CONNECT: CONtextual NamE disCovery for blockchain-based services in the IoT

Vanesa Daza
Pompeu Fabra University
Barcelona, Spain

Roberto Di Pietro
Nokia Bell Labs
Paris, France

Ivan Klimek
Excalibur s.r.o.
Krakow, Poland

Matteo Signorini
Nokia Bell Labs
Paris, France

***Abstract*—The *Internet of Things* is gaining momentum thanks to the provided vision of seamlessly interconnected devices. However, a unified way to discover and to interact with the surrounding smart environment is missing. As an outcome, we have been assisting to the development of heterogeneous ecosystems, where each service provider adopts its own protocol—thus preventing IoT devices from interacting when belonging to different providers. And, the same is happening again for the blockchain technology which provides a robust and trusted way to accomplish tasks —unfortunately not providing interoperability thus creating the same heterogeneous ecosystems above highlighted.**

In this context, the fundamental research question we address is how do we find things or services in the Internet of Things. In this paper, we propose the first IoT discovery approach which provides an answer to the above question by exploiting hierarchical and universal multi-layered blockchains. Our approach does neither define new standards nor force service providers to change their own protocol. On the contrary, it leverages the existing and publicly available information obtained from each single blockchain to have a better knowledge of the surrounding environment. The proposed approach is detailed and discussed with the support of relevant use cases.

I. Introduction

The past few years have witnessed the growth of the Internet of Things (IoT) and the shift from its theoretical-only analysis to practical implementations [1], [2], [3]. The IoT, as a giant-sized network infrastructure, is characterized by its large-scale heterogeneous elements and a highly dynamic behavior. Current models for IoT are heavily focused on developing vertical solutions, limited by hardware/software platforms and support [4]. However, with the estimated explosion of IoT in the coming years there is a need to rethink how IoT can effectively deliver value to the end-users, value that is expected to come from the seamless interactions among the myriads of IoT devices we will be surrounded by. It is then of paramount importance to focus on the *discovery process*, a mechanism that enables applications to access the IoT data without the need to know the actual source of data. Discovery is indeed a fundamental issue, since an inherent characteristic of the IoT is *heterogeneity*, introduced by the huge set of available distinct things, each with different capabilities—such as computational, storage and energy—as well as diversity in the types of data format (audio, video, text, numeric, streams, etc.) and IoT standards (such as the ones on device IDs, data representation, IEEE IoT projects, ITU and ISO IoT) [5]. The diversity in things and the data produced by them pose significant challenges in fulfilling the ambitious dream of a truly interconnected smart world of things.

Current solutions which try to address the IoT discovery issue and to build a truly interconnected world usually focus on the definition of new naming standards and are aimed at the process of “labeling” devices. This is a well-known approach and it is currently used in the classic WEB scenario where a Domain Name Service (DNS) is used to map human readable Uniform Resource Identifiers (URIs) to IP addresses. As a consequence, URIs can be used or coded in applications, scripts and software while actual IP addresses can change over time. However, porting this approach to the IoT will force manufacturers to change their own identifiers (such as serial numbers) and, last but not least, it will remain focused on the data source (i.e. the devices) rather than on the data itself (i.e. the services). Furthermore, IoT manufacturers and administrators could be other entities rather than the service providers. As an example, companies such as Boingo do provide WiFi services even though they do not install the actual access point as they subscribe to a service that another entity installs and maintains. Hence, labeling the access points and enabling the users in finding them within the network do not help in the process of getting access the final service. Furthermore, these solutions usually require trust anchors in the forms of either trusted data or authorities.

To solve the above problem and to provide a distributed solution to realize the interconnection promise, both academic and industrial groups started to use benefits of the emerging blockchain technology [6]. The blockchain is a distributed database that can be used to collect any message sent between devices and any operation accomplished by them. Albeit this approach allows for resource constrained devices to participate and prove their identity within the IoT, it assumes that all the devices be aware of each other existence—in order to first contact a device and later to check for the provided services or applications.

978-1-4673-8999-0/17/$31.00 ©2017 IEEE

Hence, this brings back our original question, how can we use this powerful distributed technology to let things be aware about the surrounding environment (e.g. the available services) without an a priori knowledge of the physical devices.

In this paper we present a solution to the above questions which, leveraging blockchain technology, builds a new contextual discovery scheme capable of identifying a service and later, if needed, the devices that is running it. This solution, named CONNECT, is built over a new hierarchical blockchain structure that removes current boundaries in service provisioning and allows any user to become a provider. Thanks to our proposal, nowadays blockchain-based applications can be tuned into global and local chains that, in turn, are linked to each other. This allows different environments in defining their own blockchain that is then used by other peers in the discovery process. The paper is organized as follows: Section II introduces the blockchain technology and how it can be used to securely identify services; Section II-A lists all the prior works in this field, first focusing on the general problem of discovery and then describing the "naming" approach. In Section III, we introduce our solution which introduces a seamless service discovery on top of the blockchain. Conclusion and future directions are given in Section IV.

II. Background technology and related work

The blockchain technology [7] is something that has recently changed the notion of centralized authorities. It can be roughly seen as a digital ledger that sits at the core of a decentralized ecosystem and keeps track of any changes. The blockchain holds a record of every transaction made by every participant. Furthermore, each them can help in verifying any transaction it is able to listen to in the network, thus providing highly redundant verification of each movement within the decentralized environment. However, while the blockchain technology provides a trust mechanism for distributed environments, it still suffers from weaknesses which lower its adoption in real applications, specially when applied to the IoT and outside of financial contexts.

The first weakness is its scalability. Indeed, as the blockchain technology is able to store all the transactions back to the first block ever created, one global blockchain will be sooner or later hard to manage. The second weakness is about naming and discovery. The blockchain technology has not been designed for the IoT, meaning that peers were not meant to find each other in the network. An example is the Bitcoin application in which the IP addresses of some "seeders" are embedded within the Bitcoin client and used by peers to build the network topology. This approach will not work in the IoT as devices will keep moving all the time thus changing the topology. Last but not least, we have privacy issues. Having one single blockchain allows all the nodes within the network to have access to the transactions of all other nodes. Solutions are trying to solve the privacy problem (such as the work published by Zhumabekuly Aitzhan et al. [8]) but they either leverage private blockchain or obfuscate the content of the message. However, none of them can be used for IoT discovery as, while obfuscating the content makes impossible for other peers to learn about the surrounding environment, making the blockchain private just forces the peers in having a private-public key pair to join the blockchain but do not solve the problem within the blockchain.

The innovation originally provided by the blockchain is about combining a decentralized consensus protocol with a block-based organization of transactions. The blockchain is an ordered back-linked list of blocks carrying transactions. Blocks are data structures that encode the information transferred between participants. In cryptographic currencies, transactions define the flow of virtual coins from one node to another but can be even used to identify the network's *state*, defined as the actual information known or stored by each peer. In Bitcoin such a *state* is defined as the current set of *unspent transaction outputs* (for short, UTXO in Bitcoin) that represents all the possible outputs that the network is still able to process. Each different set of spent/unspent transactions thus create a distinct state of the system.

Peers within the same network have to cooperate in order to agree on the same snapshot and on their order over time. This process, named *state validation*, it is easy to accomplish with access to a central authority (as banks for payment circuits) but hard to achieve in distributed environments. A consensus algorithm among the peers is then required to create and validate subsets of the UTXO named *blocks*. Each block is identified by a hash of its header and contains other information which are of paramount importance in order for the other peers to easily verify its correctness. The main important elements within a block are i) a link to the previous block in the blockchain, ii) a root pointing to a so called Merkle tree [9] used to keep track of all the transaction written in the block and iii) a list of all the transactions.

A. Related Work

Applying the blockchain technology to the IoT world offers different possibilities. As an example, the IBM *Autonomous Decentralized Peer-to-Peer Telemetry* project (for short, ADEPT) [6] leverages the blockchain to build a distributed network of devices. As for the ADEPT project, many other approaches are trying to design a solution that will be able to merge all the different blockchain based applications [10]. However, albeit this is another step towards the design of blockchain interconnected world, all of them are still based on banking scenarios and does not address the problem of service provisioning in the IoT where a huge number of devices, with their own services, will need to communicate with each other. Furthermore, products could have distinct identifiers and use different communication protocols. Therefore, as introduced in Section I for DNS, a mapping between physical and virtual devices will be required. To do so, custom identifiers from different name-spaces such as Jabber-ID [11] or CAN-Bus ID [12] will need a name management infrastructure in order to cooperate. However, this fact will create a heterogeneous ecosystem that will suffer from i) *custom syntaxes* as different ID might be incompatible to each other and ii) *redundancy* as different devices might share the same ID.

Such a unique device-name mapping can be done with three different approaches as follows:

- **Manual Setup**: users manually configure and distribute device public names such as IP addresses. This is rather a theoretical approach but often used for demos, where discovery and identity management is *out of scope*. However, giving to users the burden of device configuration is not realistic in a real scenario;
- **Semi-automatic Setup**: a platform provider can be responsible for the mapping. Users would have to register their devices to the provider and then connect them by using the platform. This implies that all participants are somehow connected to this platform. In such an approach, each new member has to become a part of the solution and has to register with the platform.

Other projects are focused on IoT naming and resolving protocols [13], [14], [15]. However, they focus on the definition of new naming standards and work on solving either networks' security or performance [16] but not both. Such approaches could lead to efficient and secure protocols but they will at the same time either force vendors to change their own solutions [17] or rely on trusted third parties such as gateways [18] or protected data sets [19]. Various industry efforts (such as IoT-A [20]) also focused on the standardization of interoperability. Apple [21] and IBM [22] smart beacons are other examples in which device discovery is achieved by tagging devices and by broadcasting IDs and their associated information. OpenIoT [23] proposed an IoT browser that can connect to a server platform, browse for devices and then download related applications. Similarly, Wang et al. [24] proposed a solution which was more emphasized on service discovery and on notification based on user preferences as opposed to active browsing. In all these studies, some efforts have been done trying to connect IoT devices or services. However, all of them either i)try to define a new standard or ii)leverage trust anchors (such as dedicated servers) in doing the device-service mapping.

Our solution is different from all the above works as we leverage the blockchain technology. This makes our approach unconcerned of the device naming solution adopted while focusing the discovery process only at the application layer.

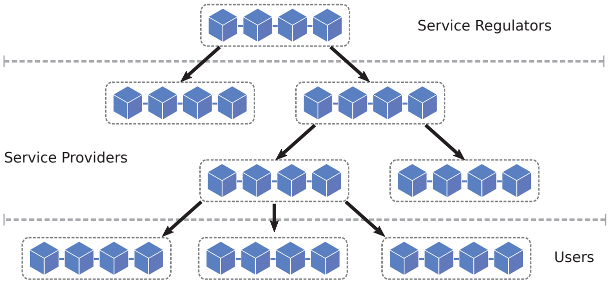

Fig. 1. CONNECT hierarchical blockchain structure where nodes represent blockchains whilst edges administrative relationship

III. CONNECT solution

In this paper we describe a novel solution for CONtextual NamE disCovery for blockchain-based services in the IoT. To do so, first we introduced the concept of "administration" within the blockchain (see Figure 1). An administrator is a peer cooperating in a blockchain application and whose final goal is to keep track and to manage other blockchains changes (such as forks, pruning, etc.) below him. A toy example of this relationship is the one given by service providers and their customers whose access is regulated and controlled. Furthermore, CONNECT extends this basic concepts to a tree structure in which each entity can become a service provider by creating a new blockchain, below him, and by regulating the access to it.

In our hierarchical blockchain tree structure, it is then possible to distinguish three different kinds of participants. *Service regulators* (the tree's root) define services and have the responsibility to link all providers together. As an example, the service regulators could be some governments or institutional organizations that control the activity of major carriers actually providing services. The second kind of participants is the one populated by service providers. Starting from the public and well-known ones, in this portion of the blockchain tree, any entity can start a new branch in the tree and start regulating it as a service provider. Last but not least, we have the third kind of participants composed by users.

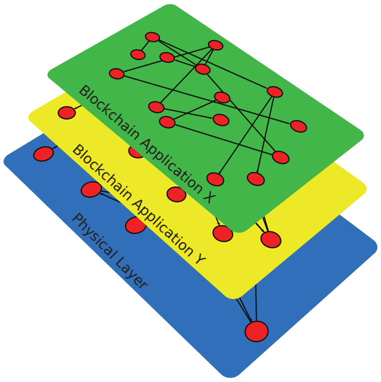

Fig. 2. Hierarchical structure of attributes.

Each single node within the blockchain tree of Figure 1 is a blockchain application in which peers cooperate to share some services (such as money in the Bitcoin application). Hence, all the blockchain applications can be represented as layers within a multi-layered structure (see Figure 2). In such a layered arrangement, there is an initial layer which is "physical" whilst all the other are virtual as they represent blockchain peers (i.e. addresses). Differently from other solutions, in CONNECT, dynamic queries, based on services and their providers' information, are used. Hence, by using service-based queries, devices will not focus on *which device* can actually provide the service they need. This approach, based on a semantic-driven IoT [25], allows non-experts users or casual devices to focus on the intent (i.e. on the service needed) rather than on the specific device.

As in CONNECT the discovery process is built on top of service providers' information, the actual state of each provider needs to be trusted by the whole network. In our approach, which is blockchain-based, each service provider change has to be validated by all the devices with active peer(s) in the same service (i.e. in the same node of Figure 1). As such, every time a device wants to change one or more of its service(s), it will have to send a request for such operations to the specific blockchains. Other devices with active peers in the same blockchain application will then analyze and validate such requests thus writing the new service status within the chain.

CONNECT is able to track services and their current status/state by relying on the blockchain as of a state transition system where there is a *state* consisting of ownership status (such as available bandwidth, video and audio capabilities etc.) and a *state transition function*. As such, taking two peers A and B, the state transition functions moves digital assets X, Y, Z (coins in the example of Bitcoin) from A to B thus changing their ownerships. However, in our solution, we are not interested in the change of ownership but rather on the effects caused by the change.

A. PNodes vs VNodes

In CONNECT, each single blockchain (depicted as dotted boxes in Figure 1) represents a single *layer* within a multi-layered graph (see Figure 2). Two different kinds of nodes have been defined in such a graph:

- **Virtual Nodes**: for short VNodes, are logic nodes in the form of blockchain peers (also known as addresses in the Bitcoin network). These nodes are responsible for managing all the operations that involve the blockchain such as creating and validating transactions as browsing the blockchain to find information on other peers. In Figure 2, VNodes belong to the *blockchain application layers*;
- **Physical Nodes**: for short PNodes, represent the devices. In Figure 2, PNodes belong to the *physical layer*, which is the first one.

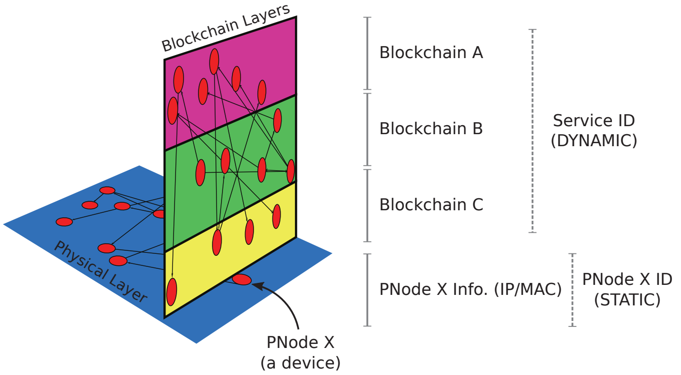

Fig. 3. Example showing PNodes within the blue layer and VNodes belonging to different attribute layers.

Each VNode belongs to a layer which describe a particular blockchain application (in the example of Figure 2, the yellow, green and pink blockchain might be data, audio and video services). Hence, each audio, video or data request will be managed by a blockchain which is located at a certain point on the blockchain tree depending on the service provider. Furthermore, within the multi-layered structure depicted in Figure 2 three different interactions are possible:

- **PNode to PNode**: connection between two physical devices in the network;
- **PNode to VNode**: connection between a physical device and a blockchain peer which can be either executed by the same PNode or by others. Devices locally running virtual nodes already have all their information. Otherwise, they need to reach the other physical node running the peer and download all its information;
- **VNode to VNode**: connection between two peers. For the sake of simplicity and clarity, we will call them VNode$_\text{s}$ and VNode$_\text{r}$ for sender and receiver respectively. If the VNodes belong to the same blockchain, the PNode hosting them just browses the chain and finds the required information. If the VNode$_\text{r}$ belongs to a different blockchain than VNode$_\text{s}$, then the PNode hosting it has to i) download the VNode$_\text{r}$'s blockchain from the client and ii) browse it for the required information.

Figure 3 depicts an example of how VNodes and PNodes are used in our solution. For each PNodes we can take as many *cuts* of the multi-layered structure depicted in Figure 2. Each layer of the cut represents a snapshot of the current configuration of a PNode. Recalling that in this example we are taking into account three services (audio - video - data) and that each service has a tree structure, the snapshot depicted in Figure 3 shows all the peers actually involved in the audio, video and data blockchain which belong to the PNode X. As an example, if we consider the cryptocurrency ecosystem we can see services as different digital currencies and VNodes as multiple wallets hosted and maintained by a single user.

B. Service Discovery

Based on the definition of PNodes, VNodes and on their available communication schemes, we can now describe our discovery process which is composed by the following steps (see Figure 4):

1) In this first step, we can scan the environment sending a *hello message*. If the services required are already known, then we can also specify them. Otherwise, we will send an empty hello message which will cause all the surrounding devices to reply with the information on all their services. The replies from surrounding devices contain the blockchain addresses of the peers which are actually providing the required services on top of the blockchain (these peer addresses are random and do not identify the underlying devices).
2) Once obtained the blockchain peer addresses, we can start browsing the relative blockchains in the cloud to find the last activities of those addresses. At this step we are looking for a proof within the blockchains that supports what the devices claimed by replying to the hello message. In the example depicted in Figure 4, let us suppose that the A, B and C devices replied to our hello message with some information about video, audio and data services (the ones that we are looking for).
3) As we trust the communities which have verified and validated blockchain transactions from A, B and C, we can now create three new transactions (one for each service on a different blockchain) in which we specify the conditions to access the service and we pay for them.
4) Once our transaction are verified and validated by the other peers within the blockchains, A, B and C will be notified thus letting as access the service we paid for.
5) (Optional) If it was not possible to embed all the communications within blockchain requests (created in the third step), we can now communicate on the physical layer. However, it has to be stressed that, this last and optional step is accomplished by using some temporary device ID which can be set up for this exact communication. Hence, this ID is not able to permanently identify the device and other devices cannot use it for further requests.

As a conclusion, in Figure 4 we can see how standard IDs such as IP or MAC addresses are not being used. All the communication are carried on the application level which is secured by the blockchain thus obtaining the following properties:

- Fully Distributed: the blockchain technology has been designed to mitigate any single point of failure. Indeed, in our solution, customers do not have to trust their service providers (opposite to what is done nowadays for Cloud services). Customers and service providers interact on top of the blockchain and their actions are validated by the whole network which is assumed to be honest (at least half of it) and thus, trusted;
- Immutable: the blockchain technology is tamper-proof by design. This means that our discovery process inherits this property thus making almost impossible for a malicious user to over-write, delete or create fake service information in the blockchain.

IV. Conclusion

In this paper we propose a solution for *contextual name discovery for blockchain-based services in the IoT*. To the best of our knowledge, the proposed solution is the first one that is able to solve the naming and discovery open issues in the IoT. More in detail, it allows services to be automatically discovered and accessed at run-time. Compared to other solutions (described in Section II-A), the core novelty of our proposal is the capability of exploiting information within the blockchain in order to define at the application level different identities for the IoT devices. Hence, our solution does not require any static and predefined syntaxes for names. CONNECT enables the identification and connection of devices without knowing their real identities—in the same way this anonymity feature is achieved in the Bitcoin network via wallets. These virtual peers will have their own blockchain addresses and will use them to send/receive transactions. Thus, by using these addresses, we can identify the devices which are providing some services even though they share the same IDs on either the physical or network level. Last but not least, our solution has been designed not only to provide more natural and user-friendly interactions among devices, but also to deliver a more secure and privacy aware environment for the IoT deployment. Ongoing research efforts are aimed at implementing a small scale physical prototype based on the described solution leveraging IoT devices (such as the Raspberry Pi or Arduino) and using MUD (Manufacturer Usage Description YANG Model) for service discovery.

Acknowledgment

This work is partly supported by the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502) and Project TEC2015-66228-P (MINECO/FEDER,UE). Furthermore, the authors would like to thank Identities of Things Discussion Group within Kantara Initiative for their input on open challenges in industrial name resolving and discovery.

References

[1] M. Lee, J. Hwang, and H. Yoe, “Agricultural production system based on IoT,” in *IEEE 16th International Conference on Computational Science and Engineering (CSE)*, Dec 2013, pp. 833–837.

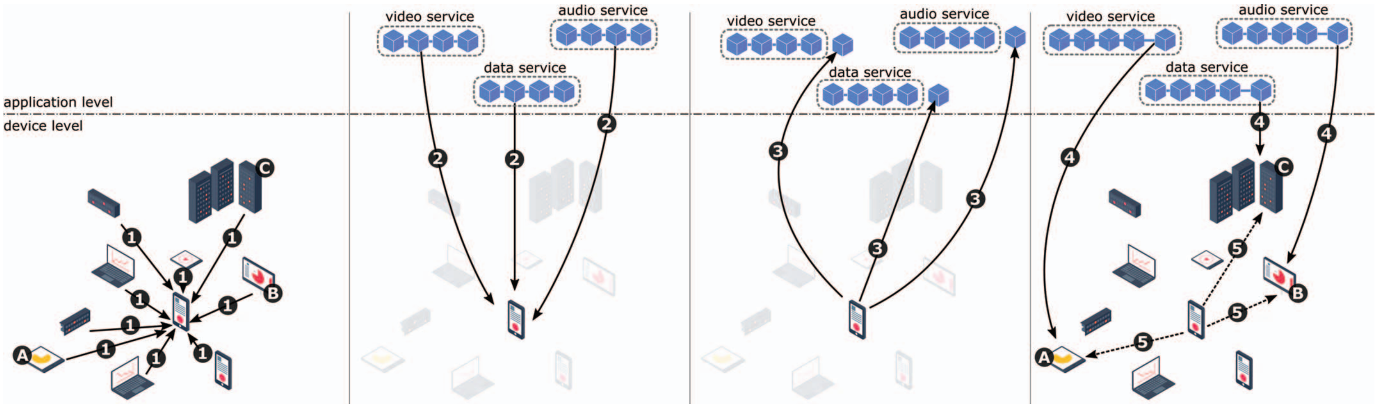

Fig. 4. Blockchain based discovery process

[2] D. Conzon, P. Brizzi, P. Kasinathan, C. Pastrone, F. Pramudianto, and P. Cultrona, "Industrial application development exploiting IoT vision and model driven programming," in *18th International Conference on Intelligence in Next Generation Networks (ICIN)*, Feb 2015, pp. 168–175.

[3] G. Yang, L. Xie, M. Mantysalo, X. Zhou, Z. Pang, L. D. Xu, S. Kao-Walter, Q. Chen, and L.-R. Zheng, "A health-IoT platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2180–2191, Nov 2014.

[4] A. Zaslavsky and P. P. Jayaraman, "Discovery in the internet of things: The internet of things (ubiquity symposium)," *Ubiquity*, vol. 2015, no. October, pp. 2:1–2:10, Oct. 2015. [Online]. Available: http://doi.acm.org/10.1145/2822529

[5] S. Jeschke, C. Brecher, H. Song, and D. B. Rawat, Eds., *Industrial Internet of Things.* Springer International Publishing, 2017.

[6] IBM, "ADEPT: An IoT Practitioner Perspective," 2015.

[7] C. Decker and R. Wattenhofer, "Information propagation in the Bitcoin network," in *IEEE Thirteenth International Conference on Peer-to-Peer Computing (P2P)*, Sept 2013, pp. 1–10.

[8] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Transactions on Dependable and Secure Computing*, vol. PP, no. 99, pp. 1–1, 2016.

[9] A. Shoufan and N. Huber, "A fast hash tree generator for merkle signature scheme," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, May 2010, pp. 3945–3948.

[10] J. H. Ziegeldorf, F. Grossmann, M. Henze, N. Inden, and K. Wehrle, "Coinparty: Secure multi-party mixing of bitcoins," in *Proceedings of the 5th ACM Conference on Data and Application Security and Privacy*, ser. CODASPY. New York, NY, USA: ACM, 2015, pp. 75–86.

[11] F.-C. Chang and D.-K. Chen, "The Design of an XMPP-based Service Integration Scheme," in *Proceedings of the 2011 Seventh International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, ser. IIH-MSP. Washington, DC, USA: IEEE Computer Society, 2011, pp. 33–36.

[12] S. Cai, M. Bakhouya, M. Becherif, J. Gaber, and M. Wack, "An in-vehicle embedded system for can-bus events monitoring," *J. Mob. Multimed.*, vol. 10, pp. pp. 128–140, May 2014.

[13] S. Lee, J. Jeong, and J. Park, "DNS name autoconfiguration for IoT home devices," in *IEEE 29th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, March 2015, pp. 131–134.

[14] Y. Liu, H. Wang, J. Wang, K. Qian, N. Kong, K. Wang, Y. Shi, and L. Zheng, "Enterprise-oriented IoT name service for agriculture product supply chain management," in *International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI)*, Oct 2014, pp. 237–241.

[15] Z. Yan, N. Kong, Y. Tian, and Y.-J. Park, "A universal object name resolution scheme for IoT," in *Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing*, Aug 2013, pp. 1120–1124.

[16] R. Pozza, M. Nati, S. Georgoulas, A. Gluhak, K. Moessner, and S. Krco, "CARD: Context-Aware Resource Discovery for mobile Internet of Things scenarios," in *A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on*, June 2014, pp. 1–10.

[17] M. Antonini, S. Cirani, G. Ferrari, P. Medagliani, M. Picone, and L. Veltri, "Lightweight multicast forwarding for service discovery in low-power IoT networks," in *22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Sept 2014, pp. 133–138.

[18] G. Tanganelli, E. Mingozzi, C. Vallati, and C. Cicconetti, "A distributed architecture for discovery and access in the Internet of Things," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2013, pp. 45–46.

[19] R. Kolcun and J. McCann, "Dragon: Data discovery and collection architecture for distributed IoT," in *Internet of Things (IOT), 2014 International Conference on the*, Oct 2014, pp. 91–96.

[20] S. De, P. Barnaghi, M. Bauer, and S. Meissner, "Service modelling for the internet of things," in *Computer Science and Information Systems (FedCSIS), 2011 Federated Conference on*, Sept 2011, pp. 949–955.

[21] Y. Yang, Z. Li, and K. Pahlavan, "Using iBeacon for intelligent in-room presence detection," in *2016 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, March 2016, pp. 187–191.

[22] A. Akinsiku and D. Jadav, "Beasmart: A beacon enabled smarter workplace," in *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, April 2016, pp. 1269–1272.

[23] J. Kim and J. W. Lee, "OpenIoT: An open service framework for the internet of things," in *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, March 2014, pp. 89–93.

[24] E. Wang and R. Chow, "What can i do here? IoT service discovery in smart cities," in *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, March 2016, pp. 1–6.

[25] C.-J. M. Liang, B. F. Karlsson, N. D. Lane, F. Zhao, J. Zhang, Z. Pan, Z. Li, and Y. Yu, "SIFT: Building an Internet of Safe Things," in *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*, ser. IPSN. New York, NY, USA: ACM, 2015, pp. 298–309.