
Chapter 7. Vulnerability Mapping

Vulnerability mapping is the process of identifying and analyzing the critical security flaws in a target environment. This terminology is sometimes known as vulnerability assessment. It is one of the key areas of a vulnerability management program through which the security controls of an IT infrastructure can be analyzed against known vulnerabilities. Once the operations of information gathering, discovery, and enumeration are complete, it is time to investigate the vulnerabilities that might exist in the target infrastructure, which could lead to compromising the target and violating the confidentiality, integrity, and availability of a business system.

In this chapter, we will discuss two common types of vulnerabilities, present various standards for the classification of vulnerabilities, and explain some of the well-known vulnerability assessment tools provided under the Kali Linux operating system. This chapter constitutes the following topics:

- The concept of two generic types of vulnerabilities: local and remote.
- The vulnerability taxonomy that points to the industry standards that can be used to classify any vulnerability according to its unifying commonality pattern.
- A number of security tools that can assist us in finding and analyzing the security vulnerabilities present in a target environment. The tools presented are categorized according to their basic function in a security assessment process. These include OpenVAS, Cisco, fuzzing, SMB, SNMP, and web application analysis tools.

Note that the manual and automated vulnerability assessment procedures should be treated equally while handling any type of penetration testing assignment (internal or external). Relying strictly on automation may sometimes produce false positives and false negatives. The degree of the availability of the auditor's knowledge to technology-relevant assessment tools may be a determining factor when forming penetration tests. The tools used and the skill of the auditor should be continually updated to ensure success. Moreover, it is necessary to mention that automated vulnerability assessment is not the final solution; there are situations where the automated tools fail to identify logic errors, undiscovered vulnerabilities, unpublished software vulnerabilities, and the human variable that impacts security. Therefore, it is recommended that both automated and manual vulnerability assessment methods be used. This will heighten the probability of successful penetration tests.

Types of vulnerabilities

There are three main classes of vulnerability by which the distinction for the types of flaws (local and remote) can be made. These classes are generally divided into design, implementation, and operational categories:

- **Design vulnerabilities**: These are discovered due to the weaknesses found in the software specifications
- **Implementation vulnerabilities**: These are the technical security glitches found in the code of a system
- **Operational vulnerabilities**: These are the vulnerabilities that may arise due to the improper configuration and deployment of a system in a specific environment

Based on these three classes, we have two generic types of vulnerabilities, local and remote, which can sit in to any class of the vulnerabilities explained.

> **Note**
>
> **Which class of vulnerability is considered to be the worst to resolve?**
>
> Design vulnerability takes a developer derive the specifications based on the security requirements and address its implementation securely. Thus, it takes more time and effort to resolve the issue compared to the other classes of vulnerabilities.

Local vulnerability

A condition on which the attacker requires local access in order to trigger the vulnerability by executing a piece of code is known as local vulnerability. By taking advantage of this type of vulnerability, an attacker can increase the access privileges to gain unrestricted access to the computer.

Let's take an example in which Bob has local access to MS Windows Server 2008 (32-bit, x86 platform). His access has been restricted by the administrator through the implementation of a security policy, which will not allow him to run the specific application. Under extreme conditions, he found out that using a malicious piece of code can allow him to gain a system-level or kernel-level access to the computer. By exploiting this well-known vulnerability (for example, CVE-2013-0232, GP Trap Handler nt!KiTrap0D), he gained escalated privileges that allowed him to perform all the administrative tasks and gain unrestricted access to the application. This shows us a clear advantage that was taken by the malicious adversary to gain unauthorized access to the system.

> **Note**
>
> More information about CVE-2013-0232 MS Windows privilege escalation vulnerability can be found at: http://www.exploit-db.com/exploits/11199/.

Remote vulnerability

Remote vulnerability is a condition where the attacker has no prior access but the vulnerability can still be exploited by triggering the malicious piece of code over the network. This type of vulnerability allows an attacker to gain remote access to a computer without facing any physical or local barriers.

For instance, Bob and Alice are individually connected to the Internet. Both of them have different IP addresses and are geographically dispersed over two different regions. Let's assume that Alice's computer is running on a Windows XP operating system, which holds secret biotech information. We also assume that Bob already knows the operating system and IP address of Alice's machine. Bob is now desperately looking for a solution that can allow him to gain remote access to her computer. In the meantime, he comes to know that the MS08-067 Windows Server Service's vulnerability can be easily exploited against a Windows XP machine remotely.

Tip

More information about MS08-067 MS Windows Server Service vulnerability can be found at: http://www.exploit-db.com/exploits/6841/.

He then triggers the exploit against Alice's computer and gains full access to it.

Note

What is a relationship between vulnerability and exploit?

A vulnerability is a security weakness found in a system, which can be used by the attacker to perform unauthorized operations while the exploit takes advantage of that vulnerability or bug.