Ovalyzer: an OVAL to Cfengine Translator

Martín Barrère, Rémi Badonnel, Olivier Festor

Ovalyzer: an OVAL to Cfengine Translator
(IEEE NOMS 2012 - Demo Abstract)

Martín Barrère
Ph.D Student
Madynes Research Team

LORIA - INRIA Nancy Grand Est, France
martin.barrere@inria.fr

Topic. Vulnerability Management for Safe Configurations in Autonomic Networks and Systems.

Background. The continuous growth of networks as well as the diversification of their services have considerably increased the complexity of their management. Traditional network management approaches are not suitable for supporting this sustained dynamics because they do not scale. Autonomic computing [7] provides new perspectives with respect to this issue, through the automation of the management tasks. Autonomic networks and systems are responsible for their own management. They have to adapt their configurations with respect to their environment, to protect themselves against security attacks, to repair their own failures, and to optimize their various parameters. When autonomic related operations are performed, the environment is modified in order to achieve specific objectives. However, such operations may lead to potential vulnerable states and increase the exposure to security threats. Indeed, as systems and technologies evolve, new space for vulnerabilities comes into scene. Autonomic networks and systems should therefore integrate support mechanisms for preventing vulnerabilities. As happens in the real world, autonomic elements coexist within dynamic environments, interacting with others autonomic and non-autonomic elements. If an autonomic element is compromised, its functions and abilities become untrustworthy and eventually disabled; thus autonomic elements that use services of the former become compromised as well. This inevitably leads to distrust and failure of the autonomic environment. Thus, vulnerability awareness constitutes a fundamental property that must be present in self-governed entities. Autonomic elements unable to support this capability will age with time, becoming more vulnerable, insecure and useless. Vulnerability management is a crucial activity for ensuring safe configurations and reducing the exposure of such autonomic systems. It consists in checking their configurations, identifying the presence of vulnerable states and performing the required maintenance operations (typically, modification of configuration parameters and/or application of security patches).

Objectives. In order to integrate vulnerability descriptions in the management plane of autonomic networks and systems, we have taken advantage of external knowledge sources such as OVAL repositories enabling the ability of highly increasing vulnerability awareness in such self-governed environments. Cfengine [1], a widely deployed configuration and administration system, has been taken as the autonomic part of this approach while the OVAL language is the resource that provides support for vulnerability descriptions. We have chosen the IOS platform for Cisco devices as a case study, generating Cfengine policy rules capable of analyzing and detecting vulnerabilities over such platform, thus increasing vulnerability awareness in an autonomic manner.

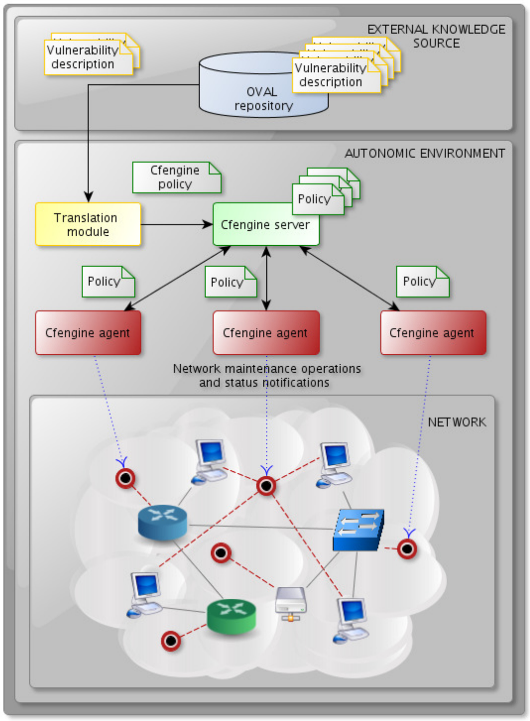

Fig. 1. High level architecture [6]

Ovalyzer. In order to provide a computable infrastructure to the proposed approach we have developed Ovalyzer, an extensible plugin-based OVAL to Cfengine translator. The translator is responsible for the translation of OVAL documents to Cfengine policy rules that represent them. The translator takes as input the content of OVAL documents and produces Cfengine code that is structured as Cfengine policy files that can be later consumed by a Cfengine running instance [6]. Figure 2 describes Ovalyzer main components and the high-level interaction between them.

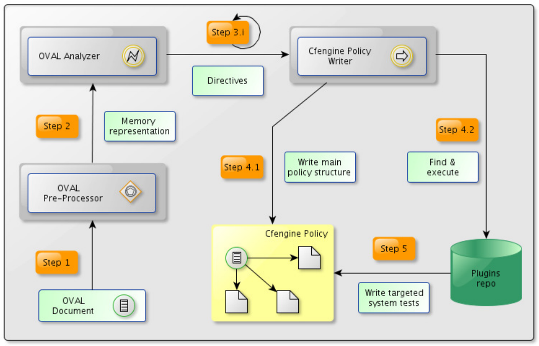

Fig. 2. Ovalyzer - High level operation [6]

At step 1, an OVAL document is consumed as the input of the translator. An OVAL preprocessor is in charge of parsing the content of the specification, adjusting some configuration aspects and feeding the OVAL analyzer module at step 2 with a memory representation of the specified input. The OVAL analyzer module is the component that orchestrates the translation flow and provides the required directives for generating Cfengine code at step 3.i. Several calls are made by the OVAL analyzer module to the Cfengine policy writer depending on the content of the OVAL document. The Cfengine policy writer is in charge of generating the main Cfengine policy entries at step 4.1 and delegating at step 4.2, specific platform rules to plugins specifically designed for generating this type of Cfengine code. Plugins will produce the required Cfengine code that will be included at step 5 inside the generated Cfengine policy files. In brief, the translator core is in charge of managing every high-level aspect of the OVAL documents it processes while available plugins provide the required functionality for generating the appropriate Cfengine code. Ovalyzer has been purely written in Java 1.6 [5] over Fedora Core [3]. The data model used by Ovalyzer is automatically generated using the JAXB technology [4]. JAXB provides means not only for modeling XML documents within a Java application data model but also for automatically reading and writing them. Such feature provides to Ovalyzer the ability to evolve with new OVAL versions with almost no developing cost. While declarative extensibility of the translator is achieved by automatic code generation using the JAXB technology, functional extensibility is supported by a plugin-based architecture.

The demo. In this demo we will show how we can increase the vulnerability awareness of self-governed environments, by feeding the autonomic system Cfengine with security advisories taken from OVAL repositories and automatically translated by Ovalyzer. During the presentation, we will first introduce the background as well as the key concepts of this work by using a small set of slides. Then we will present an scenario where an emulated Cisco router using Dynamips [2] is controlled by an autonomous Cfengine agent, and how Ovalyzer increases the vulnerability awareness of such agent by translating OVAL vulnerability descriptions into Cfengine policy rules. We will show how the approach is capable of assessing and detecting security threats by considering different vulnerable situations over the IOS platform. Vulnerability management integration into autonomic environments poses hard challenges and supporting vulnerability awareness constitutes the first step towards secure self-managed infrastructures capable of detecting and remediating potential security breaches. As to the requirements for this presentation, there are no special needs, power supply and a projector are sufficient enough.

References

[1] Cfengine. http://www.cfengine.org/. Last visited on February 7, 2012.

[2] Dynamips/Dynagen Cisco Router Emulator. http://www.dynagen.org/. Last visited on February 7, 2012.

[3] Fedora Core. http://fedoraproject.org/. Last visited on February 7, 2012.

[4] Java Architecture for XML Binding. http://java.sun.com/developer/technicalArticles/WebServices/jaxb/. Last visited on February 7, 2012.

[5] Java technology. http://www.sun.com/java/. Last visited on February 7, 2012.

[6] M. Barrre, R. Badonnel, and O. Festor. Supporting Vulnerability Awareness in Autonomic Networks and Systems with OVAL. *Proceeding of the 7th IEEE International Conference on Network and Service Management (CNSM'11)*, October 2011.

[7] Autonomic Computing. An Architectural Blueprint For Autonomic Computing. *IBM White Paper*, 2006.