A Decentralized Smart City Using Solid and Self-Sovereign Identity

Biagio Boi(✉), Marco De Santis, and Christian Esposito

University of Salerno, Fisciano, Italy
{bboi,esposito}@unisa.it, m.desantis29@studenti.unisa.it

Abstract. In the Internet of Things (IoT) context, a considerable quantity of data flows from sensors to centralized servers, holding sensitive information related to users. Unfortunately, how servers store these data instances is usually poorly documented and does not offer any transparency to the users but may pave the way to possible privacy violations. Web decentralization is a prominent solution to cope with these issues and legal obligations regarding data protection so that multiple domains are progressively adopting it as the principal technological enabler. The IoT is not among them yet, as a centralized approach is still the most common one; however, moving data location away from servers to prefer gateways or directly to devices closer to users and under their direct control can realize a more decentralized approach and alleviate the issues related to performance, throughput as well as data protection. This paper aims to exploit existing data decentralization solutions, like Social Linked Data (Solid), to define a more distributed data management for IoT and propose a proof-of-concept implementation of a Smart City platform where users can store and directly manage data produced by public or private IoT devices. Despite providing decentralized data handling, Solid is still affected by a centralized identity management and authentication implementation represented by OpenID Connect. Therefore, to fulfill our vision of a decentralized IoT, we also investigate how decentralizing authentication within Solid and a new user-centric approach based on Self-Sovereign Identity (SSI) represents a promising solution.

Keywords: Web Decentralization · Solid · Smart City

1 Introduction

Integrating intelligent sensors within the context of cities and their underlying social processes is a practice that positively impacts costs and efficiency [2]. For example, by analyzing real-time data from sensors and cameras placed in specific crucial locations, it's possible to optimize traffic flow, reduce congestion, and improve public transportation services. In addition, smart cities can also be one enabler for citizen empowerment and enhance the interaction of public administrations with their inhabitants by providing personalized services. Due to the characteristics of such sensors, and their massive horizontal and vertical

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023 Workshops, LNCS 14109, pp. 149–161, 2023.
https://doi.org/10.1007/978-3-031-37120-2_10

scale, it is unfeasible for them to process such data directly on board. Still, it is more reasonable to have them connected among each other by leveraging cellular or mobile networks and/or outsourcing computation and data to external services hosted in the cloud or different virtualized environments. Therefore, it is a practice for IoT devices to continuously send data to external servers, with consequent networking issues that have led to radical transformations, such as 5G to cite the most prominent one.

Problems in such an approach are not only bonded to QoS and technological aspects, but also legal and ethical when collected data are strictly related to the users, so they become instances of personal data. Moreover, collected data are usually stored in centralized databases, subject to possible attacks perpetrated by criminal externals to providers and dishonest employees aimed at stealing sensible information. Furthermore, such collected data can be used to profile a person by tracing down his/her habits for commercial purposes or not only. What is more dangerous is the position taken by some manufacturers which create sensors able to collect undeclared information, without specifying to the final user in which country or who will receive collected data. General Data Protection Regulation (GDPR) plays a vital role in this context by trying to give users control over their data. It defines how applications can collect the data, which limits exists in the data collection, and where a manufacturer can or cannot store user data. Despite this regulation, its application to the concrete case of IoT and smart cities is still fuzzy [10], motivating the EU to issue additional legal tools to discipline better the legal and ethical obligations in these novel ICT contexts.

Over the last few years, the concept of decentralization is taking place, which is the transfer of authority and responsibility from centric servers to decentralized nodes. This shift is motivated by the need to reduce data flows and deal with legal and ethical obligations by restraining outsourcing for personal data. Blockchain is the key technological enabler for decentralization. It can realize a consistent replica of data and state synchronization so that central nodes are removed, and all computations safely and securely occur at local nodes. As a result, we have witnessed a flourishing explosion of academic papers investigating how blockchain-based solutions can be used within the context of IoT and smart cities, where decentralization is made of smart contracts running on blockchain [19]. Using blockchains simplifies handling good performance, reduced network stress, and improved security, but it is not so evident that privacy and legal obligations are also resolved. As proof, frictions between GDPR and blockchain [8] are yet to be completely resolved. Therefore, rather than focusing on GDPR-compliant blockchain-empowered solutions, we turned attention to other existing solutions for decentralization.

Based on the decentralization concept, the major inventor of the Internet - Tim-Bernard Lee, has recently proposed the Solid project, *a decentralized platform for social Web applications*, aimed at changing how users store their data [14]. The architecture defines the concept of *personal online datastore* (pod), an online storage accessible only by the owners without restriction about the

provider. Users can only authorize third-party applications to interact (write or read) with their pod. According to Linked Data Platform (LDP) recommendation [11], data within Solid platforms are exposed by using a RESTful way: POST/PUT commands to create data instances in the pod, PUT/PATCH commands to update existing data, and GET commands to retrieve data from the pod. Solid has a natural application to traditional citizen empowerment services and/or social networks, but it is our belief that it can also be successfully used in the context of IoT; as the huge quantity of sensing data can be stored in the related user pod, rather than outsourced in the cloud or other computing environment. Despite being such an application a natural one for Solid, the current academic literature and industrial practice encompass only a few works trying to extend the Solid platform to IoT devices. As widely known, problems in applying this solution are the limitation imposed by IoT nodes, which cannot directly follow the schema defined by LDP due to their communication protocols, such as Bluetooth, usually oriented to connection-less ones [3].

This paper proposes an architecture that includes IoT devices as active nodes in the Solid ecosystem, paving the way for a user-centric and decentralized Iot for smart city. The motivation behind this research starts from the necessity for an always more interconnected world, populated by myriads of smart devices, to guarantee user privacy, which is currently under the control of centralized servers. The proposed architecture will make possible the inclusion of all kinds of smart devices and is not limited to the context of a smart city but can be adapted for each kind of application of IoT such as ehealth, industry 4.0 and so on. The starting point of our work has been the analysis of the Solid project and its related standards and technologies, to better understand the context and the limitations in terms of requested resources for using such an architecture within the IoT context. In particular, as will be explained in the next section, Solid does not use any decentralized authentication protocol for preserving user privacy or any mapping with a digital version of citizens' identities. Considering that multiple studies have been conducted on digital twins within the context of smart cities [17], our work will propose a completely user-centric existing authentication mechanism. Such a mechanism is based on Self-Sovereign Identity (SSI), which completely matches the digital twin concept by giving citizens a digital representation of their identity. The integration of SSI as an authentication mechanism for the Solid ecosystem can be a revolution in this context, guaranteeing users' complete control over their data. The document is structured as follows:

- In the second section, an overview of the Solid architecture is presented jointly with open challenges, which will be considered during the presentation of the architecture;
- The third section discusses state of art of authentication within the context of smart cities, in conjunction with progress done in IoT decentralization;
- In the fourth section, a proof-of-concept architecture that uses Solid for handling data storage decentralization is proposed within the context of IoT, including a prominent solution for decentralized authentication;

– In the last section, conclusions and future developments are discussed, with pros in the adoption of such architecture and open challenges.

2 Solid Overview

Solid was born as a solution for privacy concerns within the context of centralized social web applications and platforms, where a massive number of problems exist. This technology aims to provide data independence and a simple data management system by letting the users store data everywhere they want. As discussed in the introduction, Solid does not leverage blockchain for handling data decentralization, in favor of a mechanism that let the user the choice of where to store their data. In what follows, the principal components of such an ecosystem are presented, with a focus on techniques used for data storage, user identification, authorization, and authentication. Finally, open challenges and possible resolutions to cope with decentralization are discussed.

2.1 Architecture

Solid gives users complete control over their data; one can locate their data in a self-hosted server and some others can locate their data in a server ruined by big companies. This concept is possible thanks to the abstraction of pod, which, as said in the introduction, can be seen as the deposit of data created by the web applications. Data stored in such spaces are well structured and easily reusable and interactable by applications. Solid uses Resource Descriptor Framework (RDF) and Semantic Web technologies to store high-quality data; but agrees also with unstructured data (discourage this approach due to difficulty in reusing such data).

It's possible to uniquely identify an identity using WebID; it is a decentralized identifier that is used by agents to create their own identities by linking such identifiers to a profile document. A profile document is a document containing user information formatted according to RDF serialization, including the location of their pod. As possible to see in Fig. 1 a WebID is similar to a URI, and different types of WebID currently exist, depending on the provider which is hosting the pod. Users are able to give applications authorization at different granularity to their pods using the Access Control List (ACL). In according with Web Access Control (WAC), four access modes are defined: read, write,

https://namesurn.solidcommunity.example/profile/card#

https://id.provider.example/namesurn

Fig. 1. Two types of WebID, where the username is included as subdomain or as path.

append, and full control. In such a way the user can establish who is authorized to write or who is only able to read. The authentication instead, depends on the implementation server.

2.2 Decentralization Challenges

Data produced from applications, according to Solid recommendation [14], are stored directly on the user's pod, guaranteeing users complete control over the data produced, without any possibility for services to store or track user data. The major privacy threats of such a model come from the architecture behind such pods and the authentication mechanisms offered. In Solid users are free to use their own server, or servers that they trust, for hosting their pods, located within a centralized server which can be subject to multiple attacks or to data lock-in. Furthermore, the OIDC protocol does not prevent servers from tracking users' activity, compromising their privacy, while TLS-based authentication is poorly supported and difficult to use on mobile devices. For this reason, two big challenges are still open: the decentralization of **data storage** and the decentralization of **authentication**. An interesting study has been conducted by Parrillo et al. [12], which proposes the Interplanetary File System (IPFS) as a solution for managing data storage using a distributed file system, guaranteeing decouple between pod servers and storage location. Cai et al. [1], instead, try to give a complete view of both challenges by proposing a decentralized schema for both data storage and authentication based on blockchain. In particular, they adopt an off-chain storage repository to help store and encrypt the actual data values while guaranteeing decentralized authentication to such storage using certificateless cryptography. Despite such a solution might be really interesting for the problem of data storage; multiple problems can happen regarding the authentication schema. A strong schema must put in place some mechanism for handling with revocation or updating of user data; if a user lost the private key should be able to recover it or should be able to communicate to the system that he needs a new key pair for the access. Considering the purpose of the paper, which is more interested in user authentication, the proposed architecture will take into consideration this last open challenge by proposing a new schema based on SSI also able to cope with revocation.

3 State of Art on Authentication

The advent of password-less authentication mechanisms has radically changed the way in which users access services while guaranteeing them more control over their credentials. Limited studies have been conducted in trying to adopt these in smart cities, usually characterized by weak authentication mechanisms.

B.-C. Chifor et al. [4] presented a proof-of-concept software solution that uses the FIDO protocol for users to authenticate to their devices. Such a protocol preserves user anonymity without giving the possibility to manufacturers to create a link between different user accounts because the only user-related

information is the FIDO public key and a pseudonym. Also if FIDO can be seen as a good authentication mechanism in the case of private devices, it cannot be a suitable solution for the context of smart cities, since it obligates all citizens to buy FIDO-based hardware to comply with the standard. Other implementations of SSI can be deployed, an alternative approach SSI follows a different concept, which is based on Verifiable Credentials (VCs) enabling password-less authentication at reduced cost without the need for proprietary hardware. Sovrin Network [18] is one of the major providers of SSI, giving a solution based on public permissioned blockchain, where nodes validate transactions for issuing and verification. The adoption of such a solution within the context of smart cities can enable the vision for a digital twin of each citizen, who can use SSI to access public services. Rotunua et al. [13] provide an implementation of SSI for digital identity, which is a starting point for initiatives for using the blockchain as a platform for communications and transactions in the public sector.

Despite the decentralization of user identity management can solve privacy problems while enabling digital identity, multiple threats exist in handling data produced by applications within the IoT context, Tahirkheli et al. [15] highlight threats in smart cities that use Cloud Computing that aim at disrupting IAS-octave. Decentralization of user data can be seen as a resolution of multiple privacy issues currently existing within the application [5], their authentication mechanisms are characterized by centralized authentication or by complex mechanisms making citizens difficult to come to terms with. Considering such problems, a solution that is able to include IoT devices as they are, leveraging on gateways as mediators for decentralized solutions and IoT communication protocols must be taken into consideration. An implementation of such mediation has been developed by Troung et al. [16], which uses FIWARE and Hyperledger Fabric for proposing a secure and decentralized data-sharing framework over blockchains. However, in the FIWARE case, we also see the same issue affecting Solid, where authentication is centralised and can be decentralized with permission blockchain as in [6], to deal with identity management and verification. Despite multiple works exist in the literature, the usage of permissioned blockchain can be critical in terms of scalability; making difficult the adoption of such solutions in dense contexts, such as the smart city one.

As a result of these works it's possible to highlight the advantages of adopting SSI for handling user identity while the necessity for a decentralized schema for managing application data considering the limitations of IoT context. Solid can play the role of a data management system without the need for blockchain. This characteristic paves the way for a more secure and interconnected world. In the next section an overview of this ecosystem is given, fixing the challenges that must be taken into consideration for a fully decentralized system.

4 Methodology

A big challenge in IoT scenarios is related to poor battery equipment and power consumption: manufacturers usually deploy devices with a proprietary communication protocol or with protocols that aim to reduce complexity in favor

of consumption. As a result, devices in the IoT context usually implement a connection-less protocol making them unable to rely on the advantages introduced by RESTful protocols, required by Solid for the operation of data writing.

The proposed architecture will take into consideration such limitations by moving the data writing requests to the pod at the gateway level, leveraging on the possibility offered by the Gateways to produce well-structured data and connection-oriented protocols like HTTP(s).

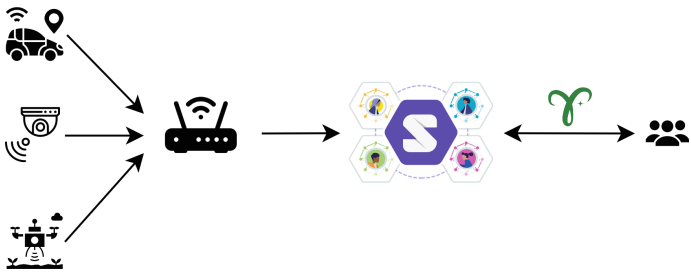

Fig. 2. Proposed Architecture.

Figure 2 depicts the overall architecture, which involves smart devices that use both non-IP and IP protocols creating a secure communication with the gateway, which handles all these communications using mechanisms that support both protocol types, such as FIWARE. As we will explain, the gateway is responsible for producing data in a well-structured form and sending such data directly to the user pod using RESTful mechanisms. To complete our decentralization proposal, an SSI schema based on Hyperledger Aries is taken into consideration for handling user identity, preventing in this way any kind of tracking of user activities.

4.1 Architecture

Before being able to receive data within their pod, citizens must register on the platform. In particular, they have to give the system access to their pod. This is possible using ACL defined within the Solid Server. In Fig. 3 is shown the URL *https://unisa.it* authorized to write into the user pod.

This first step let all the requests coming from the defined domain be accepted for writing in the user pod. At this point, the application is authorized to write to the user pod and must initialize a dataset, or more properly a SolidDataset which stores the data. Data within a dataset are organized in Thing, where each Thing has its own identifier; an example could be https://namesurn.solidcommunity.net/SmartCity#TrafficControl where the SolidDataset SmartCity contains the Thing TrafficControl. Once defined the structure in which data are saved, it's necessary to define a WebID/Things information association. This depends on the nature of the system, if we are interested in traffic control, the information

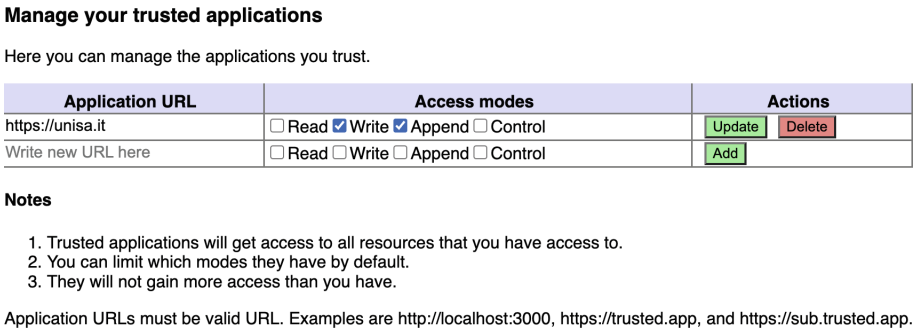

Fig. 3. Applications authorized to interact with user pod.

needed by the system could be the vehicle identifier. Table 1 shows an example of a table that must be stored in a secure place within the system server.

Table 1. WebID - VehID association.

WebID	VehID
https://namesurn.solidcommunity.net/profile/card#	AA123AA
https://id.provider.example/namesurn	AA123AB

The content of such a table can be extended to include all the information read by sensors used by the system. Such a table will be queried by the gateway to check which is the endpoint of red information. In the next subsection, the data collection step is described.

Data Collection and Writing. IoT nodes will not store any kind of data, but just collect them and transmit them to the nearest gateway. These prerequisites are fundamental for the concept of decentralization that we're implementing since the sensors mustn't know which is the owner of produced data, avoiding possible attacks aimed at stealing such association.

As previously introduced, the association table is securely stored at the gateway or server level, which can put in place security mechanisms aimed at preventing attacks. Data produced by IoT devices are sent to the gateway using secure protocols; for example, BLE can be secured using lightweight cryptography based on a symmetrical key. The structure of received data depends on the application, the gateway is responsible for the production of well-structured data according to RDF. Moreover, the gateway can apply some pre-processing techniques before sending data to the user pod.

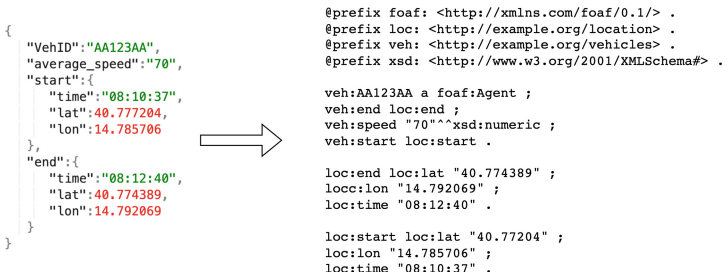

Fig. 4. Transformation from JSON to RDF and serialization according to ttl format.

Figure 4 depicts the transformation from JSON to RDF and the consequent serialization in Turtle (ttl) format of data coming from some smart webcam placed in the city. In such a way application can query users' pod using SPARQL language. To perform data writing, Table 1 is queried with the information received. For example, the gateway receiving data related to VehID will query the table with the related identifier receiving as a response the WebID of the user pod. Once the gateway obtains the WebID related to the VehID an authentication is needed to validate the identity of the application. An existing implementation based on token is offered by Enterprise Solid Server (ESS) which agrees users to register a new application by releasing a new token. Such a token is used by applications for writing or reading requests. This approach makes the authentication of applications dependent on a centralized identity provider without any kind of decentralization mechanisms. An SSI-based approach could solve this issue, by deleting the token-based one.

4.2 Solid-SSI Integration

The decentralized application is now able to write to user pod, but to fulfill our vision of a decentralized IoT it's necessary to do something more on authentication of both users and devices. As described in the previous section, Solid authentication is based on centralized methods, which means that every time a user makes a request, a track of user data can be recorded on the authentication server for advertising or tracking purposes. In our proposal users' authentication leverage SSI methods for guaranteeing users' privacy without interacting with the issuer for the verification of attributes related to the identity but relying on a self-verification system. The concept of Verifiable Credentials (VCs) completely matches such a schema, providing self-verifiable attributes using the cryptographic property. The schema to be followed is depicted in Fig. 5, where three authors (Issuer, Holder, and Verifier) interact with a Verifiable Data Registry for the operation related to the identity. By considering the Solid ecosystem, it's possible to assume that Issuer and Verifier are the same actor, namely, the Solid server, responsible for the issuing of VCs and subsequentially for their verification. A really interesting implementation of SSI based on VCs, able to handle revocation problems and data updates is characterized by Hyperledger Aries,

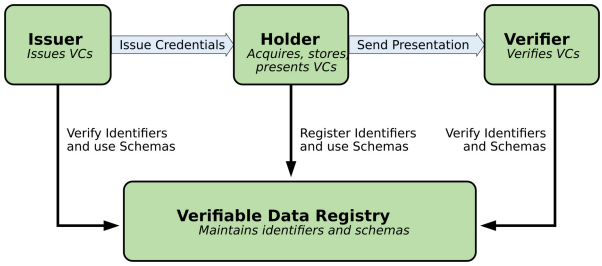

Fig. 5. VC Ecosystem defined by W3C, source: https://www.w3.org/TR/vc-data-model/

which is a project based on Hyperledger Indy for generating and managing Verifiable Credentials (VCs). It's possible to integrate SSI within an authentication mechanism by storing the relevant information in a verifiable way. What's relevant in the Solid ecosystem is the WebID of users, which is released by a Solid server during the registration process and is used by the owner for the identification when needs to authenticate. Hyperledger Aries offers a mechanism based on HTTP API for interacting with Indy wallet, which contains VCs.

WebID Issuing. To include SSI as an authentication means for Solid ecosystem it's necessary to issue the WebID at the registration phase. When a user wants to get a pod from a Solid server must enter his email and choose a username that is used for composing the WebID; for including SSI, it's necessary that the user saves the VC in his wallet to make identity verification possible. Once the Solid server authorizes the registration of a new user, shows a QR Code on the page that must be scanned by the Holder for creating a secure connection with the server. After a secure connection has been established, the Server sends a Credential Offer to the Holder containing the WebID, if the Holder accepts then the data public-facing claims are stored in the Verifiable Data Registry in order to make aware all Solid servers of this new WebID. Such issuing phase can be extended for the authentication of devices by issuing a verifiable token usable for writing and reading requests.

WebID Verification. Differently from password-based authentication mechanisms, which request the user a password, the authentication using a Verifiable WebID can be done by transmitting the proof related to the credential. Hyperledger Aries offers HTTP API for handling such credentials exchange making possible the interaction with a mobile app. Figure 6 depicts the authentication panel including a QR Code which initializes the connection between Holder and Verfier, similarly to what happens during the WebID Issuing phase.

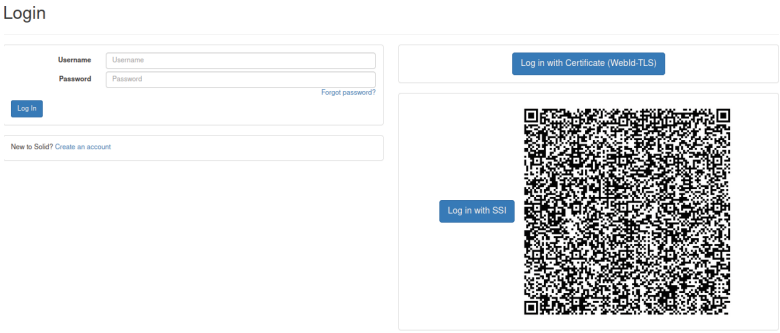

Fig. 6. Authentication form of Solid-SSI compliant server.

Once the connection has been established, the Verifier sends a request containing the requested field and the non-revocation in a given time field. In such a way, the Verifier will be able to check the user's WebID while guaranteeing that no one revoked this VC, which can be useful in case of an identity thief. Once the Holder receives the request can accept it by sharing a cryptographically verifiable proof of possession in conjunction with the requested attribute. When the Verifier receives the proof, using the Hyperledger Aries checks, by consulting the Verifiable Data Registry, if the produced proof is valid.

4.3 Discussion

The adoption of a decentralized approach for handling data produced by sensors can pave the way for a world where each citizen is the owner of data produced by himself. The application that can be built on top of Solid are multiple, for example, a machine learning model that uses data coming from sensors for giving citizens advice on which is the best road considering their habits. Such an approach can facilitate the role of stakeholders interested in users' data by simplifying the request for permission to access to a part of users' data following the WAC and ACL defined by users. More structured data can give access to different granularities, on the basis of user choice.

The other key advantage is that it enables citizens to have a single, unified identity across different services and applications. The adoption of SSI is not limited to Solid but can be adopted also for all the other public services simply by defining the attributes needed for the identification, guaranteeing in this way a more interoperable ecosystem based on SSI. This not only reduces the burden of managing multiple identities but also helps to prevent identity theft and fraud. Furthermore, our approach ensures that residents have full control over their personal data, and can choose what information to share and with whom.

However, there are also challenges associated with building a decentralized smart city using Solid and SSI technologies. For example, there may be technical challenges associated with integrating different systems and applications, and ensuring that they can communicate with each other securely and efficiently.

Data produced by the gateways must be serialized in an RDF-compliant format and require the usage of vocabularies but only a few of them are available at the time of writing. Some future development of them must include the smart city scenario, including vocabularies for sensing temperature, smart agriculture, and so on.

5 Conclusion

A proof-of-concept of a decentralized smart city based on Social Linked Data platform has been proposed and analyzed considering the recommendation of LDP regarding the RESTful way of handling data requests, while decentralized user authentication has been proven to be directly applicable to the Solid platform. We have planned to assess the security of this architecture, performing simulations of malicious attacks on the proposed system and possibly assessing the formal security of the protocol adopted for credentials exchange. Further improvement, instead, can be made on the direct support for leveraging on Web of Things [9] by non-IP protocols [7], making the devices directly responsible for writing on users' pod without the necessity for a mediator. An open challenge in Solid regards centralized data storage, which limits also the usage of SSI. Despite a Verifiable Data Registry is available from all the Solid servers, user data are stored within a pod, which is strictly connected to just one server.

Acknowledgement. This work was partially supported by project SERICS (PE00000014) under the NRRP MUR program funded by the EU - NGEU.

References

1. Cai, T., Chen, W., Yu, Y.: BCSolid: a blockchain-based decentralized data storage and authentication scheme for solid. In: Zheng, Z., Dai, H.-N., Tang, M., Chen, X. (eds.) BlockSys 2019. CCIS, vol. 1156, pp. 676–689. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-2777-7_55
2. Camero, A., Alba, E.: Smart city and information technology: a review. Cities **93**, 84–94 (2019)
3. Chituc, C.M.: Towards seamless communication in the web of things: are standards sufficient to ensure interoperability? In: 2020 13th International Conference on Communications (COMM), pp. 427–431 (2020). https://doi.org/10.1109/COMM48946.2020.9141996
4. Dissanayake, D., Abineshh, U., Zihara, M., Kuruppu, S., De Silva, D., Vidhanaarachchi, S.: Enhancement to smart cities in Sri Lanka using cutting edge technologies. Int. J. Eng. Manage. Res. **12**(5), 488–493 (2022)
5. Dwivedi, A.D., Srivastava, G., Dhar, S., Singh, R.: A decentralized privacy-preserving healthcare blockchain for IoT. Sensors **19**(2), 326 (2019)
6. Esposito, C., Ficco, M., Gupta, B.B.: Blockchain-based authentication and authorization for smart city applications. Inf. Process. Manage. **58**(2), 102468 (2021)
7. Freund, M., Dorsch, R., Harth, A.: Applying the web of things abstraction to bluetooth low energy communication. arXiv preprint arXiv:2211.12934 (2022)

8. Humbeeck, A.V., et al.: The blockchain-GDPR paradox. J. Data Protect. Priv. **2**(3), 208–212 (2019)
9. Kovatsch, M., Matsukura, R., Lagally, M., Kawaguchi, T., Toumura, K., Kajimoto, K.: Web of things (wot) architecture (2020). https://www.w3.org/TR/2020/REC-wot-architecture-20200409
10. Lučić, D., Boban, M., Mileta, D.: An impact of general data protection regulation on a smart city concept. In: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 0390–0394. IEEE (2018)
11. Malhotra, A., Arwe, J., Speicher, S.: Linked data platform specification (2015). https://www.w3.org/TR/ldp/
12. Parrillo, F., Tschudin, C.: Solid over the interplanetary file system. In: 2021 IFIP Networking Conference (IFIP Networking). pp. 1–6 (2021). https://doi.org/10.23919/IFIPNetworking52078.2021.9472772
13. Rotună, C., GHEORGHIĂ, A., Zamfiroiu, A., Smada Anagrama, D.: Smart city ecosystem using blockchain technology. Informatica Economica. **23**(4) (2019)
14. Sambra, A.V., et al.: Solid: a platform for decentralized social applications based on linked data. MIT CSAIL & Qatar Computing Research Institute, Technical report (2016)
15. Tahirkheli, A.I., et al.: A survey on modern cloud computing security over smart city networks: threats, vulnerabilities, consequences, countermeasures, and challenges. Electronics **10**(15), 1811 (2021)
16. Truong, H.T.T., Almeida, M., Karame, G., Soriente, C.: Towards secure and decentralized sharing of iot data. In: 2019 IEEE International Conference on Blockchain (Blockchain), pp. 176–183 (2019). https://doi.org/10.1109/Blockchain.2019.00031
17. White, G., Zink, A., Codecá, L., Clarke, S.: A digital twin smart city for citizen feedback. Cities **110**, 103064 (2021)
18. Windley, P.: How sovrin works. Sovrin Foundation, pp. 1–10 (2016)
19. Xie, J., et al.: A survey of blockchain technology applied to smart cities: research issues and challenges. IEEE Commun. Surv. Tutor. **21**(3), 2794–2830 (2019)