Mammoth: Monitoring the ABAC Monitor of MQTT-based Internet of Things ecosystems

Pietro Colombo
DiSTA, University of Insubria
Varese, Italy
pietro.colombo@uninsubria.it

Elena Ferrari
DiSTA, University of Insubria
Varese, Italy
elena.ferrari@uninsubria.it

Samuele Salvia
DiSTA, University of Insubria
Varese, Italy
ssalvia@studenti.uninsubria.it

ABSTRACT

Data confidentiality and privacy are becoming primary concerns for Internet of Things applications. A variety of access control approaches have been proposed to address this issue. In this demonstration we present a tool, called Mammoth, which complements an ABAC framework for MQTT-based IoT ecosystems, with a dashboard of analysis services designed for security administrators. Mammoth supports the real-time analysis of target MQTT ecosystems, allowing security administrators to analyze the effects of the enforcement mechanisms on the flow of exchanged messages. The demonstration will allow participants to try Mammoth services in a simulated MQTT-based scenario.

CCS CONCEPTS

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

KEYWORDS

ABAC, MQTT, Enforcement Monitor, Analysis services

ACM Reference Format:
Pietro Colombo, Elena Ferrari, and Samuele Salvia. 2020. Mammoth: Monitoring the ABAC Monitor of MQTT-based Internet of Things ecosystems. In *Proceedings of the 25th ACM Symposium on Access Control Models and Technologies (SACMAT '20), June 10–12, 2020, Barcelona, Spain.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3381991.3396226

1 INTRODUCTION

The diffusion and pervasivity of Internet of Things (IoT) applications is continuously growing in many application domains (e.g., healthcare). Numerous IoT devices can even be worn by users, and allow real-time monitoring of users data, such as the health conditions of a patient. Although the benefits of similar services are manifold, they represent a potential threat to user privacy. Different access control approaches have been proposed to mitigate this issue, such as the ones based on the CapBAC model (e.g., [3]), and on customized versions of RBAC (e.g., [1]) and ABAC (e.g., [2]).

An approach to ABAC enforcement within MQTT-based IoT ecosystems has been proposed in [2]. The framework in [2] introduces: i) a customized version of ABAC [4], designed to regulate the flow of messages exchanged within MQTT ecosystems, and ii) a monitor that allows enforcing policies specified according to the proposed access control model. Two classes of policies have been supported: access control policies specified by security administrators, which regulate MQTT clients' right to send / receive messages on given set of topics, and user preferences, which allow a user to restrict the privileges granted by access control policies for messages published by any of his/her MQTT clients. The enforcement monitor proposed in [2] alters the flow of messages exchanged by MQTT clients in such a way that any message receipt complies with the specified access control policies and user preferences.

Due to the potentially numerous MQTT clients and policies, it is fundamental that security administrators could use tools to analyze the effects of policies/preferences on the exchanged messages, and the decisions taken by the enforcement monitor. This is particular crucial since in [2] authorizations are implicitly denoted through conditions over subject/object and environment attributes.

In this demonstration, we illustrate Mammoth (Monitoring the ABAC Monitor of MQTT-based Internet Of THings ecosystems), a tool that has been specifically designed to fulfill the aforementioned requirements. Mammoth allows security administrators to perform real-time monitoring of the enforcement monitor activities. It provides services to analyze the message flow produced and consumed by any MQTT client, and the access control policies and user preferences that regulate the access to messages on given message topics. Mammoth provides a dashboard of tools to analyze the flow of messages exchanged within the administered ecosystems and the effects of the specified policies/preferences. Mammoth can operate at different granularity levels, and allows specifying at run-time the topics, clients, users, policies and preferences to be analyzed.

The remainder of this proposal is organized as follows. In Section 2 we shortly present the access control policies and user preferences introduced in [2] and supported by Mammoth. In Section 3, we present the architecture of the framework that hosts Mammoth and which will be used for our demonstration. Finally, in Section 4 we discuss how we would like to handle the demonstration.

2 THE ACCESS CONTROL MODEL

In this section we shortly describe key concepts of the ABAC model presented in [2] and supported by Mammoth, which allows regulating the publishing and reception of messages by MQTT clients, on the basis of access control policies and user preferences.

Access control policies are specified by security administrators, and grant clients the right to receive messages on a subscribed set of topics, or to publish a new message on a topic. In addition, a user, at any time, can specify preferences that restrict the read privileges granted by access control policies, constraining the reception of messages published by any of his/her administered client.

SACMAT '20, June 10–12, 2020, Barcelona, Spain

ACM ISBN 978-1-4503-7568-9/20/06.
https://doi.org/10.1145/3381991.3396226

Example 2.1. Let us consider an IoT application designed to monitor health conditions of the patients of a nursing home. Sensors, either worn by patients or deployed in the rooms where patients live continuously catch data that are then analyzed by medical personnel, by means of dedicated apps (e.g., physiological conditions of the nursing home inmates, as well as environmental conditions of the rooms where they live). The monitoring app can also be used by relatives authorized to check the health conditions of a hospitalized kin, and even by self sufficient patients, who can self control their own conditions. In this setting, an access control policy can for instance authorize medical personnel to monitor, during their working hours, the physiological conditions of patients hosted in the pavilions where their work.

Let us now suppose that patient *Bob* would like to restrict the privileges granted by the specified access control policies to the registered relatives. Thus, he specifies a user preference that grants the access to his physiological condition to medical personnel only.

3 SYSTEM ARCHITECTURE

The overall architecture of the system that hosts Mammoth and the ABAC framework proposed in [2] is shown in Figure 1.

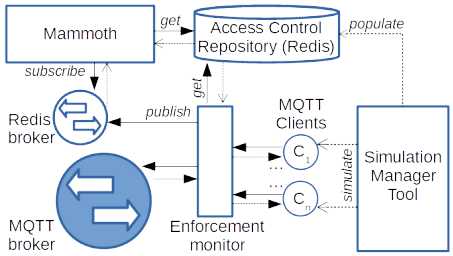

Figure 1: System architecture overview

The enforcement monitor proposed in [2] alters the flow of messages exchanged by MQTT clients and a MQTT message broker, on the basis of access control policies and user preferences. The monitor relies on an access control repository handled by a Redis datastore, which keeps track of policies, preferences, and security metadata instrumental to access control enforcement. The same repository is accessed by Mammoth for analysis purposes.

The monitor proposed in [2] has been extended to connect with a Redis message broker, to which it publishes messages encoding access requests and monitor decisions. Mammoth connects with the same broker, subscribing the receipt of these messages.

Finally, the Simulation Manager Tool, which has been introduced for the purposes of the demonstration, allows managing the simulation of an application scenario.

4 DEMONSTRATION

The demonstration relies on a simulation of the nursing home scenario introduced in Example 2.1. A simulation manager tool (smt) allows one to: i) configure the simulated scenario, ii) generate access control policies and user preferences, and iii) handle the execution of the simulation. *smt* orchestrates MQTT clients that simulate: i) the sensors worn by the patients, and the ones deployed in the rooms of the nursing home, and ii) the apps that allow medical personnel, and relatives of the inmates, to control patients' health conditions and movements, as well as the environmental conditions of the rooms where these persons live.

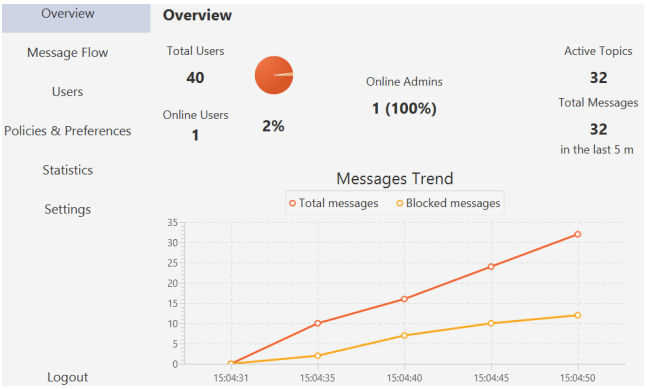

Figure 2: Mammoth GUI

Participants of the conference will be invited to use *smt* to:
(1) configure the simulation specifying the number of: i) pavilions composing the nursing home, ii) floors per pavilion, iii) rooms per floor, iv) beds per room, v) medical operators, and viii) relatives;
(2) generate a synthetically defined set of access control policies and user preferences;
(3) start/stop the simulation for the configured scenario.

Participants will then be invited to use Mammoth to analyze the activities of the enforcement monitor and the flow of messages exchanged by the simulated sensors and apps. Mammoth provides a dashboard of tools finalized at controlling different aspects of the monitored ecosystem. The graphical user interface of Mammoth, of which a screenshot has been shown in Fig. 2, is organized into panels that provide access to distinct dashboard tools. Participants will start to interact with panel *Overview* (see Fig. 2), which will give them general information on the monitored ecosystem, and then, through other panels, they will be able to perform a deep dive analysis along different dimensions, related to users, messages and policies/preferences. More precisely, by means of panel *Message Flow* they will be able to analyze the exchanged messages. Panel *Users* will allow them to control the accesses performed by users through their MQTT clients, whereas panel *Policies & preferences* will allow them to check the set of policies and preferences that have been specified. In addition, by using panel *Statistics* participants will have access to statistics related to activities performed by the enforcement monitor, whereas panel *Settings* will allow them to customize Mammoth behavior.

ACKNOWLEDGMENTS

This work has received funding from CONCORDIA, the Cybersecurity Competence Network supported by the European Union's Horizon 2020 research and innovation program under grant agreement No 830927, and from RAIS (Real-time analytics for the Internet of Sports), Marie Skłodowska-Curie Innovative Training Networks (ITN), under grant agreement No 813162.

REFERENCES

[1] A. Ben Fadhel, D. Bianculli, and L. C. Briand. 2018. Model-driven run-time enforcement of complex role-based access control policies. In *ACM ASE 2018*.

[2] P. Colombo and E. Ferrari. 2018. Access Control Enforcement within MQTT-based Internet of Things Ecosystems. In *ACM SACMAT 2018*.

[3] S. Gusmeroli, S. Piccione, and D. Rotondi. 2013. A capability-based security approach to manage access control in the Internet of Things. *Mathematical and Computer Modelling* 58, 5 (2013), 1189 – 1205.

[4] V. C. Hu, D. R. Kuhn, D. F. Ferraiolo, and J. Voas. 2015. Attribute-based access control. *Computer* 48, 2 (2015), 85–88.