Chapter 1
Dynamic Security Properties Monitoring Architecture for Cloud Computing

Antonio Muñoz
University of Málaga, Spain

Antonio Maña
University of Málaga, Spain

Javier González
University of Málaga, Spain

ABSTRACT

In this chapter, the authors provide an overview of the importance of the monitoring of security properties in cloud computing scenarios. They then present an approach based on monitoring security properties in cloud systems based on a diagnosis framework that supports the specification and monitoring of properties expressed in Event Calculus (EC) as rules as the basis. The provision of diagnosis information is based on the generation of alternative explanations for the events that are involved in the violations of rules. This approach is based on Virtualization Architectures that presents several threats identified in the instrumentation of Virtualized Environments. The monitoring model presented in this chapter focuses on runtime supervision of applications, allowing the detection of problems in the operation of individual instances of applications and supporting the automated reconfiguration of these applications. This infrastructure has been recently designed as part of the PASSIVE project. To the best of the authors’ knowledge, there is not any other infrastructure providing the same features as the one presented in this chapter. For this reason, not much directly related previous work is found in the literature.

DOI: 10.4018/978-1-4666-2125-1.ch001
INTRODUCTION

Previous chapters show a wide range of issues related to security in software engineering oriented to cloud systems, which shows the existing problems and challenges. Monitoring security properties of software systems at runtime is widely accepted as a measure of increased resilience to security attacks. Along this chapter we present an approach based on monitoring security properties in cloud systems. A diagnosis framework that supports the specification and monitoring of properties expressed in Event Calculus (EC) (Shanahan, 1999) as rules is the basis. Then the provision of diagnosis information is based on the generation of alternative explanations for the events that are involved in the violations of rules.

The emergence of highly distributed systems operating in Virtualized Environments poses significant challenges for system security and dependability (S&D) and makes it necessary to develop mechanisms supporting the dynamic monitoring and evolution of applications running on it. In these settings, seamless and dynamic evolution of software becomes a central element for ensuring the security and dependability of systems by maintaining applications up to date and ensuring that they are used correctly. This problem increases in Virtualized Environments in which sets of applications run over several virtualized environments that in turn run in parallel over the same physical layer. The term virtualization is being used inflationary to describe various different technologies. In its most general meaning, virtualization stands for an abstraction of resources that provides a logical rather than an actual physical incarnation of those resources. A deep study about the different varieties of virtualization is out of the scope of this chapter, a further work of it can be found in (Hartig et al., 2008).

The approach presented in this chapter is based on Virtualization Architectures. These architectures presents several threats have been identified in the instrumentation of Virtualized Environments. Some of them are performed through the x86 interface. Malicious guest operating systems may attack the Virtual Machine Monitor (VMM) by exploiting any security breach of the x86 interface. As VMMs cannot attack themselves since they are not directly communicated, alternatively, a malicious VMM would need to attack any other component that is shared by different VMMs, for instance a device driver. But as the communication channel between the driver and different VMMs is separated, an attack to the driver can be avoided by simply shutting this communication point. Other types of attacks are related with device drivers. Any driver that performs Direct Memory Access (DMA) has access to the entire memory of the system, and therefore could handle completely the guest. By using an Input/Output Memory Management Unit (IOMMU) the hypervisor restricts DMA transfers to its own memory avoiding this situation. Attacks to device drivers may also be performed remotely through a network card, but due to the nature of the NOVA architecture, this would only compromise the driver.

The monitoring model presented in this chapter focuses on runtime supervision of applications, allowing the detection of problems in the operation of individual instances of applications and supporting the automated reconfiguration of these applications. However, in order to enhance trust in applications, it is necessary to analyse their behaviour when used in different Virtualized Environments. A high-level dynamic analysis can detect situations that are not possible with static or local dynamic analysis, such as problems in the implementations, in the models describing them or even problems caused by the interaction of different solutions. The results of this analysis provide a basis for taking actions that can support the evolution of applications in response to the identified problems.

This infrastructure has been recently designed as part of the PASSIVE project. To the best of our knowledge, there is not any other infrastructure providing the same features as the one we present.