Combined Certificate and Resource Discovery for Dynamically (Dis-)Aggregating IoT Processes

Frank Engelhardt[1], Mesut Güneş[2]

Abstract: The concepts of Microservices and Organic Computing contribute to a fully distributed architecture of the Internet of Things (IoT), avoiding single points of failure through massive service distribution. The distribution and the lack of structure, however, come with large communication overheads. We discuss the necessity of structure in IoT networks focusing on the problem of trust handling, specifically analyzing the certificate chain discovery problem. Moreover, we provide an argument towards solving the certificate chain discovery problem in the same manner as the service discovery problem in a combined, semi-structured approach. By numerical analysis we show that the introduction of a hierarchy can avoid scalability problems and that resource directories used for service discovery can serve as hierarchical entities.

Keywords: Web of Trust; Certification; Internet of Things

1 Introduction

The modern, yet permanently evolving IoT imposes many changes to our everyday lives as the pervasiveness of intelligent gadgets enriches many sectors at once. The trust that we grow in smart things stems from the benefits that we gain from their intelligence as they improve services and systems that we interact with every day. But as computation becomes more ubiquitous and the interconnection between devices more complex, the verification of communication and computations run by smart devices becomes a huge problem. Trust in services is easy to compromise, especially when networks and systems become so complex that no single controlling instance has complete knowledge about every single system entity. Service aggregation using Organic Computing may play a key role in managing growing complexity in IoT systems [Ro16]. We speak of such IoT systems that use Organic Computing and Microservices to manage their intrinsic complexity as dynamically (dis-)aggregating IoT systems.

It is, however, obvious, that establishment of trust in such systems is far more complex than the verification and validation of each sub-component of an organic system. Communication and aggregation can also be compromised, such that the aggregation as a whole needs to be

[1] Otto-von-Guericke Universität Magdeburg, Communication and Networked Systems (ComSys), Faculty of Computer Science, Universitätsplatz 2, 39106 Magdeburg, Germany, frank.engelhardt@ovgu.de

[2] Otto-von-Guericke Universität Magdeburg, Communication and Networked Systems (ComSys), Faculty of Computer Science, Universitätsplatz 2, 39106 Magdeburg, Germany, mesut.guenes@ovgu.de

doi:10.18420/inf2020_113

trustworthy as well. Exchanging symmetric keys is not an option in huge networks, thus asymmetric encryption needs to be implemented in an efficient and scalable way.

In this paper, we investigate the certification process in IoT networks and discuss the necessity of structure to reduce communication complexity. Since in IoT devices are most commonly resource-constraint, storing certificates in order to establish trust networks is a problem that should be delegated to devices with bigger storage capacity. In numerical analysis, we show that the utilization of such resource directories enables the certificate chain discovery to be solved with reasonable overhead compared to a completely distributed approach.

The rest of this paper is structured as follows. Section 2 gives an introduction to the structure of future of dynamically (dis-)aggregating IoT processes. Section 3 introduces the trust chain management problem. Section 4 summarizes related work. In Section 5, we present a solution considering common communication approaches and the concept of resource directories. Section 6 contains a numerical analysis, and Section 7 concludes with a discussion.

2 Dynamically (Dis-)Aggregating IoT Processes

Microservice architectures are dominating in software development nowadays [AAE16; Na16]. The concept allows for aggregation of applications from small entities of software which are easier to maintain and develop. It is also easy to scale these services as migration cost is low. In the IoT market there is rapid development towards small and flexible services. Applications also migrate towards the end-user in order to decrease latency and improve scalability. Many edge cloud approaches are on the market for that purpose, for example Microsoft's Azure IoT Edge, Akamai IoT Edge Cloud, and Google's Cloud IoT.

The development of these approaches also follows the goal to decentralize server infrastructures and gain independence from big data warehouses. More and more stakeholders in industry aim to have local infrastructure close to their property. Together with the microservice approach, this continuous development improves flexibility, scalability and responsiveness of huge systems. However, the configuration and maintenance overhead for such distributed systems becomes significant. For example, the failure of a single sensor, router, or server can require the migration and duplication of many services in order to replace the failed subsystems. This process must be automated in order to keep complexity low on microservice levels.

Fig. 1 demonstrates such a scenario where services have to be migrated due to a system fault on some entities. To allow automatic migration of services and restoring of functionality, the system under observation and control must be based on a flexible service description and allow a controller to re-assemble functionality from small parts. With organic computing, for example, the overall system can restructure itself according to global rules. This self-organizing behavior allows for managing the growing complexity of several hundreds or

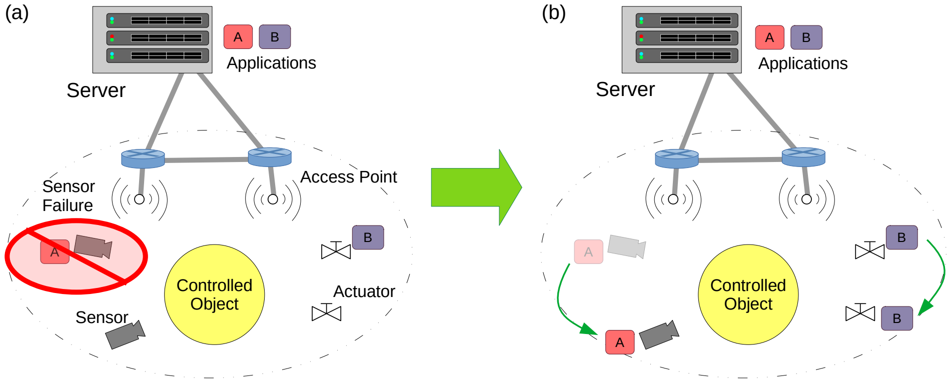

Fig. 1: Migration of a service in a redundant control system due to failure of a sensor (a). The sensor driver A migrates in order to recover from the failure and re-establish system functionality (b). Due to dependencies in the controlled object, also the actuator software has to migrate to another point in the system.

thousands of microservices dispatched on sensors, actuators, routers, and servers. However, such a restructuring demands strict system definitions with capabilities, requirements, and trust models in order to give organic controllers all the necessary information.

3 Trust in Internet of Things (IoT) Networks

Trust is a category that extends beyond security issues [YZV14]. The reliability, availability, resilience and persistence of a system are also contributing to its overall trustworthiness. Systems which generally are considered trustworthy are more often relied upon for in terms of information exchange, relaying, reasoning, or taking actions. For sensors and actuators, for example, their accuracy, precision and proper fault modeling may be considered more important than privacy or confidentiality, as the proper operation of a system depends primarily on those former properties.

In terms of security, however, trustworthiness is an important issue and covers authenticity, confidentiality, integrity and access control. Due to the heterogeneity and huge complexity of IoT networks, key exchange imposes a great challenge to applications. Configuring keys per hand comes with prohibitively high labor cost. Asymmetric encryption via certificates can be a solution to the key exchange problem without sharing secret keys, but opens up the new problem of certificate exchange. A certificate is hereby granted by a signing entity v, containing the information that another entity u has public key P_u.

$$\mathrm{Cert}(u, v) = A_u|P_u|e(h(u|P_u), S_v)$$

Where S_v is the secret key of the issuer v, $e(m, k)$ is an asymmetric encryption function for message m using key k, and $h()$ is a hash function. A_u is the address (or *common name*) associated with u. | denotes the string concatenation.

3.1 Public Key Infrastructure

Public Key Infrastructure (PKI) solves the certificate exchange problem by hierarchical structures and is well-established in the World Wide Web. With PKI, a globally trusted Certificate Authority (CA) issues certificates for users u. The CA thereby has to check the identity of u and then associates P_u with it by issuing the certificate. Every entity wishing to check the identity of u can then take the publicly available certificate and verify it with the public key of the globally trusted CA.

The approach has drawbacks, however. The CA is a single point of failure in several ways. First, it has to be designated and maintained with special effort, imposing configuration maintenance overhead. Second, it must be trusted by everyone, which is especially problematic for heterogeneous IoT networks. Third, the central organization can also impose scalability issues.

3.2 Web of Trust

The Web of Trust (WoT) which was introduced with PGP [Ca07] dismisses the hierarchical PKI idea. Instead, nodes can issue certificates for each other in a peer to peer manner, creating a non-hierarchical network of trust. Trust in an entity increases with more peers have granted certificates for it. So there are nodes that are potentially more trustworthy than others, depending on the heterogenity of the peer group that had issued certificates for them. As this approach imposes the problem of non-binary trust, it is scalable and applicable for heterogeneous IoT networks.

Let (V, E) with $E \subset V \times V$ be the certificate graph of the IoT network of the nodes V. The certificate graph stores the certification relations between nodes, i.e. $(u, v) \in E \Leftrightarrow \mathrm{Cert}(u, v)$ exists in the network. For simplicity we further assume a mutual certification process, meaning any node v that trusts u and issues a certificate $\mathrm{Cert}(u, v)$ also receives a certificate $\mathrm{Cert}(v, u)$ from u. Hence the graph is undirected.

A source node s can assume trust for a destination node d if there exists a certificate chain

$$s := v_0, v_1, ..., v_{c-1}, v_c =: d \qquad v_1, ..., v_c \in V, (v_i, v_{i+1}) \in E$$

with $\mathrm{Cert}(v_i, v_{i+1})$ for $0 \leq i < c$. To begin with, every node may have only its neighbors included in its own certificate list when it enters a network. The problem of certificate chain discovery [Ki05; Mo07] evolves, as not every node is able to have direct access to every certificate present in the network.

4 Related Work in IoT PKI Solutions

The PGP standard [Ca07] defines a WoT model for the World Wide Web that could potentially be adapted for IoT [Ki05; Mo07]. Decentralized approaches exist, e.g. [DI10] that mitigate the configuration problem. But the issuing of a root certificate is always a problem.

Blockchain-based PKI approaches have the potential to eliminate the CA as the single point of failure [LSM17; PDF18; SB18]. These approaches, however, have huge drawbacks. Each node must store at least a subcopy of the global blockchain, and there must be a mining procedure as credit source between nodes. Both add storage and computational demands that are expensive for IoT nodes and limit scalability and energy efficiency.

5 Certificate Handling in (Dis-)Aggregating IoT Networks

In our work we suggest to use resource directories that are part of the Constrained Application Protocol (CoAP) specification [SKA19] to aid in certificate exchange process.

5.1 Service Discovery and Certificate Chain Discovery

Both the service-oriented architecture and the certification process suffer from respective discovery problems. Service discovery is necessary for aggregating microservices, because the complexity of applications is so high that they can not be managed statically. Similarly, since trust can not be statically configured with every IoT device, networks have to autonomously discover certificate chains at runtime. Both mechanisms can, therefore, utilize the same infrastructure, as we show in the following sections. Our idea is basically to make use of CoAP service directories, which compose a solution to the resource discovery problem, and re-use them to additionally store and discover certificate chains in IoT networks.

We base our solution on the certificate chain discovery algorithm proposed by Kitada et al. [Ki05], which we briefly introduce before presenting our variation. Suppose every IoT node $v \in V$ stores the set of certificates

$$\begin{aligned} C_v = &\{\mathrm{Cert}(v,u) | u \text{ has signed a certificate for } v\} \cup \\ &\{\mathrm{Cert}(u,v) | v \text{ has signed a certificate for } u\} \end{aligned}$$

that it either signed itself for another node u or that were issued for v by u. The certificate graph then contains an edge between nodes u and v if and only if there is a certificate $C \in C_v \cup C_u$. A certificate chain between nodes s and d is found using the Algorithm 1.

Mohri et al. [Mo07] calculated the mean communication overhead $S_1(k)$ for finding such a path as follows.

$$S_1(k) = h \times k \times S_{\mathrm{res}} + h \sum_{i=1}^{k} m^i \times S_{\mathrm{req}}(i-1) \tag{1}$$

Algorithm 1 Find a certificate chain $v_0, v_1, ..., v_c$ between $s = v_0$ and $d = v_c$ [Ki05]

```
$(V, E') \leftarrow \text{SpanningTree}(V, E, s)$
$v_0, v_1, ..., v_c \leftarrow$ Path from $d = v_c$ to $s = v_0$ in $E'$
return $v_0, v_1, ..., v_c$
```

Where h is the average number of hops between two nodes that share a certificate, m is the average node degree in the graph (V, E), k is the height of the spanning tree, and $S_{\text{req}}(i)$, S_{res} are the packet sizes of the request and response packets in bytes. They are given as follows (including header sizes $s_{\text{req}}, s_{\text{res}}$) [Mo07]:

$$S_{\text{req}}(i) = \text{sizeof}(\text{Cert}(u, v)) \times i + s_{\text{req}},$$
$$S_{\text{res}} = \text{sizeof}(\text{Cert}(u, v)) + s_{\text{res}}.$$

Note that k is at the same time the average path length of a certificate chain. By increasing the number of edges in the graph (thus increasing m), the path length k is reduced. Because k is in the exponent, increasing m to reduce k is a good choice, but is often not possible, because the number of certificates that can be stored on a resource-constrained IoT node is limited. Kitada et al. suggest $m = 4$ as a minimum value to ensure a closed graph is formed, but that might result in long paths and high overhead.

5.2 Variation by Introducing Structure with Resource Directories

As a solution to the exponential overhead we suggest the introduction of resource directories with bigger storage. CoAP specifies such resource directories to address the resource discovery problem [Sh20]. As the standard does not restrict their use, they can serve to support the certificate chain discovery problem as well. In IoT networks, these resource directories may consist of small server nodes, with memory capacities in the Gigabyte range, thus being able to store a significant amount of certificates. Examples of these could be edge nodes, like network routers or bridges, that are common entities in IoT networks.

Assuming that a single directory can store up to l certificates, we propose a network structure where the n nodes are divided into $\lceil n/l \rceil$ groups, each group being assigned to one resource directory. The certificate graph can then be ordered as indicated in Fig. 2.

The $\lceil n/l \rceil$ directories have to form a trust network among one another, which can be a fully connected graph, or itself a meshed network similar to that proposed in [Mo07] or [Ki05]. The former method is preferable, since it reduces the path length k to 3 at maximum. We however show by numerical analysis that also a meshed connection between the resource directories reduces the overhead to a practically feasible amount.

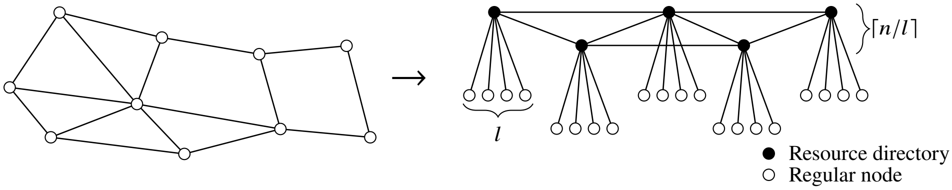

Fig. 2: Left: Unstructured certificate graph after [Ki05; Mo07]. Right: Structured graph through resource directories.

6 Numerical Analysis

We compare the communication overhead of the proposed resource directory based discovery algorithm using Eq. (1). We assume an IoT network in the form of a Wireless Multi-Hop Network (WMHN) with a variable node count n and a proactive routing scheme, that has fully populated routing tables as required by [Ki05]. The respective meshed topologies shall be chosen such that their diameter d is $O\left(\sqrt{n}\right)$. This is, for example, the case for a Manhattan grid. We estimate $d = \sqrt{n}$ for the sake of simplicity, which is for example the case for an 8-neighborhood.

Having $r = \lceil n/l \rceil$ resource directories (as part of the n regular nodes), we assume them to be interconnected with a certificate graph of diameter $d_{\text{structured}} = \sqrt{r}$. The height of the spanning tree then can grow up to a maximum of $k_{\text{structured}} = d_{\text{structured}} + 2$ between two regular nodes. We estimate the average node degree $m_{\text{structured}}$ as follows.

$$m_{\text{structured}} = 2 + \frac{r^2 - 3r}{n}.$$

The proof is given in the appendix. We compare the results to those that were assumed by Mohri et al. [Mo07] with a general network structure that does not distinguish between resource directories and normal nodes. For that, we then assume $m_{\text{unstructured}} = 4$, $k_{\text{unstructured}} = \left\lceil \sqrt{n} \right\rceil$. The structured approach thus reduces the value of k approximately by a factor of $\sqrt{l}$ compared to the unstructured approach.

Parameter	Symbol	Value
Request header size	s_{req}	16 Byte
Response header size	s_{res}	16 Byte
Certificate size	sizeof(Cert(u, v))	256 Byte
Mean hop count	h	1

Tab. 1: Common parameters used for analysis.

Moreover, the parameters in Tab. 1 are chosen for both methods.

Fig. 3 shows the results for both network structures, where $l = 100$ is chosen for the structured network. The structured network clearly outperforms the unstructured one. This is because the height of the spanning tree k and thus the communication path length is limited to the relatively small number of resource directories compared to the overall node count. In the structured approach, the flooding can omit the relatively big number of leaf nodes that are known to hold no useful information.

A disadvantage is the increased storage capacity that a resource directory needs to store certificates. They do not only have to store all certificates of their l leaf nodes, but also need additional storage to form a trust network among each other. In case of a fully connected network between the resource directories, the total number of stored certificates is $(r + l) \times \text{sizeof}(\text{Cert}(u, v))$.

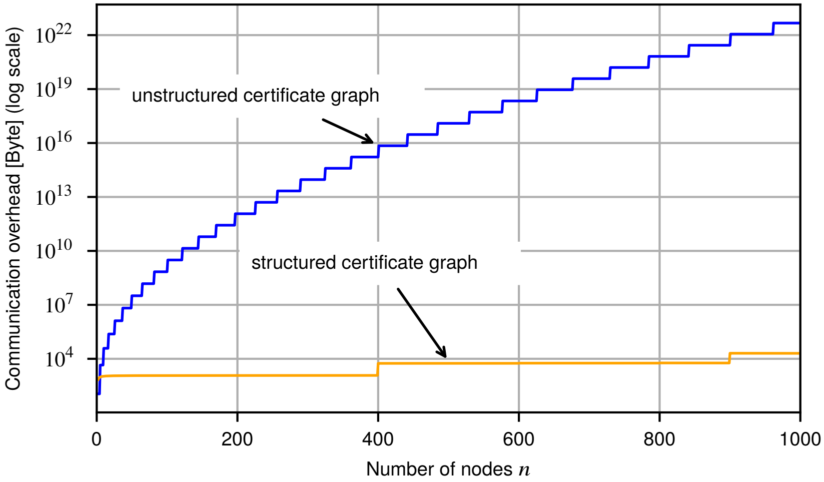

Fig. 3: Overhead analysis.

7 Discussion

Both the certificate chain discovery problem and the service discovery problem are important milestones on the pathway to realize microservice-based (dis-)aggregation of large applications in the IoT. In this paper, we provide an argument for solving both problems in the same manner in a combined, semi-structured approach. The benefits of the combination are the reduction of redundancies as well as strong synergy effects. By using specialized devices with a certain memory capacity, that act as combined service and certificate directories, the communication complexity is shown to be reduced by a significant amount. In our numerical analysis, resource directories, as they are proposed for example in IETF drafts [SKA19], show to reduce communication overhead. Naive approaches, on the other hand, can easily overburden the capacities of devices and networks even in small-sized problems.

Although the semi-structured approach is accepted and already widely adopted for the resource discovery problem, it is not so popular for the certificate chain discovery problem, where the idea of having a completely distributed WoT, in absence of any hierarchies, dominates. The semi-structured approach is here an in-between solution that utilizes elements of both WoT and PKI approaches.

Acknowledgments. This work is funded by the German Federal Ministry of Education and Research (BMBF) as part of the project "Doriot" (Dynamic runtime for organically (dis-)aggregating IoT-processes) under grant number 01IS18071A. See also the project website `www.doriot.net`.

References

[AAE16] Alshuqayran, N.; Ali, N.; Evans, R.: A Systematic Mapping Study in Microservice Architecture. In: 2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA). Pp. 44–51, Nov. 2016.

[Ca07] Callas, J.; Donnerhacke, L.; Finney, H.; Shaw, D.; Thayer, R.: OpenPGP Message Format. Internet Engineering Task Force (IETF)/, RFC 4880, Nov. 2007.

[DI10] Dahshan, H.; Irvine, J.: An Elliptic Curve Distributed Key Management for Mobile Ad Hoc Networks. In: 2010 IEEE 71st Vehicular Technology Conference. Pp. 1–5, May 2010.

[Ki05] Kitada, Y.; Takemori, K.; Watanabe, A.; Sasase, I.: On Demand Distributed Public Key Management without Considering Routing Tables for Wireless Ad Hoc Networks. In: 6th Asia-Pacific Symposium on Information and Telecommunication Technologies. Pp. 375–380, Nov. 2005.

[LSM17] Lin, J.; Shen, Z.; Miao, C.: Using Blockchain Technology to Build Trust in Sharing LoRaWAN IoT. In: Proceedings of the 2nd International Conference on Crowd Science and Engineering. ICCSE'17, Association for Computing Machinery, Beijing, China, pp. 38–43, July 2017.

[Mo07] Mohri, H.; Yasuda, I.; Takata, Y.; Seki, H.: Certificate Chain Discovery in Web of Trust for Ad Hoc Networks. In: 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07). Vol. 2, pp. 479–485, May 2007.

[Na16] Nadareishvili, I.; Mitra, R.; McLarty, M.; Amundsen, M.: Microservice architecture: aligning principles, practices, and culture. O'Reilly Media, Inc., 2016.

[PDF18] Pinto, G. V.; Dias, J. P.; Ferreira, H. S.: Blockchain-based PKI for crowdsourced IoT sensor information. In: International Conference on Soft Computing and Pattern Recognition. Springer, pp. 248–257, 2018.

[Ro16] Roca, D.; Nemirovsky, D.; Nemirovsky, M.; Milito, R.; Valero, M.: Emergent Behaviors in the Internet of Things: The Ultimate Ultra-Large-Scale System. IEEE Micro 36/6, pp. 36–44, Nov. 2016.

[SB18] Singla, A.; Bertino, E.: Blockchain-Based PKI Solutions for IoT. In: 2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC). Pp. 9–15, Oct. 2018.

[Sh20] Shelby, Z.; Koster, M.; Bormann, C.; van der Stok, P.; Amsüss, C.: CoRE Resource Directory draft-ietf-core-resource-directory-24. IETF Draft/, Mar. 2020.

[SKA19] van der Stok, P.; Koster, M.; Ansüss, C.: CoRE Resource Directory: DNS-SD mapping draft-ietf-core-rd-dns-sd-05. IETF Draft/, July 2019.

[YZV14] Yan, Z.; Zhang, P.; Vasilakos, A. V.: A survey on trust management for Internet of Things. Journal of Network and Computer Applications 42/, pp. 120–134, 2014.

Appendix

Theorem 1. $m_{\text{structured}} = 2 + \frac{r^2-3r}{n}$ *is an upper bound for the average node degree in the structured network.*

Proof. The $n - r$ regular nodes each have a degree of 1. For the r directory nodes, we assume that the regular nodes are balanced among them. So edges exists from each directory node to (on average) $(n - r)/r \leq l$ regular (leaf) nodes. Furthermore, assuming a fully connected network between directory nodes as worst-case assumption, an edge to each other of the directory nodes exist, which yields an average degree of $r - 1 + (n - r)/r$ for each directory node. That yields

$$m_{\text{structured}} = \frac{(n-r) \times 1 + r \times (r - 1 + (n-r)/r)}{n} = 2 + \frac{r^2 - 3r}{n}.$$

□