AKADÉMIAI KIADÓ

# Evaluation of a policy enforcement solution in telemedicine with offline use cases

Zoltán Szabó*

Department of Software Engineering, Faculty of Science and Informatics, University of Szeged, Dugonics tér 13, H-6723 Szeged, Hungary

Pollack Periodica • An International Journal for Engineering and Information Sciences

17 (2022) 1, 12–17

DOI: 10.1556/606.2021.00383

ORIGINAL RESEARCH PAPER

Received: December 31, 2020 • Revised manuscript received: July 11, 2021 • Accepted: July 13, 2021
Published online: November 1, 2021

**ABSTRACT**

The emerging popularity of telemedicine solutions brought an alarming problem due to the lack of proper access control solutions. With the inclusion of multi-tiered, heterogeneous infrastructures containing Internet of things and edge computing elements, the severity and complexity of the problem became even more alarming, calling for an established access control framework and methodology. The goal of the research is to define a possible solution with a focus on native cloud integration, possible deployment at multiple points along the path of the healthcare data, and adaptation of the fast healthcare interoperability resources standard. In this paper, the importance of this issue in offline use cases is presented and the effectiveness of the proposed solution is evaluated.

**KEYWORDS**

telemedicine, fast healthcare interoperability resources, access-control, security, policy enforcement, heterogeneous cloud

## 1. INTRODUCTION

In recent years, the number and variance of telemedicine applications significantly increased. This has happened due to several factors: the emergence of well-defined industry standards including the popular Fast Healthcare Interoperability Resources (FHIR) [1], the availability of patient information via smartphones, simple applications and Internet of Things (IoT) devices, and furthermore, the new opportunities provided by the various cloud providers for development teams. The only problem with this newfound popularity of healthcare application development - besides the challenges and possible bottlenecks of an IoT-based sensor network [2] – has been a general lack of proper definitions and solutions to meet several important aspects of domain requirements for a completely interoperable system, as noted in the comprehensive study by Coppolino et al. [3], and also by Garai et al. [4] in their work, focusing on a possible telemedicine interoperability solution, most notably the issue of security and access control. Even FHIR, as the most widely used standard, only provides the capabilities and guidelines to implement policy enforcement and integrate with identity and access management solutions, the most widely accepted being the OAuth 2.0 framework [5], but no examples, no strict requirements for developers, even though the datasets handled and processed by these applications are highly sensitive. Several researchers found that the classical access control methods of the past are not sufficient to meet the complex requirements of the telemedicine domain - a combination and extension of these methods is needed, with frameworks that can support and enforce these policies.

The proposed solution to this problem consists of an adaptation of the Extensible Access Control Markup Language (XACML) standard from the Organization for the Advancement of Structured Information Standards (OASIS) [6], which is used at several points of the data path and even partially in applications, and allows policy enforcement and access control with a unified, portable methodology, whose current state and evaluation in offline use cases I will present in this paper.

*Corresponding author.
E-mail: szaboz@inf.u-szeged.hu

AKJournals

## 2. STATE OF THE ART

The complexity of the security problem can easily be seen in the sheer number of ongoing research projects that aim to find a partial or complete solution to it or access control in heterogeneous clouds in general. Nirojan et al. in their work [7] investigated the feasibility of federated identity and access control in heterogeneous cloud systems. Kayes et al. in [8] also examined the challenges of access control in these situations and emphasized the importance of context for a possible attribute-based solution. The solution of Veloudis et al. [9] extended the XACML standard to include an ontology-driven, attribute-based focus for protecting resources stored in the cloud.

Focusing on the telemedicine domain, Mendes et al. with their VITASENIOR-MT solution [10] also analyzed the challenges and requirements of a heterogeneous telemedicine infrastructure that includes cloud providers and IoT devices, identifying access control and security as one of the main challenges, but limiting their approach to role-based access control, while assuming that the cloud provider in the infrastructure is secure and honest about the specifics of data storage. Michalas et al., on the other hand, experimented with a method of sharing Electronic Health Records (EHRs) between different cloud providers in their health share solution [11], using attribute-based encryption to ensure the security of the data. The research by Gelenbe et al. [12] is also similar to the introduced approach, measuring and evaluating the performance of a health data sharing system.

However, this is only a small fraction of the ongoing research in this domain. In 2019, Edemacu et al. [13] compared over 100 different approaches with varying degrees of use case coverage in their comparative study focusing on attribute-based access control solutions in collaborative systems.

## 3. APPROACH

### 3.1. Policy evaluation in heterogeneous infrastructure

To address these issues, the elements of the outlined solution were inspired by the XACML standard, but adapted to support resources that implement the FHIR standard, and combined with the Open Policy Agent (OPA) policy evaluating engine [14] as its implementation, has the ability to be deployed at any point in the datapath, as it is shown in Fig. 1, to work efficiently regardless of the actual type of backend and database solutions, determine the user's access level, allow or deny operations on the data, and even transform it in certain scenarios to hide or remove parts that are unnecessary or dangerous in the context of use. To test the proposal, a prototype of a telemedicine infrastructure has been created with a sample database modeled after the FHIR document format and structure of the MIMIC-III database [15], with the policy evaluation process handled by a separate entity. Queries and responses between the client and the cloud passed through a proxy, which then forwarded them to a separate OPA server in Go runtime to execute various policies and filter or modify them before forwarding, with query limits ranging from 10 to 2,000.

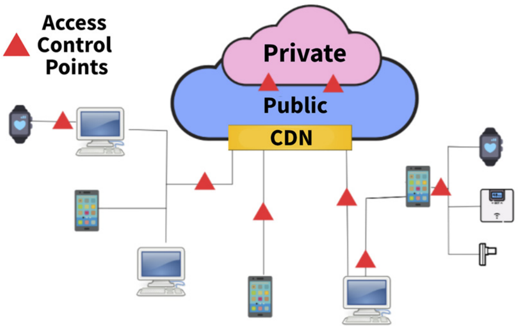

*Fig. 1.* General telemedicine infrastructure model

The evaluation of these measurements in a previous work [16] proved to be favorable, with resource consumption following a nonlinear slope and system-wide latency being relatively manageable even for 2,000 documents. The goal of this solution is to meet a specific set of requirements that have been defined as follows: the solution must be *transparent* in the system, without delay or specific requirements that signal its presence, *adaptable* to deal with the different capabilities of the different participants in the infrastructure, *portable* to be evaluated at any point in the data path as long as the documents are in standard format, and *efficient* to minimize resource consumption as much as possible, even when processing larger amounts of data.

The current step of the research is to leverage the WebAssembly runtime of the Open Policy Agent engine and evaluate how efficiently this solution can cover offline use cases when quick evaluations are required without access to the main elements of the policy. In these cases, instead of complex changes and transformations to the data, only quick evaluations are required: should the current user of the device have access or not? Just to name a few cases where this functionality is a very important requirement:

- Specific sets of the sensitive patient data are available in an offline cache on a general practitioner's device, but other users, nurses or assistants have access to it, who shouldn't be able access every document from the cache;
- The system loses connection to the security module, so the access control has to work based on locally defined, basic policies to protect sensitive data;
- If sensitive vital patient data is collected with a smartphone, the application responsible for the collecting and uploading should restrict access for other users and applications on default;
- If a new sensor or IoT device is connected, some basic policies should be enforced by the handler application even before the user defines the custom rules regarding access to his/her data.

### 3.2. Use case abstractions

To determine the exact characteristics and specifications of the offline use cases, including the average workload of a user, a benchmark was created based on the daily usage of several industrial Progressive Web Applications (PWAs) [17] created by the included development team, some of which are currently in the clinical trial phase. The benchmarked applications were the following, with each being identified by the development codename. The FOG application is used by otolaryngologists; the general practitioner of the patient is able to create recordings, pictures and videos of the patient's throat and ear, and send them through the application to otolaryngologists for analysis. The CAPD application enables practitioners to monitor the peritoneal dialysis of the patients with the app recording the various treatments and their results. The SZIVE collects vital information concerning the condition of the heart from various Bluetooth-based sensors, aggregates them and uploads them for cardiologists to analyze. The SPIRO application utilizes similar integration with a spirometer connected to the application through Bluetooth, the measurements of which are uploaded and sent to specialists. The METSZI application communicates with several components in order to monitor the metabolism of the patient and the effects of a prescribed treatment, alerting the practitioner if the values and trends diverge significantly from the predicted results. The INZULIN application fills a similar role with the difference being the monitoring of insulin treatments and blood sugar levels instead of metabolism. The PERIFERB application monitors and sends the vital signs of a patient when using a step machine, while the STRESSZ application is utilized by psychologists to monitor doctors treating patients infected Covid to monitor the level of stress and predict or even prevent a possible burnout. The results of these usage benchmarks can be seen in Table 1.

*Table 1.* Summary of various telemedicine applications and daily data traffic

| Project name | Average patients per doctor | Maximum daily uploads per patient |
|---|---|---|
| FOG | 100 | 3 |
| CAPD | 10 | 4 |
| SZIVE | 100 | 5 |
| SPIRO | 25 | 3 |
| METSZI | 60 | 8 |
| INZULIN | 60 | 8 |
| PERIFERB | 23 | 5 |
| STRESSZ | 80 | 2 |

While the MIMIC database contains, among other records, 27,854,055 laboratory measurements and 17,527,935 care values collected from 46,520 patients, it is clear from the summary that in a general, real-world application covering a single subset of telemedicine, the amount of data to be processed by a single practitioner is much more manageable, increasing the feasibility of offline use cases. Based on this, the abstract use case for evaluating the offline access control solution has the following usage metrics:

- A single practitioner handles 50–100 patients in an application;
- A single patient generates 3–8 measurements on average during a single day;
- A practitioner requires access to 75–500 documents during the course of a single day;
- All of these applications utilize paging methodologies, listing only 100–200 documents at a time.

## 3.3. Policy categories

The previous work, also defined the four main categories of policies that are able to cover the complex user requirements of the telemedicine domain. These categories are as follows:

- *Role Evaluation*: This is the simplest access control policy type, where the only task is to determine whether the current user is the primary owner (the patient), secondary owner (the practitioner), or has some other indirect, role-based access to the document (e.g., the user is part of a care team to whom the patient has granted access);
- *Contextual Evaluation*: A more complex variant of the role evaluation category, in which certain contextual elements (time, IP address, etc.,) are also taken into account in determining the level and type of access to the document in question. In the previous work, it has been recognized that these policies work most efficiently when the contextual information is provided by the client application and only checked and then used by the policy engine;
- *Contextual Modification*: An extension of contextual evaluation, where the result of the evaluation process is no longer a simple access level, but also a possibly transformed document that can be forwarded to the user;
- *Break-the-Glass*: Break-the-Glass is the most famous and difficult requirement in the field of telemedicine, cases of emergencies when there is no way to provide regular access to the user, but some degree is required to provide immediate care to a patient. In these cases, strong filtering, modification and encryption are required, which marks them as the most challenging policies in the system.

## 3.4. Test environment

From these values, the exact requirements were possible to be defined, which were later analyzed in the current phase of the research. The WebAssembly runtime of OPA currently allows only the first two categories (Role Evaluation and Contextual Evaluation) which in theory should be able to cover the identified offline use cases. The policies selected for evaluation were the following:

- *role1*: checks if the user is the secondary owner (practitioner) of the document;
- *role2*: checks if the user has indirect access as a member of a care-team;
- *context1*: checks if the status of the requested document is active and the access is requested in working hours;
- *context2*: checks the exact type of the measurement and whether a specific value exceeds a defined limit.

The goal was to check whether evaluating these policies in an application on a set of documents with 100–500 entries is feasible and efficient for users. The measurements were run multiple times in the following environments:

- Progressive Web Application on an iPad Air 2 tablet;
- Progressive Web Application on a Samsung Galaxy S6 Edge smartphone;
- Chrome 87.0.4280.88, Microsoft Edge 87.0.664.66 and Firefox 83.0 on a desktop computer with Ryzen 5 3600 central processing unit and 16 GB DDR4 RAM.

## 4. RESULTS

On the basis of the evaluations it was possible to establish the following results. First and foremost, it became clear that the latency of WebAssembly-based policy enforcement in the application was significantly slower than with the standalone OPA server, as it is shown in Fig. 2, especially after the document query size reached 500 entries. In some cases, as with 1,000 entries on role1, the evaluation in WebAssembly proved to be over 10 times slower than the standalone version. Also, after the query limit exceeded 500 documents, the latency of the WebAssembly evaluations began to converge to a linear increase, instead of the nonlinear pattern we discovered in the standalone OPA deployment.

However, as long as the size of the query stayed below 500 documents, the latency was below 1 s even in the worst case scenario, as it can be seen in Fig. 3. *This proves that if the evaluation process is combined with the paging mechanisms of the applications, they can solve access control in offline situations.* However, the policy role2 proved to be the slowest due to the breadth-first search algorithm in the array structure containing the connections between care teams and handlers, which could lead to marking these types of policies as anti-patterns in the WebAssembly runtime.

In addition, latency can improve or worsen depending on the exact operating environment, as it is shown in Fig. 4. On the desktop PC, the Edge and Chrome-based evaluations were very similar, but Firefox's results proved to be different, less efficient, due to a possible difference in the handling of heap memory, which warrants further measurements and a deeper investigation of the possible optimizations in this browser.

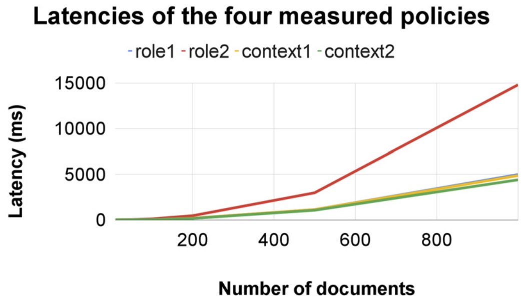

*Fig. 3.* Latency (ms) comparison between the four policies in Chrome browser with the WebAssembly runtime

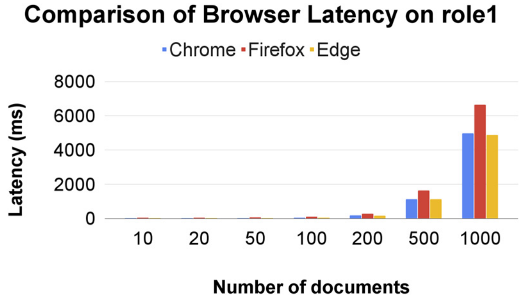

*Fig. 4.* Latency (ms) comparison on role1 between browsers on the desktop PC test environment with the WebAssembly runtime

Before the measurements, it was assumed that due to processing unit and memory constraints, latency would be highest on mobile devices. This assumption was confirmed by the results shown in Fig. 5, even though the process was slightly more efficient on Samsung Galaxy S6. *With the paging mechanism, the process can still work efficiently with a load of 100-300 documents, but even in these cases the latency is higher than on PC with the same dataset.*

Finally, a conjecture from the previous work also became confirmable, where the results at that stage suggested that the policy category Contextual Evaluation might be more efficient than the category Role Evaluation, as it is shown in Fig. 6. In the current set, the contextual attributes were

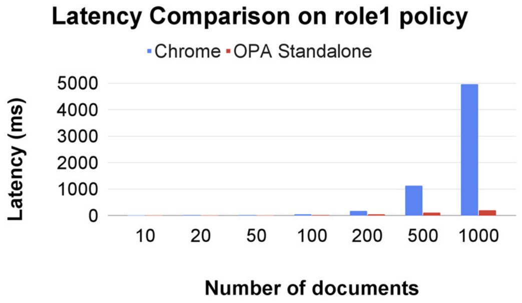

*Fig. 2.* Latency (ms) comparison between standalone OPA server and WebAssembly runtime in Chrome

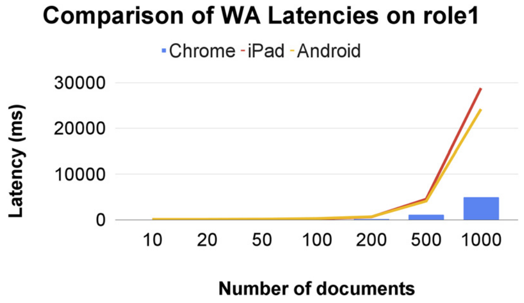

*Fig. 5.* Latency (ms) comparison on role1 between Chrome on desktop PC and mobile devices with the WebAssembly runtime

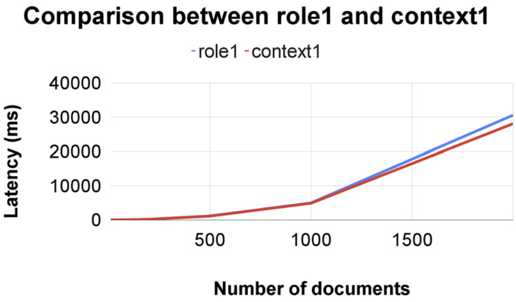

*Fig. 6.* Latency (ms) comparison between role1 and context1 on desktop PC Chrome environment with the WebAssembly runtime

provided externally to the OPA runtime, and *this produces significantly better results in the category that we originally assumed to be more complex and challenging.*

This is due to the evaluation methodology of OPA. Instead of imperative execution, OPA attempts to evaluate every condition in the policy simultaneously, and if even one of them returns with a false Boolean value, execution immediately terminates with an access denial.

Naturally, to check a single field describing whether the status of the document is ‘active’ or ‘closed’ is a significantly more efficient step than a breadth-first search in the array containing the mapping between care teams and practitioners. This leads to better latency even with the increase in document volume.

## 5. CONCLUSIONS

Based on these results, it is clear that the WebAssembly runtime of the chosen policy engine is able to cover and efficiently manage the offline use cases, even though the latency is significantly higher compared to results from earlier evaluations when running the policies on a separate standalone node. The next step in the research will be to combine these methods and further evaluate them with a runtime deployment at the edge of the cloud. Building on this, the definition of different deployment and policy definition patterns for the access control and policy evaluation solution will be possible, which can and will be verified in actual real-world use cases and infrastructures in the future to provide both definitions, guidance, and examples for efficient access control in a heterogeneous telemedicine environment, solving or at least bringing one step closer to solving one of the biggest challenges of this evolving domain.

## ACKNOWLEDGMENTS

This research was supported by the EU-funded Hungarian grants EFOP-3.6.1-16-2016-00008, EFOP-3.6.3-VEKOP-16-2017-00002, 2018-1.1.1-MKI-2018-00249 and GINOP-2.2.1-15-2017-00073.

## REFERENCES

[1] Inex-FHIR v4.0.1-HL7, [Online]. Available: http://www.hl7.org/fhir/. Accessed: Dec. 22, 2020.

[2] L. Hajdu, B. Dávid, and M. Krész, “Gateway placement and traffic load simulation in sensor networks,” *Pollack Period.*, vol. 16, no. 1, pp. 102–108, 2021.

[3] P. Natsiavas, J. Rasmussen, M. Voss-Knude, K. Votis, L. Coppolino, P. Campegiani, I. Cano, D. Marí, G. Faiella, F. Clemente, M. Nalin, E. Grivas, O. Stan, E. Gelenbe, J. Dumortier, J. Petersen, D. Tzovaras, L. Romano, I. Komnios and V. Koutkias, “Comprehensive user requirements engineering methodology for secure and interoperable health data exchange,” *BMC Med. Inform. Decis. Making*, vol. 18, 2018, Paper no. 85.

[4] Á. Garai, I. Péntek, A. Adamkó, and Á. Németh, “Methodology for clinical integration of e-Health sensor-based smart device technology with cloud architecture,” *Pollack Period.*, vol. 12, no. 1, pp. 69–80, 2017.

[5] RFC 6749: The OAuth 2.0 authorization framework. [Online]. Available: https://www.hjp.at/doc/rfc/rfc6749.html. Accessed: Dec. 22, 2020.

[6] Extensible access control markup language. [Online]. Available: http://xml.coverpages.org/xacml.html. Accessed: Dec. 22, 2020.

[7] S. Nirojan, D. Jayakody, and V. Damjanovic-Behrendt, “Federated identity management and interoperability for heterogeneous cloud platform ecosystems,” in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, Canterbury, United Kingdom, Aug. 26–29, 2019, pp. 1–7.

[8] A. S. M. Kayes, W. Rahayu, T. Dillon, E. Chang, and J. Han, “Context-aware access control with imprecise context characterization for cloud-based data resources,” *Future Generation Comput. Syst.*, vol. 93, pp. 237–255, 2019.

[9] S. Veloudis, I. Paraskakis, C. Petsos, Y. Verginadis, I. Patiniotakis, P. Gouvas, and G. Mentzas, “Achieving security-by-design through ontology-driven attribute-based access control in cloud environments”, *Future Generation Comput. Syst.*, vol. 93, pp. 373–391, 2019.

[10] D. Mendes, D. Jorge, G. Pires, R. Panda, R. António, P. Dias, and L. Oliveira, “VITASENIOR-MT: A distributed and scalable cloud-based telehealth solution,” in *IEEE 5th World Forum on Internet of Things*, Limerick, Ireland, Apr. 15–18, 2019, pp. 767–772.

[11] A. Michalas and N. Weingarten, “Healthshare: using attribute-based encryption for secure data sharing between multiple clouds,” in *IEEE 30th International Symposium on Computer-Based Medical Systems*, Thessaloniki, Greece, June 22–24, 2017, pp. 811–815.

[12] E. Gelenbe and M. Pavloski, “Performance of a security control scheme for a health data exchange system,” in *IEEE International Black Sea Conference on Communications and Networking*, Odessa, Ukraine, May 26–29, 2020, pp. 1–6.

[13] K. Edemacu, H. K. Park, B. Jang, and J. W. Kim, “Privacy provision in collaborative health with attribute-based encryption: survey, challenges and future directions,” *IEEE Access*, vol. 7, pp. 89614–89636, 2019.

[14] Open Policy Agent. [Online]. Available: https://www.openpolicyagent.org/. Accessed: Dec. 22, 2020.

[15] A. E. W. Johnson, T. J. Pollard, L. Shen, L. W. H. Lehman, M. Feng, M. Ghassemi, B. Mondy, P. Szolovits, L. A. Celi, and R. G. Mark, "MIMIC-III, a freely accessible critical care database," *Scientific Data*, vol. 3, no. 1, pp. 1–9, 2016, Paper no. 160035.

[16] Z. Szabó and V. Bilicki, "EHR data protection with filtering of sensitive information in native cloud systems," in *CSCS – The Twelfth Conference of PhD Students in Computer Science*, Institute of Informatics, University of Szeged, Hungary, June 24–26, 2020, pp. 163–166.

[17] Included. [Online] Available: https://inclouded.sed.hu/. Accessed: Dec. 30, 2020.