

# A HIGH LEVEL SECURITY FRAMEWORK FOR THE GRID: THE JAVA CARD GRID TESTBED

Serge Chaumette  
LaBRI, UMR CNRS 5800  
Université Bordeaux 1  
351 cours de la Libération  
33405 Talence CEDEX  
FRANCE  
[http://www.labri.fr/  
serge.chaumette@labri.fr](http://www.labri.fr/serge.chaumette@labri.fr)

Damien Sauveron  
XLIM, UMR CNRS 6172  
Université de Limoges  
123, avenue Albert Thomas  
87060 Limoges CEDEX  
FRANCE  
[http://www.xlim.fr/  
damien.sauveron@xlim.fr](http://www.xlim.fr/damien.sauveron@xlim.fr)

*Keywords*— Smart Cards, Java Cards, Distributed Systems, Security, Grid

*Abstract*— The work presented in this paper is part of the Java Card<sup>TM</sup><sup>1</sup> Grid project<sup>2</sup> carried out at LaBRI, Laboratoire Bordelais de Recherche en Informatique. It consists in connecting together a number of Java Cards in a grid-like manner. The aim of this project is to build a hardware platform and the associated software components to experiment on the security features of distributed applications. We use Java Cards because they offer a high level of security thanks to their hardware architecture and to the way the software layers that they embed are tested and certified.

The current Java Card Grid platform serves as a testbed. We believe that the experimental and formal results we can get on this by nature inefficient but by nature secure grid, will make it possible to setup high level security mechanisms on regular grids. Two directions can be explored : (1) the use of big efficient Java Card-like processors, such as those being designed today by some companies; (2) the use of a Java Card Grid infrastructure jointly with a standard grid infrastructure, the former being used to support security mechanisms and services that can be offered to the latter.

The goal of this paper is to offer a synthesis of our work on the Java Card Grid project and to show that we have developed all the required features that will make it possible to explore in the short term the two directions described above.

## I. INTRODUCTION

The notion of Grid (see for instance [1] and [2]) is a well known way to connect computing resources and to federate them in an overall platform. This model potentially allows anyone to execute applications on resources that they do not own. The effective owner of the computing resources can be a university, a private company or even an individual person. Thus, the users of such platforms must accept to have their applications executed on resources that are under the control

<sup>1</sup>Java and all Java-based marks are trademarks or registered trademarks of Sun microsystems, Inc. in the United States and other countries. The authors are independent of Sun Microsystems, Inc. The other marks are the property of their respective owner.

<sup>2</sup>The Java Card Grid received the best innovative technology award at e-Smart2005.

of someone else who they potentially do not even know. The goal of this paper is to explain why smart cards can help in dealing with the resulting security issues.

In a grid context and despite all the security mechanisms commonly used, two big security concerns remain. These are:

- the enforcement of the integrity of the computing resources. If this property is not ensured, the owner of the resources should not want to share them. Thus, even though sand box approaches can solve some of the problems, it should be guaranteed that if someone uploads a code to the machine of someone else, it will not be able to do malicious operations (*e.g.* an application could take advantage of hardware errors [3]). Indeed, when a code is loaded inside a card, it can neither damage the card nor access the assets that it contains.
- the enforcement of the confidentiality and the integrity of the execution of the code. The owner of an application executed on the grid often requires strong guarantees about the security of the execution of its code in terms of confidentiality (*i.e.* it must be impossible to spy the execution – *e.g.* tracing the instructions that are executed or dumping the memory where the code has been loaded to work out what it is doing) and in terms of integrity (*i.e.* it must be impossible to modify the execution). Indeed, none of the standard workstation processors can offer this level of protection. Smart cards can ensure such constraints at hardware and software levels thanks to their tamper resistant properties and to the intermediate checking mechanisms integrated in the operating system that they embed.

Moreover, the card features described above are enforced by the security evaluation and certification process (*e.g.* ITSEC [4] - Information Technology Security Evaluation Criteria - or CC [5] - Common Criteria -) they are subjected to.

Therefore we have decided on using smart cards to experiment on secure grid computing and/or to offer secure mechanisms for real size grids. This work is known as the Java Card Grid project. It is the topic of

this paper.

The rest of this paper is organized as follows. First, in section II we describe the overall Java Card Grid platform at both hardware and software levels. In section III, we present the main challenges that we have solved and those that remain but that are most likely to be solved thanks to the evolution of the technology. We eventually discuss in section IV the future of our platform.

## II. THE JAVA CARD GRID PLATFORM

The Java Card Grid platform is both a hardware architecture and a software environment.

### A. Hardware Platform

The hardware platform is presented Fig. 1. This fig-

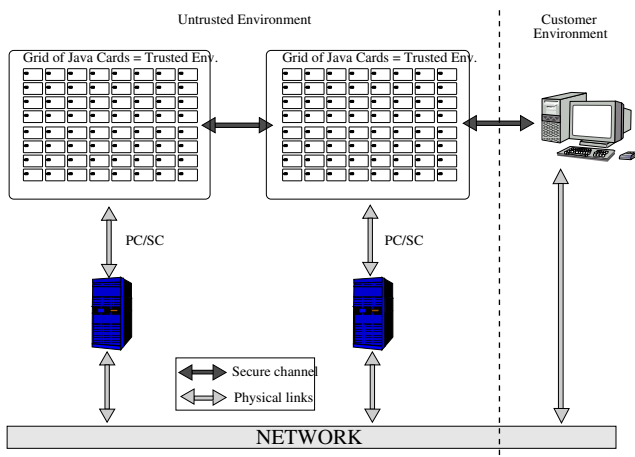


Fig. 1. A hardware platform based on Java Card grids.

ure shows that we have in fact deployed two grids that are connected together by the network. The hardware fits in a wall mount cabinet of 19U. Each grid is composed of:

- a PC which needs 2U;
- two 2U racks from SmartMount, each having 8 CCID readers from SCM Microsystems, *i.e.* we have a total of 16 CCID readers;
- three USB 7-port hubs (placed in a empty 2U rack) to connect the readers to the PC and to power the readers;
- Java Cards of different manufacturers plugged in the readers which then power them.

A picture of the real platform is shown Fig. 2.

### B. Software Framework

The software framework that we have designed and implemented comprises two layers: a low level layer that handles the PCs, the readers and thus the smart cards, and a high level layer that manages the distributed computing framework that we offer. This framework makes it possible to have codes deployed on the cards. It also makes it possible for these codes to communicate with each other. It is basically a distributed grid-like system, supplemented with all the built in security features of the Java Cards.



Fig. 2. The Java Card Grid platform.

A number of tools are also available or under development:

- a tool that supports remote administration of the grid. It makes it possible to see and control the state of the possibly many grids (two in our case) and of each reader and each card;
- a tool to deploy codes on the cards and to control them has just been prototyped. It must be tested further but it already appears to be useful to effectively use the platform.
- as a side effect, we have developed Java packages to support the GlobalPlatform<sup>3</sup> standard. They not only implement Global Platform but also make it possible to use it remotely thanks to RMI.

## III. MAIN CHALLENGES

To work with a set of smart cards in a grid-like manner, *i.e.* with codes that are distributed over the cards and that communicate with each other, we have coped with several challenges. By using the features provided by the software stack that we have implemented to solve them, it is possible to easily and quickly develop high level services or applications that will be embedded in the cards of the grid. More details can be found in [6].

In this section we describe the main issues that we had to deal with and those that remain to be solved.

### A. Solved challenges

- First, smart cards are very limited in terms of memory. To overcome this problem we have implemented a solution that makes it possible for them to support a secondary storage (the hard disk of the host), the effect of which is to increase the memory available for the embedded applications. Our solution furthermore ensures the security of the swapped information (integrity and

<sup>3</sup>GlobalPlatform is a standard to handle and manage multiapplication cards in a secure way.

confidentiality). More details about this feature are available in [7] and [8].

- Second, smart cards are passive. A card is always waiting for a command to execute, and can never initiate a communication with the outside. Because of this passive mode, the card can neither explore its environment nor interact in a dynamic manner with external components or services. To achieve a grid-like approach, the codes that we distribute over the cards must be able to communicate with each other, and thus a piece of code must be able to initiate a communication with another card. To solve this problem, we have designed and implemented a mechanism similar to what is used for SIM cards in cellular phones [9], [10]. It basically consists in tunneling or embedding the emission of a request by a card inside its answer to the previous request that it received. The details of this solution are available in [6].

- Third, the communication protocol between a card and its CAD is extremely poor. It is basically composed of sequences of bytes that obey the ISO7816-4 standard [11]. To make it easier to use, we have set up a software stack that handles all the low level details and that manages a set of smart cards in a transparent, distant, asynchronous and secure manner. More details about this stack are given in [12].

### B. Remaining challenges

Despite our developments to integrate the smart cards in a grid-like architecture, either standalone or as a real size grid support infrastructure, two main challenges still remain. First, smart cards are by design dependent on a device called the Card Acceptance Device (CAD), generally known as card reader. As of writing, such devices do not support high speed communication that would enable very efficient communication using our framework. Second, smart cards still have very limited resources in terms of computing power.

We have no way to deal with these constraints at a software level. Nevertheless, the next generation Java Cards will offer hardware solutions. First, they will use a USB connection that will suppress the need for a specific reader and that will support high speed communication. Second, very powerful processors with multithreaded operating systems are planned. These cards will also provide a TCP/IP stack, what will make them directly accessible on the network. These next generation Java Cards will also implement a communication model in which the cards will be active.

Nevertheless, it might be quite a long time before they effectively reach the market. Based on our experience gained by using our hardware infrastructure and software framework, we will then be able to adapt our solution and take the best out of these new cards, still being able to solve the remaining challenges.

## IV. CONCLUSION AND FUTURE WORK

The Java Card Grid platform as described in this paper is operational. We have developed a number of (sample) applications that have been running in the

safe context of the grid. We have solved the challenges that consisted in: (1) making the cards easier to use despite the poor protocol available to communicate with them; (2) making them proactive; (3) overcoming the memory limitation. This work got the “most innovative technology award”, delivered by a committee comprising both industry and university leaders, at e-Smart 2005 [13].

The next step consists in scaling the solution. We are in the process of defining a roadmap that will make it possible to scale the platform up to 1000 cards. By doing so, we will be able to run real size problems. We will be supported in this effort by one of the major cards companies.

In the longer term, we will also experiment with next generation cards that should provide 1 Gigabyte of memory, efficient processors, a full Java virtual machine and a TCP/IP stack.

### THANKS

Our project is supported by: Axalto, Gemplus and IBM BlueZ Secure Systems (for the cards); SCM Microsystems and SmartMount (for the readers); Sun microsystems (for the overall platform).

We also thank: Fujitsu, Giesecke&Devrient, Oberthur Card Systems and Sharp for the Java Card samples; David Corcoran and Ludovic Rousseau for their work on `pcsc-lite` and the CCID generic driver.

### REFERENCES

- [1] Berman, F., Hey, A.J., Fox, G.: Grid Computing: Making The Global Infrastructure a Reality. John Wiley & Sons (2003)
- [2] Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers (1998)
- [3] Govindavajhala, S., Appel, A.: Using Memory Errors to Attack a Virtual Machine. In: Proceedings of IEEE Symposium on Security and Privacy. (2003)
- [4] : ITSEC. ([http://www.ssi.gouv.fr/site\\_documents/ITSEC/ITSEC-fr.pdf](http://www.ssi.gouv.fr/site_documents/ITSEC/ITSEC-fr.pdf))
- [5] : International Common Criteria home page. (<http://www.commoncriteriaportal.org/>)
- [6] Chaumette, S., Karray, A., Sauveron, D.: Secure Collaborative and Distributed Services in the Java Card Grid Platform. In: Proceedings of Workshop on Collaboration and Security (COLSEC'06), Las Vegas, Nevada, USA (2006)
- [7] Chaumette, S., Karray, A., Sauveron, D.: Secure Extended Memory for Java Cards. In: Proceedings of the 2006 International Conference on Computational Science and its Applications (ICCSA 2006), Glasgow, UK (2006) (Poster).
- [8] Chaumette, S., Karray, A., Sauveron, D.: Secure storage for the Java Card Grid. (In: (Submitted))
- [9] Jurgensen, T.M., Guthery, S.B.: Smart Cards: The Developer's Toolkit. Prentice Hall (2002)
- [10] Guthery, S., Cronin, M.: Mobile Application Development with SMS and the SIM Toolkit. McGraw-Hill Professional (2001)
- [11] International Organization for Standardization: Identification cards – Integrated circuit cards – Part 4: Organization, security and commands for interchange. ISO (2005)
- [12] Chaumette, S., Karray, A., Sauveron, D.: The Software Infrastructure of a Java Card Based Security Platform for Distributed Applications. In: Proceedings of the 4th International Workshop on Security In Information Systems: WOSIS 2006. (2006)
- [13] Atallah, E., Chaumette, S., Darrigade, F., Karray, A., Sauveron, D.: A Grid of Java Cards to Deal with Security Demanding Application Domains. In: Proceedings of e-Smart 2005, Nice, France (2005)

## V. AUTHOR BIOGRAPHIES



### **SERGE CHAUMETTE**

Professor at the University Bordeaux 1, leader of the Distributed Systems and Objects team - Laboratoire Bordelais de Recherche en Informatique (LaBRI), UMR CNRS 5800. Serge Chaumette began his research activities in the domain of tools for parallel and distributed applications. Within this framework he has been using the Java technology since its early beginning. He then applied this knowledge to

design tools to help in the process of evaluating Java Cards and Java Card applications within government funded industrial projects. Java Cards are now one of the key components of the distributed software platforms developed in his research team. The Java Card Grid developed by Serge Chaumette and his team was awarded at e-Smart 2005 as the best innovative technology. He is a member of the IFIP WG 8.8 Smart Cards.

E-mail: [serge.chaumette@labri.fr](mailto:serge.chaumette@labri.fr)

Web: <http://www.labri.fr/~chaumett/>



### **DAMIEN SAUVERON**

Assistant Professor at the University of Limoges. He is a researcher at the XLIM, UMR CNRS 6172. Currently, he has a postdoctoral position at the Royal Holloway, University of London. Damien Sauveron worked during three years for the ITSEF of SERMA Technologies on the Java Card security. During his thesis that he carried out in the Distributed Systems and Objects team of the LaBRI he was one

of the main developers of a Java Card emulator, he introduced the concept of pre-persistence in Java Card and he highlighted a new category of attacks on the open multiapplication smart cards. He is one of the persons who have been involved at LaBRI in the initial conception of the Java Card Grid. Damien Sauveron is member of the IFIP WG 8.8 Smart Cards and member of the IFIP WG 11.2 Small System Security.

E-mail: [Damien.Sauveron@xlim.fr](mailto:Damien.Sauveron@xlim.fr)

Web: <http://damien.sauveron.free.fr/>