Common Vulnerability Scoring System, V3 Development Update

2013-03-15

With the release of an open letter to FIRST regarding CVSS v2, I wanted to take some time to update the security community on some of the work that has been going into the development efforts toward CVSS v3. For reference, that OSVDB letter was posted here:
http://blog.osvdb.org/2013/02/27/cvssv2-shortcomings-faults-and-failures-formulation

More details about the background including the Call for Papers and SIG selection process can be found online:
http://www.first.org/cvss/v3/development

First of all, on behalf of the SIG I'd like to express my thanks to Carsten Eiram and Brian Martin for posting their thoughts on some of the challenges with CVSS v2. Back in April 2012 the CVSS-SIG opened its Call for Subjects to solicit input from the community regarding proposed improvements to CVSS. Since June 2012, the SIG has been working on classifying and categorizing those subjects, developing a plan of attack, and working through how to implement solutions to those subjects which the SIG believes are in scope for CVSS and would represent a significant improvement to the standard.

As I read the OSVDB letter, I was struck by how similar the complaints and suggestions found in it lined up with the proposed subjects we received back in early 2012, though neither Mr. Eiram or Mr. Martin, nor any representatives from Open Security Foundation, Risk Based Security, or OSVDB submitted subjects during the Call. I believe that this provides confirmation that we have succeeded in capturing the community's requirements for improving CVSS. I'll highlight here some of the broad efforts of the SIG, which address these and other points which the community has told us need improvement in CVSS v3.

The CVSS v3 development process is ongoing; our draft is due by the end of this year, with a completed and approved specification expected in Summer 2014 (most likely to coincide with the FIRST Annual Conference). That being said, some of what I'm sharing here is work that has been completed, proposed, and voted for approval by the SIG; other work is in progress and should not be considered fully formed. Please take this letter as a view into the process, and an encouragement for more feedback.

Seth Hanford
Chair, CVSS-SIG
seth@first.org

The "Scope" Problem

By far, the most frequently and passionately communicated problem that we have heard from the community is that the v2 concept of impacts being scoped to the host operating system (Section 3.1.1, Scoring Tip #2). This problem also manifests itself in other ways, but has also become more prevalently encountered as virtualization and sandboxing moved into the mainstream since the 2007 release of CVSS v2. It has resulted in community-led solutions like Oracle's inclusion of "+" on any Partial scores which could (from a non-host-centric perspective dictated by CVSS v2) be considered a "Complete" compromise (from an application- or component-centric perspective). This problem has also meant that extra-host impacts like those encountered by network devices are very poorly represented.

This is not an easy problem to solve in a repeatable and simple manner, which promotes consistency in scoring (one of CVSS' fundamental goals). The concepts related to scoping make sense intuitively to many people, but some subtle challenges have given us some very real problems, not the least of which is documentation which would clearly solve the issue without opening the door to wildly inconsistent application of the scoring system. There is a proposal in place which would address scope in a fairly comprehensive manner, and which could directly or indirectly solve the "Cyberspace Five-O Problem", "Plus-sized Scoring Problem" (p. 3), as well as some issues expressed in "Sandbox Escape Reality Deviation" (p. 9).

CVSS v2 was a major improvement over v1, in no small part because one of the most voiced concerns (that single "Complete" impacts could score lower than multiple "Partial" impacts) was addressed. I don't believe that solving the "scope" problem is any less important for CVSS v3. It may not be possible to address this cleanly, and it may be a fundamental problem for CVSS for some time, but our hope is that it becomes a major improvement that drives many users to adopt CVSS v3 as a significant improvement over v2.

The "Scoring Tips" and Other Inconsistencies

As mentioned previously, consistency is a core requirement for CVSS. It's one of the meanings of "Common" for the "Common Vulnerability Scoring System". During the documentation process for v2, "Scoring Tips" were included with the hope that they would guide users of the system to overcome common headaches or questions that were encountered by the SIG team. Unfortunately, even the name "Tips" suggests that they might be optional.

A "Scoring Philosophy" or similar guidance, in place of Scoring Tips, should help to more clearly guide end users of CVSS. Right now, many of the letter's issues center around differing approaches used by vendors and vulnerability intelligence services / vulnerability databases that provide scores. This will never be entirely eliminated,

however there are areas where CVSS as a specification can provide clearer guidance, and we intend to do so.

Further, there are other subtle ambiguities that were uncovered during a review of metric frequencies. We have taken the stance that for v3, we will first avoid subjective choices whenever possible, and if that is not practical then we will strive for other means to limit their impact to the resulting score, including some ideas we have about weighting subjective choices for metrics more closely than we weight metric choices that have unambiguous or objective choices. For example, when considering User Interaction (a new metric in v3 with options for "None", "Simple", and "Complex") we might have a large weight difference between None and Simple, while the weights between Simple and Complex might be weighted more closely, due to an inability to clearly and objectively delineate the differences between Simple and Complex User Interaction.

Complexity and Other Multi-use Metrics

Access Complexity is a very overloaded and subjective metric in CVSS v2. When it's applied, it's impossible to tell if it's being used for user interaction from social engineering, from a race condition, uncommon configuration, attacker starting privileges, or anything else. As more or less a catch-all for CVSS v2, it has gotten a lot of negative feedback.

This certainly does not help CVSS v2 with regard to repeatable and clear scoring. As a result, the CVSS-SIG has taken a few steps to improve the consistency of Access Complexity, one of which I've already mentioned: User Interaction now stands on its own. Given the prevalence of code execution vulnerabilities that require the user to visit a malicious web site or open a malformed document since CVSS v2, we believe that there is a significant justification for specifically calling these vulnerabilities out in individual metrics. This also promotes clarity, as in the "Access Vector: Context Dependent" cases suggested in the OSVDB letter.

The SIG will also be looking at Access Vector to see if we can make better distinction between Physical and Local attackers, as those are both lumped under "Local" in v2.

Authentication vs. Privilege

Another item that rose to our attention with statistical analysis of the CVSS v2 data was that Authentication: Multiple was rarely, if ever, used. So in v3, we are looking at measuring the privileges required by the attacker, instead of whether or not they are authenticated. This will answer a number of issues with CVSS v2, including the some of the "Context Dependent" issues, "Authentication Bifurcation" (it is specifically the attacker's privileges; the user's privilege gained in a User Interaction flaw will be reflected by the Impacts), and will assist with "Locality Certainty".

Chaining vulnerabilities

Finally, there are some areas where CVSS v3 hopes to make itself more applicable to modern concerns. It's clear that CVSS should always be scoped to individual vulnerabilities. If not, there's room for inflation of severity and fear-mongering as many vulnerabilities could be combined to "make mountains out of mole hills".

But at the same time, vulnerabilities do not always exist (or get exploited) in isolation. Therefore, we hope to provide guidance on how to provide (and explicitly specify) CVSS scores for multiple related vulnerabilities. That is to say, when one or more vulnerabilities make conditions or resources available to an attacker that are required in order to exploit follow-on vulnerabilities that are also present, then it makes sense to derive a score for that chain of vulnerabilities.

In some cases, chains will expose a series of low-impact vulnerabilities that result in a final, higher impact. In others, chains will describe how rollbacks, downgrades, or regressions in software can be exploited to reintroduce prior vulnerabilities from earlier, more vulnerable versions to newer software.

In all cases, CVSS will require that each vulnerability be given its own, independent score. Then, the chain of vulnerabilities can be described and given a combined score for the chain itself. Chains might be described specifically (such as one CVE chained with one or more other CVEs) or generically (such as one or more vulnerability classes or CWEs being chained in order to exploit a specific CVE). But in the end, we believe that we could add value through CVSS to common scenarios without sacrificing the integrity of a scoring system that specifically addresses distinct vulnerabilities independent of each other.

Conclusion

The CVSS-SIG has been hard at work over the last several months, and there is quite a bit of work left to do before our target release date in June 2014. We haven't answered exhaustively all of the points raised in the OSVDB letter, but instead continue to take their points and examples alongside the other submissions to the Call for Subjects into consideration as we move on toward the goal of an improved CVSS v3 revision.

In many cases, the problems they raised can be dealt with through better documentation that drives consistency of scoring execution; in other cases, their proposals or something similar should accomplish a more granular, specific, actionable or complete scoring system. Overall, the SIG hopes that CVSS v3 will be clear, consistent, and repeatable, as well as flexible enough to handle not only the challenges that have arisen in vulnerability scoring in the last several years, but for a few years to come.

Please consider subscribing to cvss-sig@first.org, or reaching out to me directly with input. I and other members of the SIG will also be looking for opportunities to present publicly in the coming months as the drafts progress, at FIRST venues and elsewhere as we are able. Thank you again for your interest and investment in using and improving CVSS, and ultimately in working to make security more measurable and mature.