INVUS: INtelligent VUlnerability Scanner

Turker Akyuz and Ibrahim Sogukpinar

Dept. of Computer Engineering,
Gebze Institute of Technology 41400 Gebze / Kocaeli,
{takyuz, ispinar}@bilmuh.gyte.edu.tr

Abstract. This paper presents a new vulnerability scanning model named as INVUS, which contains aspects of both network and host-based vulnerability scanners. INVUS model has client/server architecture and provides an option to repair the found vulnerabilities by working with the firewall.

1 Introduction

Vulnerability scanning tools are the proactive security tools that scan computer systems in order to find exploitable points before an attacker does [1]. Most of the current vulnerability scanning tools fall into one of the two categories: host-based and network-based. While network–based scanners view the system from attacker perspective and they do not install any agent software on the target host, host-based scanners view the system from perspective of a local user and scan for password problems, inappropriate access permissions or software configuration errors [2, 3]. The main idea of the INVUS model has been improved via combining the advantages of both network and host based vulnerability scanning tools.

2 INVUS Design and Implementation

Proposed model has client/server structure which means it consists of one server and one or more clients. A general view of the proposed model is shown in Figure 1. The model is implemented on Linux operating system by using C++ language.

INVUS core which is located on the target host is responsible to manage the vulnerability scanning process. Before starting the scanning process, the user establishes a connection between the client and the server and then selects the vulnerability types to be tested by using the interface. While selected network-based vulnerability types are sent to the INVUS Server VS Engine, host-based vulnerability types are sent to the INVUS Client VS Engine as scanning parameters.

INVUS Server VS Engine focuses on network-based scanning process. This process includes operating system detection, port scanning, service detection, vulnerability searching and vulnerability proving. Services running on open ports and detailed information about the software used for these services are obtained as the result of port scanning and service detection processes. Then scanning engine starts to find known vulnerabilities related to this software. On the other hand, INVUS Client VS Engine performs host-based vulnerability scanning process. By using the results

J. Dittmann, S. Katzenbeisser, and A. Uhl (Eds.): CMS 2005, LNCS 3677, pp. 264 – 265, 2005.

obtained from scanning engines, INVUS Core creates a report. The information related to the found vulnerabilities is obtained from the vulnerability database. Also, the user is given choice to close the ports which are thought as unnecessary. This process is implemented by the help of the firewall.

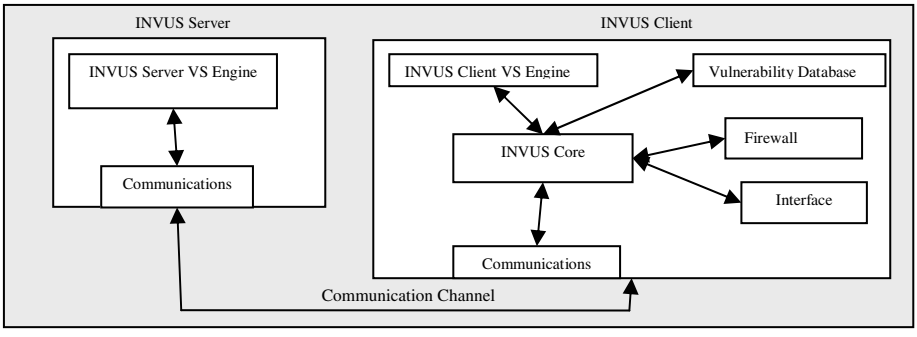

Fig. 1. INVUS model overview

3 Experimental Results and Conclusion

Proposed model has been implemented to conduct experiments. The number of host-based and network-based vulnerability types that are available to be scanned by the INVUS is increasing continuously. At that moment, INVUS can scan for 32 host-based vulnerability types; and 71 network-based vulnerability types. Some experiments were performed in order to illustrate the performance and usefulness of the INVUS model. INVUS was installed on eight different hosts running different Linux operating systems and versions. While three of these hosts were servers running services like web, ftp or mail services, the other five hosts were workstations. Results of the tests obtained from these hosts show that INVUS can detect all of the existing vulnerabilities that belong to the available vulnerability types.

In this work, a new model for vulnerability scanners is proposed. This model can scan a system for both network and host based vulnerabilities and as a result prepares comprehensive reports. By using these reports, administrators can patch vulnerabilities found. Also, the model can work with the firewall installed on the target host so vulnerable points can be repaired easily.

References

1. Venter, H. S., Eloff, J. H. P. Assessment of Vulnerability Scanners. *Network Security*, February 2003, Volume 2003, Issue 2, pp. 11-16.
2. Humphries,J.W, Carver,C.A, Pooch,U.W. Secure Mobile Agents for Network Vulnerability Scanning. *IEEE Workshop on Information Assurance and Security*, June 2000 pp 19-25.
3. Sharma, A., Martin, J.R., Anand, N., Cukier, M., Sanders, W.H. Ferret: A Host Vulnerability Checking Tool. *Proceedings of the 10th IEEE Pacific Rim International Symposium on Dependable Computing*, March 2004 pp. 389-394.