https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf.

[20] G. Gröger, T.H. Kolbe, C. Nagel, K.H. Häfele, OGC City Geography Markup Language (CityGML) Encoding Standard | Version 2.0.0 | OGC Document No. 12-019. 2012. URL: http://www.opengeospatial.org/standards/citygml.

[21] M. Botts, Sensor Model Language (SensorML) | OGC Document No. 12-000. 2014. URL: http://www.opengeospatial.org/standards/sensorml.

[22] P. Vretanos, OpenGIS Web Feature Service 2.0 Interface Standard (WFS) | OGC Document No. 12-100. 2010. URL: http://www.opengeospatial.org/standards/wfs.

[23] A. Bröring, C. Stasch, J. Echterhoff, Sensor Observation Service Interface Standard (SOS) | OGC Document No. 12-006. 2012. URL: http://www.opengeospatial.org/standards/sos.

[24] S. Liang, C.Y. Huang, T. Khalafbeigi, SensorThings API Part 1: Sensing | OGC Document No. 15-078r6. 2015. URL: http://docs.opengeospatial.org/is/15-078r6/15-078r6.html.

[25] N. Douglas, U. Voges, L. Bigagli, Catalogue Services 3.0 - General Model | OGC Document No. 12-168r6. 2014. URL: http://docs.opengeospatial.org/is/12-168r6/12-168r6.html.

[26] K. Chaturvedi, T.H. Kolbe, Towards establishing cross-platform interoperability for sensors in smart cities, Sensors 19 (3) (2019) 562, http://dx.doi.org/10.3390/s19030562.

[27] K. Chaturvedi, T.H. Kolbe, Intersensor service: Establishing interoperability over heterogeneous sensor observations and platforms for smart cities, in: IEEE International Smart Cities Conference (ISC2), 2018, pp. 1–8, http://dx.doi.org/10.1109/ISC2.2018.8656984.

[28] Z. Yao, C. Nagel, F. Kunde, G. Hudra, P. Willkomm, A. Donaubauer, et al., 3DCityDB - A 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML, Open Geospatial Data Softw. Stand. 3 (1) (2018) 5, http://dx.doi.org/10.1186/s40965-018-0046-7.

[29] T.H. Kolbe, Representing and Exchanging 3D City Models with CityGML, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-540-87395-2, 2009, pp. 15–31, http://dx.doi.org/10.1007/978-3-540-87395-2_2.

[30] A. Borrmann, M. König, C. Koch, Beetz J., Building Information Modeling, Technologische Grundlagen Und Industrielle Anwendungen: Vieweg+ Teubner Verlag, 2015, http://dx.doi.org/10.1007/978-3-319-92862-3.

[31] G. Percivall, OGC Smart Cities Spatial Information Framework | OGC Document No. 14-115. 2015. URL: https://portal.opengeospatial.org/files/?artifact_id=61188.

[32] OGC. Interoperability Program. 2016. URL: http://www.opengeospatial.org/ogc/programs/ip.

[33] FCP1. OGC Future City Pilot Phase 1. 2017. URL: http://www.opengeospatial.org/projects/initiatives/fcp1.

[34] J.P. Exner, P. Elisei, Smart Cities and standards–The approach of the Horizon2020-project ESPRESSO, in: REAL CORP 2016–SMART ME UP! how to Become and how to Stay a Smart City, and Does this Improve Quality of Life? Proceedings of 21st International Conference on Urban Planning, Regional Development and Information Society, CORP–Competence Center of Urban and Regional Planning, 2016, pp. 937–943.

[35] SCIRA. OGC Smart City Interoperability Reference Architecture. 2018. URL: www.opengeospatial.org/projects/initiatives/scira.

[36] D. Bhattacharya, M. Painho, Smart cities intelligence system (SMACiSYS) integrating sensor web with spatial data infrastructures (SENSDI), in: ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 4, 2017, pp. 21–28.

[37] S. Meiling, D. Purnomo, J. Shiraishi, M. Fischer, T.C. Schmid, MONICA in hamburg: Towards large-scale iot deployments in a smart city, in: European Conference on Networks and Communications (EuCNC), 2018, pp. 224–229, http://dx.doi.org/10.1109/EuCNC.2018.8443213.

[38] M. Grothe, J.V. Broecke, L. Carton, H. Volten, R. Kieboom, Smart emission - building a spatial data infrastructure for an environmental citizen sensor network, vol. 1762, in: S. Jirka, C. Stasch, A. Hitchcock (Eds.), Proceedings of the Geospatial Sensor Webs Conference 2016, CEUR-WS.org, 2016, pp. 1–7, URL http://ceur-ws.org/Vol-1762/Grothe2.pdf.

[39] S. Trilles, A. Luján, s. Belmonte, R. Montoliu, J. Torres-Sospedra, J. Huerta, Senviro: A sensorized platform proposal using open hardware and open standards, Sensors 15 (3) (2015) 5555–5582, http://dx.doi.org/10.3390/s150305555.

[40] F.S. Ferraz, C. Sampaio, C. Ferraz, Towards a smart-city security architecture proposal and analysis of impact of major smart-city security issues, in: SOFTENG 2015: The First International Conference on Advances and Trends in Software Engineering Information, 2015, pp. 108–114.

[41] P. Lämmel, N. Tcholtchev, I. Schieferdecker, Enhancing cloud based data platforms for smart cities with authentication and authorization features, in: Companion Proceedings of The10th International Conference on Utility and Cloud Computing, in: UCC '17 Companion, ACM, New York, NY, USA, ISBN: 978-1-4503-5195-9, 2017, pp. 167–172, http://dx.doi.org/10.1145/3147234.3148087.

[42] T.Q. Thanh, S. Covaci, B. Ertl, P. Zampognano, An integrated access control service enabler for cloud applications, in: R. Doss, S. Piramuthu, W. Zhou (Eds.), Future Network Systems and Security, Springer International Publishing, Cham, ISBN: 978-3-319-19210-9, 2015, pp. 101–112.
[43] J. Hernández-Serrano, J.L. Muñoz, A. Bröring, O. Esparza, L. Mikkelsen, W. Schwarzott, et al., On the road to secure and privacy-preserving iot ecosystems, in: I. Podnar Žarko, A. Broering, S. Soursos, M. Serrano (Eds.), Interoperability and Open-Source Solutions for the Internet of Things, Springer International Publishing, Cham, ISBN: 978-3-319-56877-5, 2017, pp. 107–122.
[44] FIWARE. Security Architecture Overview. 2016. URL: https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/Security.
[45] G. Suciu, A. Vulpe, S. Halunga, O. Fratu, G. Todoran, V. Suciu, Smart cities built on resilient cloud computing and secure internet of things, in: 19th International Conference on Control Systems and Computer Science, 2013, pp. 513–518, http://dx.doi.org/10.1109/CSCS.2013.58.
[46] J. Harrison, Authentication for OGC Web ServicesOGC Document No. 10-192. 2011. URL: http://www.opengeospatial.org/projects/initiatives/authie.
[47] A. Matheus, How single-sign-on improves the usability of protected services for geospatial data, in: Fifth International Conference on Computing for Geospatial Research and Application, 2014, pp. 95–99, http://dx.doi.org/10.1109/COM.Geo.2014.4.
[48] C.I. Higgins, J. Williams, D.G. Leibovici, I. Simonis, M.J. Davis, C. Muldoon, et al., Citizen OBservatory WEB (COBWEB): A generic infrastructure platform to facilitate the collection of citizen science data for environmental monitoring, Int. J. Spat. Data Infrastruct. Res. 11 (2016) 20–48, http://dx.doi.org/10.2902/1725-0463.2016.11.art3.
[49] B. Resch, B. Schulz, M. Mittlboeck, T. Heistracher, Pervasive geo-security – a lightweight triple-a approach to securing distributed geo-service infrastructures, Int. J. Digit. Earth 7 (5) (2014) 373–390, http://dx.doi.org/10.1080/17538947.2012.674562.
[50] L. Lynch, Inside the identity management game, IEEE Internet Comput. 15 (5) (2011) 78–82, http://dx.doi.org/10.1109/MIC.2011.119.
[51] S. Bröring A. Schmid, C.K. Schindhelm, A. Khelil, S. Kabisch, D. Kramer, et al., Enabling IoT ecosystems through platform interoperability, IEEE Software 34 (1) (2017) 54–61, http://dx.doi.org/10.1109/MS.2017.2.
[52] FIWARE. Open source Platform for the Smart Digital Future. 2018. URL: https://www.fiware.org/.
[53] A. van Kesteren, Cross-Origin Resource Sharing. 2014. URL: https://www.w3.org/TR/cors/.
[54] M. Jones, The OAuth 2.0 Authorization Framework: Bearer Token Usage. 2012. URL: https://www.ietf.org/rfc/rfc6750.txt.
[55] J. Richer, OAuth 2.0 Token Introspection. 2015. URL: https://www.ietf.org/rfc/rfc7662.txt.
[56] MrSwitch. A Javascript RESTFUL API library. 2016. URL: https://github.com/MrSwitch/hello.js.
[57] SimpleSAMLphp. An Application written in native PHP that deals with Authentication. 2015. URL: https://simplesamlphp.org/.
[58] SWITCH. SAML2 Discovery Service. 2018. URL: https://www.switch.ch/aai/support/tools/wayf/.
[59] B. Shaffer, A library for implementing an OAuth2 Server in PHP. 2018. URL: https://github.com/bshaffer/oauth2-server-php.

Kanishk Chaturvedi received his M.Sc. degree in Geoinformatics from University of Twente, the Netherlands in 2014. Currently, he is a Research Associate and Ph.D. Candidate at the Chair of Geoinformatics of the Technical University of Munich, Germany. His research is focused on extending Semantic 3D City Models for supporting time-dependent properties. His research interests are Semantic 3D City Models, Smart Cities, Digital Twins, and Internet of Things (IoT).

Dr. Andreas Matheus is the Managing Director of the company Secure Dimensions GmbH located in Munich, Germany. Dr. Matheus is an expert in the field of Geospatial Security. He has been chairing the Security working groups in the Open Geospatial Consortium (OGC) for a decade and has participated in numerous European Union projects and OGC testbeds on these topics.

Son H. Nguyen received his M.Sc. degree in Computer Science from Technical University of Munich, Germany in 2017. Currently, he is a Research Associate and Ph.D. Candidate at the Chair of Geoinformatics of the Technical University of Munich, Germany. His current research is focused on Spatio-Semantic Comparison of Large 3D City Models using a Graph database. His research interests are Semantic 3D City Models, Graph Databases, 3D Web GIS, and Machine Learning.

Prof. Thomas H. Kolbe is a full professor and head of the Chair of Geoinformatics at the Technical University of Munich, Germany. The research field of Prof. Kolbe is the development of methods for the spatial, temporal and semantic modeling, storage, analysis and visualization of the environment. Key areas are virtual 3D city and landscape models, city system modeling, Smart Cities, 3D geodatabases, 3D geoinformation systems, GIS & simulations and indoor navigation. Prof. Kolbe is the initiator and co-author of the international standard CityGML for semantic 3D city and landscape models.