MODELLING AND ANALYSING CONFIDENTIALITY OF PATIENT INFORMATION

Alejandro Hernandez,
Interamerican Open University, Neuralsoft, Argentina
Tel: +54-341-425-2004, E-mail: aleh@fceia.unr.edu.ar

***Keywords**: Information security, Distributed systems, Modelling, Analysis, Interoperability*

***Biography:** Alejandro Hernandez is a Computer Scientist originally coming from Rosario, Argentina. He graduated from the National University of Rosario, and then he worked as a Software Engineer in the Industry for around 3 years, in the fields of Oil&Gas and Embedded Software for agricultural machinery. He was also involved in teaching assistant jobs for several years in his home University, including some before graduating. He has experience with Machine Learning, Artificial Intelligence, Software Engineering, and Algorithms Design, among others. In 2008, he moved to Denmark to work as research assistant and later became a PhD student at the Technical University of Denmark, working with Formal Methods for modelling and reasoning about distributed systems and security policies. During that period, he was actively involved in the Board of the PhD Association, and he even became President for one year (2011). During 2012, he was one of the two Danish delegates towards Eurodoc. He finished his PhD Studies in late 2012 and joined Microsoft Development Center Copenhagen, in Denmark. In the middle of 2013, he repatriated to Argentina, and is currently performing technical tasks at Neuralsoft, and researching and teaching at the Interamerican Open University.*

EXTENDED ABSTRACT

This extended abstract will explain one of the main applications that can be derived from the PhD Thesis by the author [1].

In this thesis, we describe a formal language for modelling and analysing distributed systems, in particular for providing distributed security by means of access control enforcing mechanisms [2]. The idea is that a distributed system can be modelled using this formal language, which uses mathematics as its base for building robust models of systems. Then, thanks to the mathematical background, automatic analyses can be done over the models, following methods also described in the thesis and in [3]. With this, we can ensure in advance that a distributed system meets some desired properties before constructing it (namely before programming it) [4].

A main application also described in the thesis is about an interoperable system of medical databases and middleware applications for sharing patient data across Europe [5]. A person who travels around Europe (either for business or holidays) can fall sick in the destination country. For a Doctor to be able to treat the person, he should be allowed to access the patient information so he gets to know some previous disease and/or treatment the patient might be following, in order to provide the best possible treatment with the current sickness. However, not any Doctor and/or principal should access information about anybody without a prior agreement by the patient, otherwise the interoperability of systems can violate confidentiality between countries.

A FORMAL LANGUAGE FOR DISTRIBUTED SYSTEMS

For achieving the results expected, we create a formal language that follows a specific syntax and semantics. The syntax allows us to create models for distributed systems following rigorous mathematical symbols. The semantics allow us to understand how the system evolves and, more importantly, allow us to later construct some software tool for understanding it *automatically*. Then, the tool can tell us if the modelled system meets some specified property before implementing it.

*To whom all correspondence should be addressed.

CREATING PROVABLE SECURE MODELS

With this language, we are able to create rigorous models of distributes systems, and then attach to them different access control security policies we might be interested in, which also follow a formal syntax and semantics.

Then, we might be interested in assuring that the entire system meets some global security property, expected in advance. With yet another formal way of expressing these, we can prove that the created model satisfies the property. Therefore, we can be sure that the system does not lack of any fundamental design error, that can lead to very complex bugs if they make it into the code.

This means that the following steps can be taken in order to create a globally secure application:

1) Create a model of the distributed system, using our syntax
2) Attach some access control security policies to each of the locations of the model
3) Describe some desired global property that we expect the system to satisfy
4) Prove, using our prototype tool, that the model with the security policies satisfy the property
5) Construct some implementation code for realising the system (outside our scope for now)

USING THE LANGUAGE TO MODEL ELECTRONIC MEDICAL INFORMATION

In the thesis [1], we demonstrate this methodology by taking an interoperable system of medical information and creating a model of it, with the necessary security policies and the desired global property, and prove that the system satisfies it. The system can be easily explained by an example, depicted in the following Figures.

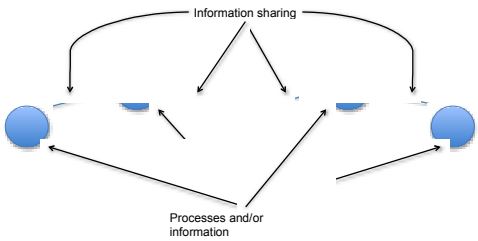

Fig. 1: Example of how locations interact

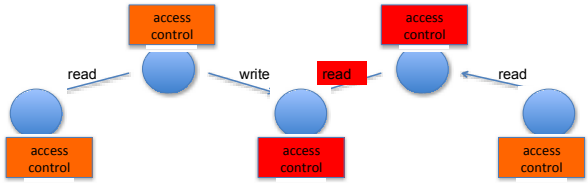

Fig. 2: Example of how access control mechanisms become relevant

Each location of the system represents some computer connected to the network. Some of these might belong to a Doctor, that is currently treating a patient, and needs to access the patient information from a database located in another country. Some middleware computer enters into play, allowing this specific Doctor, but no other, to access the information. The access control mechanisms, attached to the various locations, play a fundamental role in enforcing this, and with them we are able to prove that the global security property is satisfied.

REFERENCES

[1] Alejandro Mario Hernandez. Distributed security in closed distributed systems. PhD thesis, Technical University of Denmark, 2012.

[2] Alejandro Mario Hernandez, Flemming Nielson, and Hanne Riis Nielson. Designing, capturing and validating history-sensitive security policies for distributed systems. Scientific Annals of Computer Science, 21(1):107–149, 2011.

[3] Alejandro Mario Hernandez and Flemming Nielson. Position paper: A generic approach for security policies composition. In Proceedings of the ACM SIGPLAN 7th Workshop on Programming Languages and Analysis for Security, PLAS '12. ACM, 2012.

[4] Alejandro Mario Hernandez. Globally reasoning about localised security policies in distributed systems. Technical report, DTU, 2011.

[5] http://www.epsos.eu/home.html