Challenges from the Identities of Things

Introduction of the Identities of Things Discussion Group within Kantara Initiative

Ingo Friese
Telekom Innovation Laboratories
Deutsche Telekom AG
Berlin, Germany
Ingo.Friese@telekom.de

Jörg Heuer
Telekom Innovation Laboratories
Deutsche Telekom AG
Berlin, Germany
Joerg.Heuer@telekom.de

Ning Kong
China Internet Network Information Center
Beijing, China
nkong@cnnic.cn

Abstract—**The Internet of Things (IoT) becomes reality. But its restrictions become obvious as we try to connect solutions of different vendors and communities. Apart from communication protocols appropriate identity management mechanisms are crucial for a growing IoT. The recently founded Identities of Things Discussion Group within Kantara Initiative will work on open issues and solutions to manage "Identities of Things" as an enabler for a fast-growing eco-system.**

Keywords—identity; identifier; name service; authentication; authorization; privacy; Kantara Initiative

I. INTRODUCTION

The "Internet of Things" (IoT) is beginning to evolve and early solutions are now being implemented. We can find implementations in areas like logistics, farming, industry, home automation and many others. But its restrictions become obvious as we try to connect solutions of different vendors, communities or standard groups.

From a business point of view the IoT enables a plethora of new opportunities, use cases and scenarios. From a technical point of view the IoT consists of uncountable devices, sensors or actuators or simply objects connected to services in the Internet. Today, devices and sensors speak a lot of different protocols, but most of them are not HTTP. That is why application development in the IoT is hard to be implemented. There is a lack of decent application integration layers. The next logical step is to use common Web technologies for the IoT. Identity management is one of the most important common technologies.

II. IOT IDENTITY FRAMEWORK

Apart from adapting communication protocols an overarching identity framework is crucial for a growing IoT. Today we have many separated solutions and niche standards. As a consequence, there is no overall framework for how to recognize and manage identities across different solutions. That is why we decided to found a new discussion group called "IDentities of Things" (IDoT) within Kantara Initiative [1].

This group will discuss open issues and possible solutions to manage IDoT as an important enabler for a fast-growing eco-system in support of IoT broader landscape. In this paper we are going to introduce the open issues the new group wants to address encompassing identifier, mapping, authentication, authorization and privacy. Today all these objectives can be solved somehow but none of them is solved in an overarching manner enabling fast-growing and Internet-wide use cases.

Ownership and identity relationships

Things or objects in the IoT often have a relationship to real persons. These could be owner(s), manufacturer(s), user(s), administrator(s) or many other functions. These relationships are affected by lifecycles. A product might be owned by a manufacturer first and subsequently by a user who bought the product. The owner, user or administrator of an object might change over time. Objects finally disappear from the IoT after a certain lifetime. Identity lifecycles in the IoT can be much longer or shorter than in classic user-related identity management systems.

Ownership and identity relationships in the IoT have an impact on other identity related processes like e.g. authentication, authorization or governance of data. The owner of a thing might be challenged for authentication or be asked for authorization policies. The user of an IoT object might have control about certain data collected by that object. These impacts will be explained in more detail in the next paragraphs.

Object Identifier and Namespace

A namespace is a collection of identifiers [2]. The today's Internet has a worldwide valid namespace. Computers connected with the Internet have an IP-address as an identifier. As long as it is a public one it can be addressed from every other computer. On top of IP-addresses there is a domain name service (DNS) mapping "human readable" identifiers to IP-addresses. The assignment of an IP-address to a domain name is a managed process. The organization "Internet Assigned Numbers Authority" (IANA) [3] manages the assignment of IP-addresses and domain names in the Internet.

The IoT would also benefit from one kind of identifiers. But we do not start on a green field. There are many existing solutions outside already. As a consequence an identity framework has to integrate many different object identifiers ranging from IPv6 addresses, RFID chips and QR Codes up to business domain specific IDs like license plates on a car or

SIM cards in mobile devices etc. The today's DNS is not sufficient here due to the fact that it mainly maps IP-addresses to domain names.

In order to work with different kinds of identifiers a flexible mapping or object- /thing name service is necessary that would be able to map between different object identifiers and their regarding namespaces as well as to express various object identifiers in a defined syntax. Although many objects would benefit of an "object name service" it won't work for all.

There won't be a worldwide unique namespace in the IoT and not all objects might be addressed from everywhere in the near future.

The following example explains the challenge: In the farming industry a harvester and a truck want to communicate with each other. The harvester machine wants to send a message to the truck. The truck should pass by in order to unload the crop from the harvester. The harvester and the truck are manufactured and owned by different companies so they have no common infrastructure or proprietary solution to communicate with each other. For simplicity we assume both have IP-connectivity via mobile LTE/3G network. When the harvester- and the truck driver meet in the morning what kind of identifier they do exchange in order to connect their machines? The IP-address might be changed due to the fact that mobile providers use IP-address pools. With every mobile login the mobile client might be assigned to another IP-address. So IP-addresses are not a good choice for this specific use case.

What kind of identifier and what namespace to use?

In general there are two options: The first option is to use an Internet-wide valid identifier for both the harvester and the truck. An "object name service" could map an identifier like "JoesHarvester@jd.machines" and "TomsTruck@californiaTrucks" (the syntax is just an example) to the current IP-addresses of harvester and truck. By exchanging their identifier once they can build up a connection anytime.

The second option is to form a temporary identifier and namespace across all machines belonging to the specific use case. Both harvester and truck agree on registering at a certain server e.g."california_farming_service". Only this server is able to map the nick names like "Tom@california_farming_service" to current IP-addresses. This service could be managed by a local company or even by the truck or harvester owner himself.

Both of these options are opposite approaches. While the first option describes theoretically one huge Internet-wide namespace and name service the second option is about the smallest namespace. Just two identifiers are valid in our example.

Although these solutions are different they do not exclude each other. A favorable solution could cascade smaller namespaces towards bigger ones. In our example the smaller name space service "California_farming_service" could be a part of a bigger namespace and thus it could also be a part of an Internet-wide spanning namespace.

This calls for mechanisms and protocols that are able to handle different object identifier syntax, mapping mechanisms and resolution protocols. One exemplary protocol is OASIS Extensible Resource Identifier (XRI) [4]. XRI describes a scheme and a resolution protocol for so called abstract identifier. Abstract identifiers are independent of domain, application or communication protocols [5]. That is why they can be shared across different domains and namespaces.

Example:
xri://technical-university-berlin.library.com/(urn:isbn:3-823-140-0)

This identifier names a book with a certain ISBN-number in the library of the technical university of Berlin. The XRI cross-reference syntax allows combining the namespace ISBN and HTTP-URLs. In the same way XRI could be used to combine other kind of identifiers and namespaces as well.

XRI is currently just one example used to illustrate possible solutions. Whether this protocol is a possible solution or not is up for further discussion.

Authentication

The classic authentication mechanisms (ex.: login / password) may not directly work in the IoT. Objects have to provide some sort of lightweight token or certificate. For the stronger authentication of individuals we usually combine multiple factors. These factors are based on following proofs "Something that you have", "Something that you know" and in case of biometry also "Something that you are". In the IoT the last two proofs are not applicable to objects anymore. On the other hand many objects are going to have an owner, manufacturer or user that has a relationship to his object which can be used for authentication.

How to strengthen authentication means in the IoT?

Context-based authentication

A valuable resource for additional authentication proofs is the context and environment of the authentication request. Additional information for example could be taken from the network layer, from geographical information or from other use case specific factors.

The basics of this approach are not new. In classic access management it is also known as risk based authentication [6]. The main idea is to check an incoming request first for different characteristics.

The following example explains risk-based authentication in a traditional user authentication use case: A sensitive service in a company is complementary protected with a risk-based authentication. When a user requests access, the IP-address of the request is checked. If the IP-address belongs to an internal network the user has just to provide username and password. If the IP-address is unknown the user has to answer additional security questions or present a certificate on her/his smartcard.

The basic principle could also be applied to the IoT. If a request of a thing comes from a known network, an access token might be sufficient. Otherwise other factors like e.g. LTE cell ID or geo location of the requester are also checked.

In the classic approach an authentication request could be facilitated theoretically from everywhere in the whole Internet. That is why only Internet-wide available information can be used to calculate the risk (e.g. IP-address, client type, geographical region).

In the IoT usually much more use case specific factors can be taken into account (e.g. internal machine IDs, specific geo-location data like LTE cell ID in use case where all machines have mobile connectivity).

Although this approach might lead to harder authentication, it should be seen just as an complimentary method in addition to e.g security tokens due to the fact that all additional information might be available to the public.

Another source for additional authentication proofs is ownership or identity relationship of an object. Although in the IoT objects and things are communication endpoints most of them are managed or somehow related to real persons. An IoT object might have an owner, administrator, user or even groups of them. This relation can be used to authenticate an object.

If for example an IoT device wants to access very sensitive information from another IoT device the owner of the requesting device might be contacted and asked for authentication.

Authorization

More sophisticated devices need authorization and different access rights. An administrator might have the entitlement to set certain values while a normal user can just read data. This is not a new challenge and there are different approaches especially for enterprise access management. The new situation coming with the IoT is that access management becomes important for a broad range of sensors and actuators. Traditional approaches like Role Based Access Management (RBAC) [7] were developed to manage user access in organizations or enterprises.
The OAuth 2.0 authorization framework [8] addresses the authorization problem in the Web. It was adopted by major websites like Google and Facebook. The OAuth protocol flow describes how a user authorizes an application to get access to a resource of the same user hosted at another application. A typical example is the following situation: A user wants to authorize a printing service to access his pictures on a photo website like e.g. flickr.com. The printing service has to obtain an access token that grants access to the requested photos. The request is send to the authorization server of the photo website.

Most of these requests require an authentication step where the user logs in with her/his photo website account. After logging in, the user has to agree on granting the permissions to access users pictures. When the permission was granted the printing service gets an access token or an authorization code that can be used to obtain an access token. With a subsequent request the printing service can access the photos of the user by presenting the access token.

The OAuth flow usually requires the presence of the user in order to authenticate and to grant access to the requested resource. This might be a problem in a typical IoT communication e.g. between two devices. The User Managed Access (UMA) profile [9] built on top of OAuth might offer authorization based on policies even when the user or owner is offline.

In the UMA flow an application requests a so called protected resource. The requesting application has to present a requesting party token (RTP). This can be obtained from an authorization server. The requesting party has to provide to the authorization server any identity claims needed in order to associate sufficient authorization data with that requesting party. These identity claims might be e.g. certificates. An online presence of a user is not necessary.

The authorizations server could be equipped with a sophisticated policy engine where the owner or administrator of a resource can define what other applications may have access to certain resources and what kind of identity claims they have to provide. UMA seems to be an appropriate authorization framework for many IoT use cases.

But UMA is just an example and could be one option among others. Authorization frameworks for the IoT still need further studies and discussions.

Governance of data and Privacy

The IoT will lead to an increasing amount of data sources producing a tremendous amount of data related to people. A single dataset of a sensor might not affect the privacy of a person. But data could theoretically be combined or related to valuable – but also critical – pieces of information about a person, a person's behavior, the personal situation or any other private information. For example a truck of a logistics company might be equipped with a GPS tracker, velocity sensors and other on-board diagnostic devices. The GPS data allows to track the driver's whereabouts. If this data becomes publicly available it can be combined with the home address of the driver, delivering valuable information for burglars and other criminals. So the truck driver is interested in hiding data of the GPS tracker. On the other hand the data might be important for statistics and optimization of timetables of the logistics company which they may only use in accordance with strict laws. Also communities and local authorities might have an interest in real time traffic information etc. so they might want to use this data too. In fact driver, the corporate owner, and e.g. communities have different claims regarding the same data.

The 'claim-to' approach

The example shows that different actors in a scenario might have different claims to the data produced by a particular source. A simplified dataflow in this approach starts with the collection of data in a sensor - the data source. The data is transmitted to a data sink - a device or an application where the data is finally consumed and evaluated. The data sink is tightly related to one of the actors in a scenario who has a 'claim-to' the data and some quality of that data. One or more intermediate devices might relay the data (e.g. router, gateways, etc.). Data source and data sink are in the hands of legal entities that have responsibility for the data created, transmitted and consumed. These entities need to be provided with all necessary means to control the data according to requirements set by other technical systems employed, business rules, laws or even cultural standards.

In our example we have a data flow from a sensor (GPS tracker) through the Internet to an application or data base in terms of a data sink (the servers or applications of the logistic company, community or a smart device of the driver). The data flow passes several routers and other equipment of the transport network as intermediates.

Now we apply certain operations to the data on their flow depending on the claim of a certain person in a scenario. What does this mean for our example? The truck driver claims secrecy of the GPS data. He just wants to allow himself to see the data. So the flow from the GPS tracker to the smartphone of the driver should be end-to-end encrypted. The logistics company claims visibility of the data. In this case there is no conflict. The solution here is to provide anonymous data to the company. Certain data could be e.g. aggregated over a group of drivers. Position data and average velocity are still valuable for timetable optimization of the company.

The 'claim-to' approach requires few basic operations to be conducted on the data which are appropriate to satisfy the claims of different actors in a use case. Secrecy could be achieved by end-to-end encryption between the sensor and the data sink. So no intermediary has the ability to read the data. Anonymity could be provided by masking out a subset of data. Data could be avoided by discarding them directly at the source. Non-repudiation could be ensured by using digital signatures.

Now the different claims of different actors in a use case can be analyzed and an appropriate data operation could be chosen to satisfy the claim. The operations can be combined – and be partially overlapping. There are few limits to the complexity setups of actors and intermediaries can reach; our goal is to identify a small set of operations usable across any constellation of source, intermediary and sink. There should be no general obligation to implement every aspect in every component of a system – in effect, only very few components will really need to be equipped with the capability to perform a specific operation – and rarely will they need to support all of them. We aim to give guidance on how to provide a certain quality of the data transfer if required – and how to combine components in a standardized way to grant maximum flexibility and always appropriate privacy for the overall system.

But how to proceed with conflicting claims? The goal of this approach is to support architects to design configurable privacy enabled IoT infrastructures. The configuration depends on use cases, scenarios, rules, national laws and cultural standards. In some countries the most restrictive claim might be enforced. In our example it could be prohibited to track the position of an employee in general. In other countries it could be legal and normal to use a GPS tracker in that way and there might also be no obligation to anonymize the data. A system could be configured either way as long as it is able to apply the necessary operations to the data produced by the system.

The advantage of this approach is that we propose to consider the implementation of basic operations for every data flow in a system and every claim of an actor in a scenario. The configuration what to use under what circumstances should be done by the administrators or users in their specific domain, component manufacturer, system vendor, regional sales/ service office, operator of the system or the user - every one of them according to their specific role, requirements and restrictions. So the final decision what data are private or not are not up to the technical architects although they enable the system to support all possible configurations.

III. Identities of Things Discussion Group and its Future Work

The Kantara Initiative - Identities of Things Discussion Group was founded in June 2013. A discussion group in Kantara Initiative is a lightweight ancestor for a working group. Usually it is used to prepare the foundation of a working group that provides technical documents and recommendations.

In a first step the group wants to identify and analyze open issues and gaps. It intends to describe the object identity problem in the IoT with exemplary use cases taken from various areas and environments. The discussion group intends to deliver a state-of-the-art analysis of the current object identity management approaches describing issues and challenges. This analysis will be the base for overarching identity framework for the IoT.

This paper introduces the work of the Kantara Identities of Things Discussion Group. We discuss open issues on Identity Management and explain first potential and exemplary solutions for being able to be a part in an overarching identity framework.

References

[1] Kantara Initiative Identities of Things Discussion Group http://kantarainitiative.org/groups/idot/

[2] URN Namespace Definition Mechanisms RFC 2611 http://www.ietf.org/rfc/rfc2611.txt

[3] Internet Assigned Numbers Authority (IANA) http://www.iana.org/

[4] OASIS Extensible Resource Identifier (XRI) https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xri

[5] Wikipedia Extensible resource identifier http://en.wikipedia.org/wiki/XRI

[6] Williamson, G. "Enhanced Authentication In Online Banking" Journal of Economic Crime Management 4.2 (2006): 18–19. Print

[7] NIST Computer Security Devision - role engineering and rbac standards http://csrc.nist.gov/groups/SNS/rbac/standards.html

[8] The OAuth 2.0 Authorization Framework RFC 6749 http://tools.ietf.org/html/rfc6749#page-16

[9] User-Managed Access (UMA) Profile of OAuth 2.0 draft http://tools.ietf.org/html/draft-hardjono-oauth-umacore-07