Security Vulnerabilities and Exposures in Internet Systems and Services

Rui C. Cardoso
Universidade de Beira Interior, Portugal

Mário M. Freire
Universidade de Beira Interior, Portugal

INTRODUCTION

In order to guarantee a global security solution in network environments, it is necessary to take into account several issues such as security mechanisms for exchange and access to remote information; mechanisms for protection of networked systems and administrative domains; detection of new vulnerabilities and exposures; and monitoring and periodic audit of the implemented security mechanisms and disaster recovery plans.

This article is focused on the problem of detection of security vulnerabilities in an active way, using software agents. There are multiple threats to the security of computer systems and networks. The number of newly discovered vulnerabilities reported to CERT (http://www.cert.org) since 1999 continues to more than double each year. Besides, new classes of vulnerabilities are discovered each year, and subsequent reviews of existing code for examples of the new vulnerability class often lead to the discovery of evidence in hundreds of different software products. Moreover, system administrators often found themselves attacked before they even knew the existence of the vulnerability.

This article presents an overview of available software for detection of vulnerabilities and exposures in TPC/IP systems and discusses a new approach developed by the authors, based on software agents, to actively detect security vulnerabilities and exposures in Internet-based systems.

EVOLUTION OF SYSTEMS SECURITY

Network security risks are rising every day (Householder, 2002). As networks become more interconnected, the number of entry points increases and, therefore, exposes each network to threats. The widespread availability of Internet access allows the dissemination of new vulnerabilities and the know-how of hackers. While networks and applications are becoming more complex and difficult to manage, the IT industry does not appear to significantly increase the allocation of human resources to the task of securing its products. This problem is compounded by the software industry trend of shorter product lifecycles, resulting in flawed or poorly tested releases that usually have a large number of potential security weaknesses. On the other hand, hackers have suitable tools that require less technical skills and allow large-scale attacks. The time between the identification of new vulnerabilities and the exploit attempt has been reduced substantially, giving less time for administrators to patch the vulnerabilities. Moreover, hackers often have access to that information before the vendors are able to correct the vulnerabilities, in which case, it is difficult to reach the administrators in a reasonable time.

Research activities on intrusion and fault detection started in the early 1980s with the introduction of the concept of computer threats and detection of misuse by Anderson (1980). The goal of intrusion detection is simply to detect intrusions. However, intrusion detection systems (IDSs) do not detect intrusions. They only identify evidence of intrusions, either while they are in progress or after they have occurred (Manikopoulos, 2002). On the other hand, detection of security faults (holes) in hosts can anticipate the occurrence of service failures and compromises.

There are two main approaches to the problem:

1. The security companies approach mainly concentrates on the development of automated security programs capable of analyzing the attacks within a single system such as Nessus

(http://www.nessus.org), Nmap (http://www.nmap.org), SAINT (http://www.saintcorporation.com), SARA (http://www-arc.com/sara), and SNORT (http://snort.org). All of these software products use a standalone approach; they never share knowledge except when downloading updates from the central server. These tools used by systems administrators include databases of security vulnerabilities and exposures. However, there is a significant difference among them, and there is no easy way to determine when different databases are referring to the same problem. The consequences are potential gaps on the security coverage and no guaranty of effective interoperability among them. In addition, each tool currently uses different metrics to state the number of vulnerabilities or exposures they detect (Quo, 2002), which means there is no standardized basis for a common evaluation of these tools. The security organizations approach (Mell, 1999), followed by CVE (http://www.cve.mitre.com), ICAT (http://icat.nist.org), ISS (http://www.iss.net), NIST (http://www.nist.org), and SecurityFocus (http://www.securityfocus.org), make the publication of security alerts aimed at system administrators. In this case, it is difficult to reach the administrators in a reasonable time. Therefore, the need arises for cooperation among systems in order to manage such diverse sources of information.

2. The second approach is based on software agents for detection of vulnerabilities and exposures (Cardoso & Freire, 2003, 2004). The main objective of this approach is the development of a multi-agent system, where tasks are delegated to agents to make them cooperate with each other through agent communication language (ACL) in order to share information. This system allows the automation of tasks while minimizing the amount of needed human intervention. There are multiple threats and vulnerabilities in the security of computer systems and networks. By gathering information from those systems using software agents, it is possible to determine the nature of attacks against that networked systems.

This article briefly presents the design and implementation of an agent-based system built using JADE (Bellifemine, 1999). The main task of software agents is the detection of vulnerabilities and exposures (Cardoso, 2004; Humphries, 2000). Each agent can exchange knowledge with other agents in order to determine if certain suspicious situations actually are part of an attack. This procedure allows them to warn each other about possible threats. ICAT Metabase, a search index of vulnerabilities in computerized systems, was considered for external source of vulnerabilities used to the agent system up-to-date. The ICAT binds the users with diverse public databases of vulnerabilities as well as patch sites, thus allowing network administrators to find and repair the existing vulnerabilities in one given system. ICAT is not properly a database of vulnerabilities, but an index used by network administrators to know some reports of vulnerabilities as well as the information about patches currently available.

VULNERABILITY ASSESSMENT AND INTRUSION DETECTION

Vulnerability assessments (VA) tools automate the detection of vulnerabilities, allowing network administrators to assess the security status of their networks. These tools provide a means of detecting security holes before a malicious intruder. Some of them also provide a way to close them. Security policies, ACLs, and signed user agreements mean little, if systems are full of exploitable holes. Although host-based vulnerability assessment tools continue to be popular products, other solutions are arising. Host-based vulnerability assessment tools usually identify the version and distribution of the operating system (OS) running on a given host and test it for known vulnerabilities and exposures. Most of these tools test common applications and services on each platform. Application-layer vulnerability assessment tools are directed toward application servers. The difficulty of correctly securing a public server cannot be overstated. Most servers are exploitable due to underlying operating systems or holes in the applications. There are vulnerability assessment tools that cover more than one category. Only rarely will a host-based vulnerability assessment tool check for commonly