Distributed access control and privacy for the internet of me

Full-Text version

Title: **Distributed Access Control and Privacy for the Internet of Me**

Authors: Daniel Díaz-Sánchez, *Senior Member, IEEE*
Telematic Engineering Department, Carlos III University, 28911, Leganés, Madrid, SPAIN
(e-mail: dds@it.uc3m.es)

R. Simon Sherratt, *Fellow, IEEE*
School of Systems Engineering, the University of Reading, RG6 6AY, UK
(e-mail: sherratt@ieee.org)

Florina Almenares, *Member, IEEE*
Telematic Engineering Department, Carlos III University, 28911, Leganés, Madrid, SPAIN
(e-mail: florina@it.uc3m.es)

Patricia Arias, *Member, IEEE*
Telematic Engineering Department, Carlos III University, 28911, Leganés, Madrid, SPAIN
(e-mail: ariasp@it.uc3m.es)

Andrés Marín López, *Member, IEEE*
Telematic Engineering Department, Carlos III University, 28911, Leganés, Madrid, SPAIN
(e-mail: amarin@it.uc3m.es)

Publication: **2016 IEEE International Conference on Consumer Electronics (ICCE)**
pp.: 17-18
Date: 7-11 Jan. 2016
Location: Las Vegas, Nevada, USA
DOI: 10.1109/ICCE.2016.7430506

Funding

This work has been partially funded by project INRISCO TEC2014-54335-C4-2-R and Jose Castillejo Mobility Grant CAS144/00364

Abstract

This article presents an experimental scalable message driven IoT and its security architecture based on Decentralized Information Flow Control. The system uses a gateway that exports SoA (REST) interfaces to the internet simplifying external applications whereas uses DIFC and asynchronous messaging within the home environment.

I. INTRODUCTION

IoT will become a revolution in the future of home environment. Information about preferences and habits are essential to increase comfort so IoT requires such information to flow adequately. Devices or things, retrieve information, feed other information systems and trigger processes that may have effect in the real world as it happens with automation.

To cope with this huge amount of information the cloud is used for offloading. Despite this opens the possibility to create a number of other new services, it raises concerns about privacy given the sensitive nature of the information that is delivered to the cloud. It is also worrying critical devices may be hijacked and misused from anyone all over the internet.

This article presents an experimental scalable message driven IoT and its security architecture based on Decentralized Information Flow Control (DIFC) [1]. It allows several entities to contribute to a distributed security policy that governs not only the access to the information or services provided by things but also controls the information delivered by things to the cloud preventing information leakage. However scenarios in which external applications need to communicate directly with things inside a domain would require the application to have knowledge about the target home domain increasing its complexity. Due to that, we propose the use of a gateway that exports SoA (REST) interfaces to the internet simplifying external applications whereas uses DIFC and asynchronous messaging to interact with home devices. The external entities are authenticated and their calls to the APIs authorized using digital identity and Oauth.

II. Distributed Information Flow Control

We use a simple DIFC as described in [2]. The system is considered closed and any communication with the outside world is handled through managed channels. Each principal (user or device) can specify policies controlling how the information is propagated within the system. The information is labeled and policies are expressed of the form {O:R}. This means the owner O allows this information to be read (disclosed) only by readers R. O is a principal and R is a set of principal. If several principals have interest on the data, a policy would look like this L {o1:r1,r2,r4; 02:r2,r3} where on, rk are principals. The effective reader set is the intersection of the individual readers set, so {r2}. To control how the information is accessed and declassified, it is necessary to ensure the policies of each owner are enforced as the data is read and written. When data is read (accessed) the principal requesting the data should be included in the effective reader set to be able to read it. When written within the system, the data is relabeled. This happens only if the new label allows fewer accesses than the original, for instance, the new label has fewer readers. A relabeling, so letting the data flow to L1 to L2 is defined (ow means owners, rd readers):

$L_1 \subseteq L_2$ if
$ow(L1) \subseteq ow(L2) \forall O \in ow(L1),$ (1)
$rd\ (L1, O) \supseteq rd\ (L2, O)$

It is also possible to combine data from different labels. The resulting outcome is labelled with a derived label. If two values labeled L1 and L2 are combined, the resulting label should have a least restrictive label that maintains all the flow restrictions specified by L1 and L2. The owner set of the resulting label is the union of the owner set of L1 and L2 and the reader set is the intersection of their reader sets:

$L_1 \cup L_2$ if
$ow(L_1 \cup L_2) = ow(L1) \cup ow(L2),$ (2)
$rd(L_1 \cup L_2) = rd(L1, O) \cap rd(L2, O)$

Finally, data can flow in and out the system by using channels. Channel creation is controlled and every channel has a label. If the data is read from a channel, the data copy the channel label. In such a way, the potential readers of that data can be controlled. Moreover, if the data is written to a channel, the label of the channel should be that the aforementioned writing rules are kept so owner's policies are respected.

III. EXPERIMENTAL HOME IOT ENVIRONMENT

A foreseeable future IoT home domain can be considered a highly distributed system with a high rate of device replacement and reconfiguration, dynamic offloading and re-orchestration. Due to that, we believe a central policy repository would require constant updates and would constitute a single point of failure. Such a system should be designed as resilient, responsive, elastic and message driven.

Under those premises a device in our experimental system contains an execution platform that supports multi-tenancy, allowing several small programs to run simultaneously inside the device. In such a way, beyond the device's main application, a device in our IoT system may have other applications owner by other principals. For instance one owned by the manufacturer that monitors its usage and one owned by a user that manages his preferences. Every of them can send and receive messages. From the security perspective, multi-tenancy and message driven communication are key for the adoption of the DIFC since every principal can enforce its policy to control the flow of the information they own.

A. Architecture and Operation

The communication among different devices is governed by DIFC and the communication among devices and the Internet is performed through managed channels (Fig. 1).

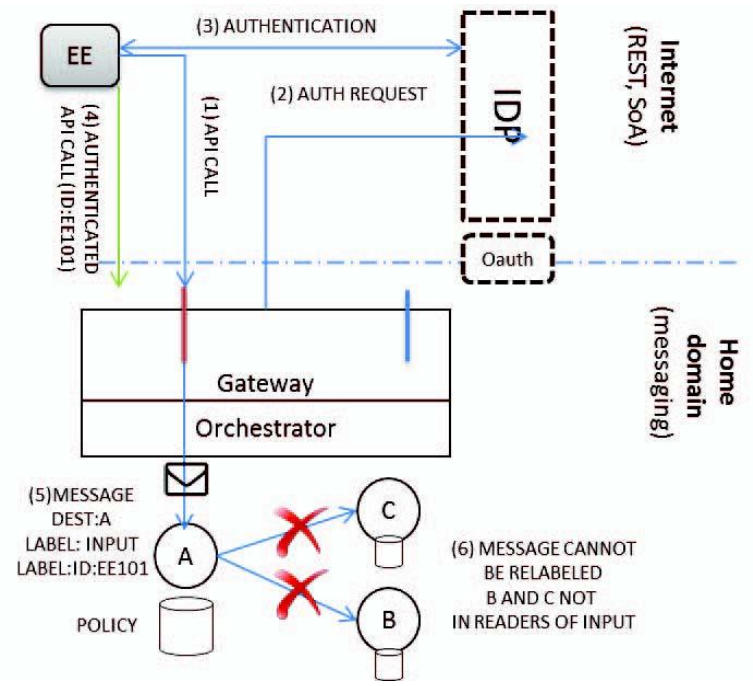

Fig. 1. Experimental IoT home environment

The gateway implements the input and output channels that provide communication with the internet. The gateway exposes well known interfaces, as HVAC, to the internet. Calls to the API are eventually translated by the gateway or the orchestrator in a combination of asynchronous messages and delivered to the appropriate devices inside the home environment. In this way, external entities (EE) can interact with the system without requiring any knowledge about the home network. The identity provider (IdP) allows external entities to authenticate and authorize to the system and eventually to obtain access to an interface by means of OAuth.

B. Use cases and operation

Due to space limitations this section describes how the system controls interaction with external entities that illustrate also the internal operation. An example is depicted in Fig.2. An EE requests access to the HVAC interface to increase the temperature of the home that is a privileged operation. The gateway controls N input channels from which devices inside the home domain can read external messages. Input channels are privileged and their creation is controlled by the gateway. Only authenticated and authorized EE can write information in these channels. That prevents denial of service attacks, but if there is no policy restraining an authenticated EE from writing into a given channel, the policy enforcement will be handled in a distributed fashion at device level using DIFC. When messages are read from these channels, the data is labeled with the channel label. The message generated by the gateway and injected into the home network is labeled also with an attribute assertion obtained from the IdP that can be either an identity expression, a role or a group membership assertion.

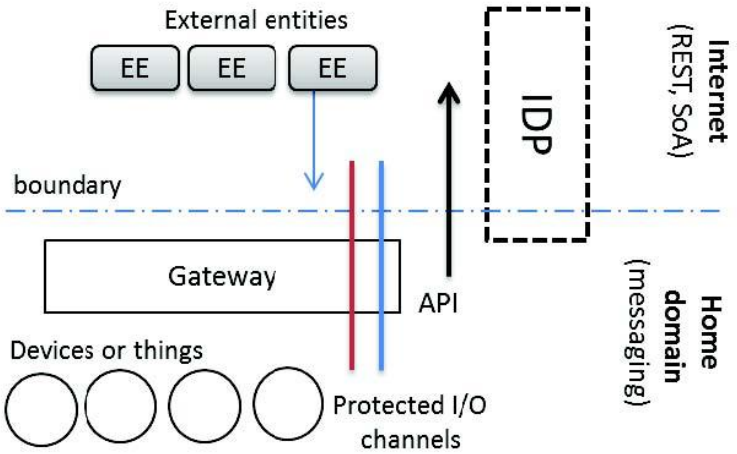

Fig. 2. Use case of distributed policy enforcement for information flow

In the example, the call to the interface generates a message that is delivered to device A (a thermostat). Device A is in the group of readers of the input channel since should accept queries for the current temperature value only. Nevertheless, due to a misconfiguration of its policy, it accepts the operation. As a consequence, the device A should switch the boiler on and activate a pump by sending messages to devices B and C. To send a message to device B, the device A needs to relabel the original request. Relabeling is only authorized if the resulting label respects the original restrictions, however in this case, devices B and C (critical) are not in the set of readers for the input channel so the original query cannot be relabeled nor delivered to B and C. Information leakage is also controlled. Consider a rogue device (D) that sends usage information to the manufacturer. The user can assert a policy to control the information removing the output channels from the device D readers.

IV. IMPLEMENTATION AND CONCLUSIONS

To implement this IoT system and its distributed policy system we relied on the actor model [3]. The IdP implements OpenId. Our playground includes several raspberry Pi acting as things. The gateway and the IdP are co-located in our test bed and running in a PC.

DIFC has been already proposed as a candidate to control information leakage in IoT but the addition of a gateway and the usage of REST APIs, digital identity and OAuth does not break current internet trends whereas prevents information leakage, controls external access to the home environment and respect the dynamic character of IoT.

REFERENCES

[1] N. Zeldovich, S. Boyd-Wickizer, and D. Mazières, "Securing distributed systems with information flow control," in Proceedings of the 5th NSDI'08, pp 293-308, 2008

[2] A. Myers and B. Liskov, "A Decentralized Model for Information Flow Control," in Proceedings of SOSP '97, pp 129-142, 1997.

[3] Diaz Sanchez, Daniel; Simon Sherratt, R.; Arias, Patricia; Almenarez, Florina; Marin, Andres, "Enabling actor model for crowd sensing and IoT," Consumer Electronics (ISCE), 2015 IEEE International Symposium on, 2015