Asset-Oriented Access Control: Towards a New IoT Framework

Thomas Cattermole
University College London
tcatterm@cs.ucl.ac.uk

Simon Docherty
University College London
simon.docherty@ucl.ac.uk

David Pym
University College London
david.pym@ucl.ac.uk

M. Angela Sasse
Ruhr-Universität Bochum
Martina.Sasse@ruhr-uni-bochum.de

ABSTRACT

Controlling asset-access has traditionally been considered a matter for systems in which assets reside. Centralized approaches to access control are, however, problematic for the IoT. One reason for this is that devices may not be confined to a single system of control. In this abstract, we argue for a new paradigm in which assets are empowered to make their own access decisions. To facilitate this shift in perspective, we propose a policy-neutral framework based on principles adapted from object-oriented programming. This approach establishes assets as active, message-passing entities that store and determine their own access control. We describe initial work modelling the interaction of such assets and point to future formal work for reasoning about protocols and policy composition.

CCS CONCEPTS

• **Security and privacy** → *Access control*;

KEYWORDS

Internet of Things; Access Control; Object-Oriented.

IoT'19, October 2019, Bilbao, Spain

ACM Reference Format:
Thomas Cattermole, Simon Docherty, David Pym, and M. Angela Sasse. 2019. Asset-Oriented Access Control: Towards a New IoT Framework. In *Proceedings of The 9th International Conference on the Internet of Things (IoT'19).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

INTRODUCTION

Access control (AC) is about managing the operations that principals can perform on assets. As a consequence of the field of AC emerging from attempts to secure essentially passive data assets — in particular, files — it has become an established paradigm that access decisions are made on behalf of assets centrally at the level of the system in which they reside. For example, operating systems typically have a reference monitor that controls the operations that are permitted by users on the files on the system [7]. All well-known policy-neutral frameworks, including Role-Based AC [10], Attribute-Based AC [6], and Usage Control [11] commit implicitly to this system-centred assumption.

As securing IoT devices emerges as a topic for AC, it is becoming clear that traditional centralized approaches are inappropriate [8]. There are two broad reasons for this. Firstly, they are ill-suited to meet the new requirements called for by the IoT. Four are commonly cited in the literature [4]:

- Uncontrolled environments: devices cannot be assumed to be in trusted, secure environments;
- Heterogeneity: devices can perform distinct processes and communicate through distinct protocols;
- Scalability: the security of an indefinite number of devices needs to be ensured;
- Resource Constraints: devices can be limited in terms of memory and processing power.

Secondly, they are ill-placed to address old access control requirements that are made more challenging in the context of the IoT. These include, but are not limited to:

- Dynamic controls: access decisions needs to be made and changed quickly and often [9];
- Fine-grained controls: device operations include and exceed those present for data assets [3];
- Break-glass mechanisms: in exceptional circumstances, trusted parties should be able to contravene controls through auditable means [12].

Consider the scenario in Sidebar 1 adapted from Calo et al. [1]. This asset is prone to be in uncontrolled environments, and its subsidiary systems require securing even though they perform different functions and may use different protocols. It also needs to be possible for indefinitely many new devices to be introduced, and some devices (e.g., the key fob) will have limited storage capacity and processing power. The wide number of use cases and the need for temporary usage means that there will need to be fine-grained and dynamic controls. Emergency usage of the smart vehicle will require break-glass mechanisms. More generally, we are led towards a crucial question for the IoT:

How can access control be carried out in a decentralized manner that satisfies both old and new requirements associated with securing access to IoT devices?

It is already common for newly manufactured cars to have in-built navigation systems capable of storing, and providing directions to, home addresses and previous locations. Leaving access to these subsidiary systems open is a single point of failure security risk: an intruder can break into the car, gather the additional sensitive information and use it to commit further crime. As cars get smarter, the consequences of this open access vulnerability get more severe; keyless ignition systems, remote control garage fobs, and other IoT devices in the car increase the amount of damage criminals can do.

Given that a car is, by its nature, independent of other systems (it may even, from time to time, lose GPS signal), it cannot defer its access control decisions to some external centralized authority. Furthermore, these decisions will be contextual, covering a wide range of use cases. Most obviously, the car will need to allow for different sorts of access to passengers as opposed to the owner. It may need to allow for temporary access, for example in the case of hire cars, and it may even need to allow for exceptional circumstances, for example, when emergency services need access in the event of an accident.

Sidebar 1: Smart Car Example

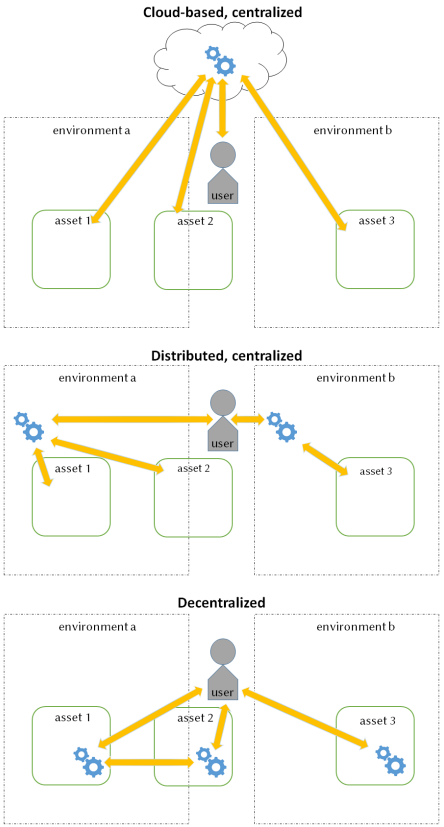

Figure 1: Three models of access control showing how access between assets and a user is controlled within environments. In our approach, decision making is decentralized and occurs within each asset, compared to the centralized approaches where it occurs within the cloud and distributively.

[1]Available at https://github.com/cora711/2D-Simulator-for-Robot-Example.

ASSET-ORIENTED ACCESS CONTROL

To address this research question, we propose a new paradigm for AC in which devices are empowered to make their own access decisions. An overview of this decentralized approach in comparison to two common centralized approaches is shown in Figure 1. We propose a framework for facilitating this shift in perspective based on adapting principles from object-oriented programming, an approach we call Asset-Oriented AC (AOAC). AOAC consists of the following principles:

- Class-Based: assets are class instances made up of fields (properties) and methods (behaviours);
- Inheritance: the fields and methods of a class transitively carry over to subclasses;
- Encapsulation: the fields of an asset are modifiable only through the asset's own methods;
- Polymorphism: an asset's method functions according to the number and nature of its arguments (overloading) and depends on definitions in its own class before its superclass (overriding).

In programming, it is argued these principles make code portable (i.e., usable in many environments). In the IoT, we want devices to be *securely* usable in many environments. Returning to the smart car example, by being class-based, the car stores its AC in its fields and operates through its methods. Access can be controlled through each method allowing for dynamic and fine-grained controls. Through inheritance, the car's security properties and behaviours can carry over to the subsidiary devices present in the car. This removes the single point of failure vulnerability. By exhibiting encapsulation, the IoT devices ensure they only interact with external entities through their own secure methods. Polymorphism (through overriding) allows inheritance to be broken which is useful for low-computing devices like key fobs to defer their access decisions to more capable devices. Overloading allows for break-glass mechanisms in emergency situations by modifying method behaviour according to the principal requesting access. As described, the general message-passing protocol of AOAC allows arbitrary policies to be implemented in a decentralized manner that satisfies IoT requirements.

To test the viability of AOAC, we have developed software using Python[1] for simulating a generic scenario involving autonomous robots interacting securely in an uncontrolled environment. A screenshot of the simulator is shown in Figure 2: a number of worker robots (colored in black) perform a primary task of collecting resources (green), while a supervisor robot (red) performs updates on the workers. These updates occur securely through message-passing between the robots, as shown in Figure 3. Intuitively, the simulator models, say, robots transporting resources, or robot vacuum cleaners in an arbitrary environment. More generally, it provides a proof of concept that assets do not need a centralized reference monitor for asset-access to be controlled.

RELATED WORK, FURTHER WORK AND CONCLUSION

The idea of assets generating their own access control policies through AI has been posed by Calo et al. [1]. This is different from the present work, as we assume policy generation has occured and instead

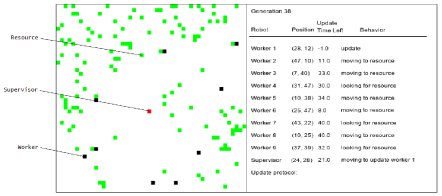

Figure 2: A screenshot of the simulator showing the basic elements including worker robots, resources and supervisor robot, as well as the diagnostic box indicating the current behaviour of each robot.

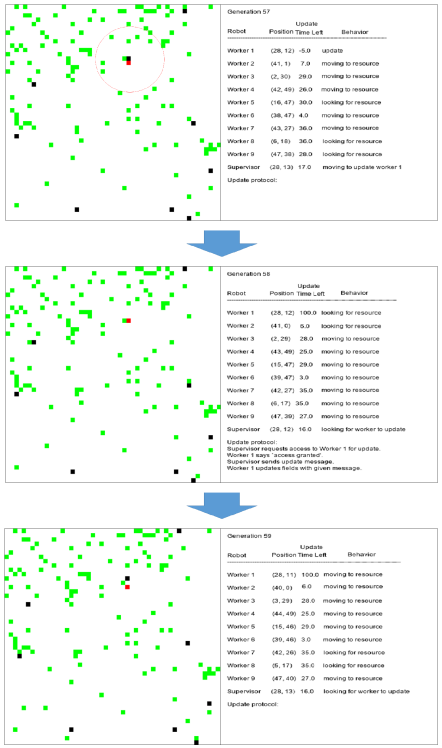

Figure 3: Three successive states of a simulation run showing how Worker 1 autonomously controls access to its updates through the protocol indicated in the diagnostic box.

focus on how it can be implemented in a decentralized, asset-oriented manner. We see this research direction as a necessary precursor to AI-based systems. A general framework should first establish how assets can make access decisions to implement policies before it is shown how assets can generate policies intelligently. In the earlier work of Hernandez-Ramos et al. [5] a low-level capability-based implementation in which "access control is embedded into the end devices" is given, but this approach only deals with severely resource constrained devices, and they provide no general framework.

In future, we plan to use AOAC to implement existing polices applicable to real-world IoT devices. We also hope to formalise the principles of AOAC logically, in order to reason about AC decisions for arbitrarily many devices. Such a logic will need to incorporate ideas involving local knowledge and is likely to build on work in dynamic epistemic logic [2]. Central to this work will be an investigation into the composition of policies initially set for individual assets. There is also plenty of work to be done regarding the more prominent role authentication plays when assets need to directly interact with potentially harmful agents.

The IoT poses new challenges to access control, but also suggests a new approach to empower devices to meet these challenges. We have argued that object-oriented ideas can facilitate this shift in perspective, and we have outlined a research plan to further explore this approach.

REFERENCES

[1] Seraphin Calo, Dinesh Verma, Supriyo Chakraborty, Elisa Bertino, Emil Lupu, Gregory Cirincione. 2018. Self-Generation of Access Control Policies. *SACMAT'18.*
[2] Hans van Ditmarsch, Wiebe van der Hoek, Barteld Kooi. 2008. *Dynamic Epistemic Logic.* Springer.
[3] Earlence Fernandes, Amir Rahmati, Kevin Eykholt, Atul Prakash. 2017. Internet of things security research: A rehash of old ideas or new intellectual challenges? *Security & Privacy '17.*
[4] David Gil, Antonio Ferrández, Higinio Mora-Mora, Jesús Peral. 2016. Internet of things: A review of surveys based on context aware intelligent services. *Sensors '16.*
[5] José Hernández-Ramos, Antonio Jara, Leandro Marín, Antonio Skarmeta. 2013. Distributed capability-based access control for the internet of things. *JISIS'13.*
[6] Vincent Hu, David Ferraiolo, Rick Kuhn, Adam Schnitzer, Kenneth Sandlin, Robert Miller, Karen Scarfone. 2014. *Guide to Attribute Based Access Control (ABAC).* NIST
[7] Ted Hudek, Tim Sherer. 2018. Windows Kernel-Mode Security Reference Monitor. https://docs.microsoft.com/en-us/windows-hardware/drivers/kernel/windows-kernel-mode-security-reference-monitor.
[8] Zaigham Mahmood (Ed.). 2016. *Connectivity frameworks for smart devices: the internet of things from a distributed computing perspective.* Springer.
[9] Htoo Maw, Hannan Xiao, Bruce Christianson. 2012. An adaptive access control model with privileges overriding and behaviour monitoring in wireless sensor networks. *Q2SWinet'12.*
[10] Ravi Sandhu. 1998. Role-based access control. *Advances in Computers '98.*
[11] Ravi Sandhu, Jaehong Park. 2003. Usage control: A vision for next generation access control. *MMS-ACNS'03.*
[12] Yang Yang, Ximeng Liu, Robert Deng. 2017. Lightweight break-glass access control system for healthcare internet-of-things. *IEEE Transactions on Industrial Informatics '17.*