



Evening the Score: Ranking Flaws
infectionvectors.com
July 2006

Overview

What makes a flaw in software a “critical vulnerability?” Are vendor rankings trustworthy? The idea of a flaw being more or less dangerous than another is addressed by a number of models, which take some objective data and some subjective input to craft a score. This score can be integrated into a vulnerability management plan, although there is no specific means for doing this within the well-documented scoring models. This report examines a few of the models. In addition, it takes a brief look at how the output of those models could be incorporated into a security plan.

Subject

The subjective nature of model criteria is a difficult obstacle to overcome for someone hoping to inject a scientific rigor into their security programs (for more information on this issue, please see the infectionvectors.com report “Sharing the Unverifiable”¹). The products of a subjective set of data analyzed by a security guru make for a tough sell inside many organizations (much less between companies or across the Internet). There are two scenarios that are likely outcomes when a risk analysis is based on subjective data: 1) an analyst may attempt to champion a personally-interesting issue where the threat is much less than they present (a false positive), or, 2) an issue receives little attention because it is not well understood or believed to be a target for exploit coders (a false negative). Both scenarios are understandable, the flood of security data requires that an analyst filter many of the reports before they reach a CIO. As a security administrator, it may be much more practical to “err on the side of caution” and accept a good number of false positives. However, as someone charged with making business-impacting decisions, a CIO is interested in limiting the subjective element in their reports.

This type of information (threat predictions) often fills the risk manager’s fault tree – a tool used to map the introduction of a flaw with possible outcomes (and their impact). To populate such a schematic a manager looks to both the risk assessor (generally a security professional within the group) and the mathematical equations dominating probabilistic risk assessments (PRA). Data can be the result of very specific models, two of which are noted below:

Common Vulnerability Scoring System (CVSS)

The CVSS² is a model attempting to provide a standard framework to the entire security community. As noted on the website:

CVSS is designed to rank information system vulnerabilities and provide the end user with a composite score representing the overall severity and risk the vulnerability presents. Using CVSS, security professionals, executives, and end-users will have a common language with which to discuss security vulnerability severity.³

The system is made up of various categories of scores: base, temporal, and environmental. Base metrics are calculated at the time a security bulletin is released by the software manufacturer. They contain such things as the access vector. Temporal metrics change over time; they are concerned with such things as exploitability and remediation level. The third set of metrics, environmental, attempt to incorporate the unique aspects of the company using the model.

The base metrics should not change, they are assigned by the vendor. This would seem to satisfy the requirement of removing subjective scores, however, an examination of the metrics show that subjective data is needed to complete the CVSS process. When put into practice within an IT shop, determining the complexity of the attack (or more correctly, reaching a consensus on how simple an exploit is to craft) requires merging opinions into a score of “high” or “low.” In addition, the base metrics ask for scoring the degree of impact on confidentiality, availability, and integrity – scores that must be represented as “None, Partial, or Complete.” That type of scoring may not sit well with administrators seeing a big difference between 1% impact and 99%, which would receive the same score of “Partial” in the CVSS – and lead to the scorer adjusting the value in order to give the “correct” impression to the CIO.

US CERT

Another metric posited as a possible means of scoring flaws is that used by US CERT. Their method accounts for the following:

- Is information about the vulnerability widely available or known?
- Is the vulnerability being exploited in the incidents reported to US-CERT?
- Is the Internet Infrastructure at risk because of this vulnerability?
- How many systems on the Internet are at risk from this vulnerability?
- What is the impact of exploiting the vulnerability?
- How easy is it to exploit the vulnerability?
- What are the preconditions required to exploit the vulnerability?⁴

The US CERT description of the model concedes that the value produced by these criteria should not be used for strict remediation prioritization as the subjective answers will create non-standard scoring. They note, however, that the model can be used to obtain a general threat level for a given flaw/exploit.

The CERT model also introduces some very good criteria for flaw evaluation, in that many of the questions can be answered with a definitive “yes” or “no.” This is the foundation of the model introduced in “Sharing the Unverifiable.”

OSVDB

The OSVDB (Open Source Vulnerability Database)⁵ incorporates a number of the criteria that would satisfy the need for objective data, albeit without actually scoring any of them. Inside the “Vulnerability Classification” for reports in the OSVDB is a note on how the exploit is executed (i.e.: local or remote), what security tenet is at risk (confidentiality, integrity, availability), and whether an exploit is available.

For example:

“OSVDB ID: 24095
Disclosure Date: Mar 13, 2006

Description:

Microsoft Internet Explorer contains a flaw that may allow a malicious user to execute HTA files (HTML Applications) in the context of targeted users. The issue is triggered when unspecified condition occurs. It is possible that the flaw may allow to execute code and potentially to compromise affected system resulting in a loss of integrity.

Vulnerability Classification:

- Remote/Network Access Required
- Input Manipulation
- Loss Of Integrity
- Exploit Unavailable
- Verified

Products:

Microsoft Corporation Internet Explorer 6.0
Microsoft Corporation Internet Explorer 6.0 SP1
Microsoft Corporation Internet Explorer 6.0 SP2
Microsoft Corporation Internet Explorer 7.0 Beta 2 Preview (March edition) (Not Affected)”

(excerpted from OSVDB: http://www.osvdb.org/displayvuln.php?osvdb_id=24095)

Incorporated

Vulnerability scoring systems can provide a very valuable quantification of threat for CIOs. If the score is agreed upon among all groups within an organization (between IT, the security administrators, management, etc. that is no small task), then the score serves as a foundation for tactical discussions. However, across the Internet, or even between two organizations, only a score built from purely objective data will help foster practical dialog.

References

1. Linzey, Jack. "Sharing the Unverifiable: Prediction Exchange." 2006. infectionvectors.com.
2. <http://www.first.org/cvss/>
3. Schiffman, Mike. CVSS FAQ: "A Complete Guide to the CVSS" June 2005. <http://www.first.org/cvss/cvss-guide.html>.
4. US CERT Vulnerability Notes. <http://www.kb.cert.org/vuls/html/fieldhelp#metric>.
5. <http://www.osvdb.org/>