Measuring vs. Modeling

DAN GEER AND MICHAEL ROYTMAN

Dan Geer is the CISO for In-Q-Tel and a security researcher with a quantitative bent. He has a long history with the USENIX Association, including officer positions, program committees, etc.
dan@geer.org

Michael Roytman is responsible for building out Risk I/O's predictive analytics functionality. He formerly worked in fraud detection in the finance industry, and holds an MS in operations research from Georgia Tech. In his spare time he tinkers with everything from bikes to speakers to cars, and works on his pet project: outfitting food trucks with GPS.
mikeroytman@gmail.com

> It is a capital mistake to theorize before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts.
>
> —Sir Arthur Conan Doyle, 1887

Punchline: Using CVSS to steer remediation is nuts, ineffective, deeply diseconomic, and knee jerk; given the availability of data it is also passé, which we will now demonstrate.

Vulnerability data is often used to describe the vulnerabilities themselves. This is not actually interesting—it's like using footprints to describe bear paws. Sure, a black bear has different ones from a polar bear . . . but a more interesting fact is what kind of fur they have.

Strategies for vulnerability remediation often rely on true, but irrelevant, facts. The problem begins with how vulnerabilities are defined. There are several places that define vulnerabilities, but Common Vulnerabilities and Exposures (CVE), while not the most complete, is the most universal set of definitions with which we have to work. Yet thinking of CVEs as elements on the periodic table is a grave mistake; before creating synthetic polymers (read: useful analytics) out of these elements, we need to understand the biases and sources of uncertainty in the definitions themselves. For example, take a look at this finding from a research team at Concordia University in their 2011 paper "Trend Analysis of the CVE for Software Vulnerability Management" [1]:

"Our finding shows that the frequency of all vulnerabilities decreased by 28% from 2007 to 2010; also, the percentage of high severity incidents decreased for that period. Over 80% of the total vulnerabilities were exploitable by network access without authentication."

There are many such papers out there and they may be guiding organizational decision-making, but, to our point, that type of analysis misses the boat on what is being analyzed. An increase or decrease in vulnerability frequency or the enumeration of vulnerability types seen in successive time intervals can have wildly varying biases. CVE is a dictionary of known infosec vulnerabilities and exposures. It is a baseline index for assessing the coverage of tools; it is not a baseline index for the state of infosec itself.

Looking at the volume of CVEs seems to suggest that steadily increasing CVE disclosures mean "the state of security is getting worse" or some similar inference. However, CVE is not a dictionary. It is from a company attempting to streamline a process with limited resources. If you want to understand why the unit of risk we're so used to isn't a unit at all, take a look at Christey and Martin's "Buying Into the Bias: Why Vulnerability Statistics Suck" [2]

CVSS, the most widespread vulnerability scoring system, is a model for scoring the relative likelihood and impact of a given vulnerability being exploited. Among other inputs, the model takes into account impact, complexity, and likelihood of exploitation. Next, it constructs a formula based on these by fitting the model parameters to a desired distribution. This comment was made during the drafting of CVSS v2:

"Following up my previous email, I have tweaked my equation to try to achieve better separation between adjacent scores and to have CCC have a perfect (storm) 10 score...There is probably a way to optimize the problem numerically, but doing trial and error gives one

Week	CVEs affected	Breach count
1	67	754588
2	13	191
3	4	157
4	18	3948
5	15	9361
6	81	62307
7	70	41619
8	71	39914

Table 1: Breach traffic June–August 2013

plausible set of parameters...except that the scores of 9.21 and 9.54 are still too close together. I can adjust x.3 and x.7 to get a better separation . . .” [3]

So what facts is this model twisting? Well, for one, at the time of the creation of the model, there was no data available about the likelihood of an exploit. Today, we have SIEM logs with CVE attack pattern signatures, and most enterprises have both a vulnerability scanner and a SIEM installed on their networks. This allows us to correlate a CVE to the attack signature and track exploits. No need to blame the model, it's just that the theory was created, as Sherlock so aptly put, before there was any data. Moreover, when a CVE gets a score, an analyst does some research, and assigns a point-in-time likelihood value.

We can do better than that. The biggest problem with the CVSS model is not the way in which it is executed but rather what it seeks to expose. It is trying to capture (in the temporal component) a snapshot of what the live instances of attacks against these vulnerabilities look like—but it is attempting to do so without looking at any live data. Instead, the CVSS model is a static definition of the very stochastic process of exploit and breach traffic.

The present authors have access to 30 million live vulnerabilities across 1.1 million assets (hostnames, IPs, files, URLs) and 10,000 organizations. Additionally, using a different data set of 20,000 organizations' SIEM logs, analyzing them for exploit signatures, and pairing those with vulnerability scans of the same environments (data collected on the Open Threat Exchange), we construct a stochastic picture of breach traffic over the months of June to August 2013, affecting the 135 unique CVE identifiers that presented themselves in that period. No possible interpretation of that data (see Table 1) lends itself to a static conception of likelihood of exploit.

This is where the correlation gets fuzzy. The breaches come from a different set of organizations than the live vulnerabilities we have access to. However, as the sizes of both sets get bigger,

CVSS score	CVSS v1 Pr(breach)	CVSS v2 Pr(breach)
1	0.210%	0.210%
2	-0-	0.36%
3	-0-	-0-
4	1.033%	0.480%
5	0.642%	1.220%
6	0.266%	0.220%
7	0.102%	0.070%
8	0.811%	1.432%
9	2.283%	2.438%
10	4.726%	3.530%

Table 2: Probability of exploit using CVSS as the measure

the conclusions we can draw from the correlations between them gain significance. Because this is observed data, per se, we contend that it is a better indicator than the qualitative analysis done during CVSS scoring.

How much better? Let's assess a couple of possible strategies for choosing which vulnerabilities to remediate. If one chooses a vulnerability at random from the set of possible vulnerabilities, then the probability that a breach has been observed via that vulnerability is roughly 2%. This is our baseline. In Table 2 we show the probability of breach for vulnerabilities with particular CVSS scores, which pale by comparison to the probabilities of breach for vulnerabilities with entries in Exploit-DB or Metasploit or both as seen in Figure 1.

Luca Allodi from the University of Trento [4] has already done this type of analysis on the definitional level. Correlating the National Vulnerability Database (NVD) to the Symantec Threat

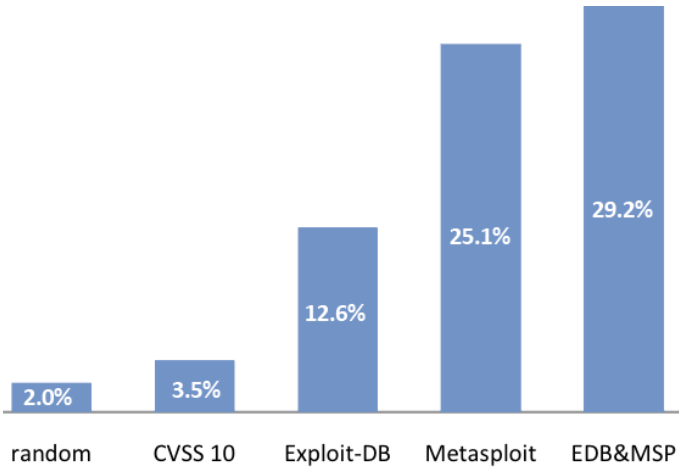

Figure 1: Probability of exploit using other measures

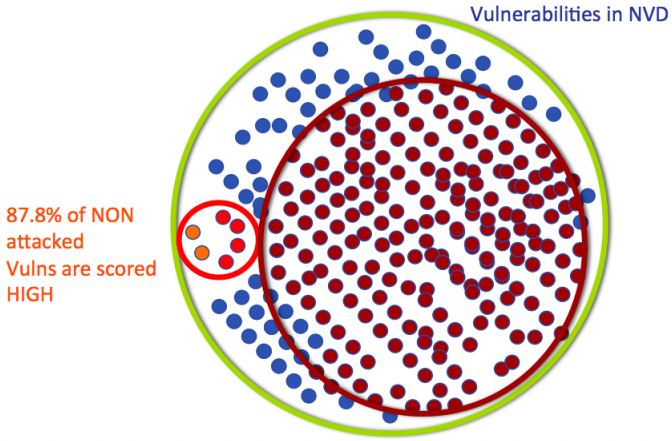

Figure 2: Attacks vs. CVSS score

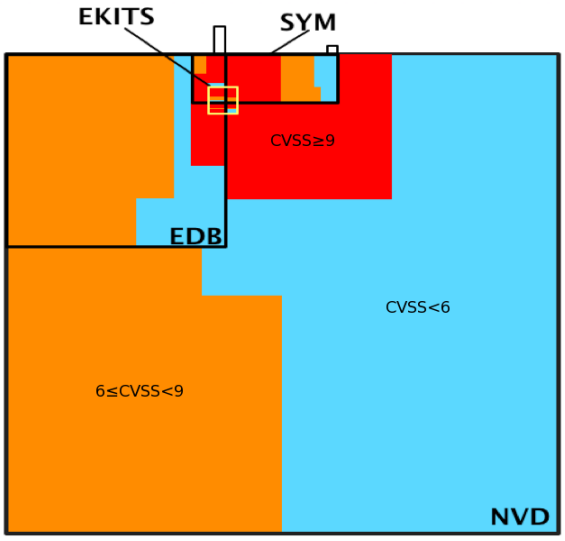

Figure 3: CVSS scores vs. EDB/EKITS/SYM/NVD

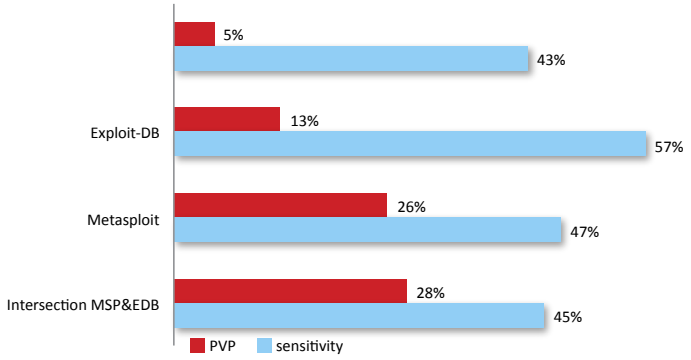

Figure 4: PVP & sensitivity comparison

Exchange in Figure 2, the outer circle encloses all NVD vulnerabilities, the smallest circle is the 2.4% of the NVD vulnerabilities that are actually attacked, and the larger interior circle represents the 87.8% of vulns that are scored ≥9 but are not attacked—which is the point: a high CVSS score does not imply impending risk in need of immediate mitigation.

Allodi's research further correlates the data with Exploit-DB and EKITS (an enumeration of CVE entries in blackhat exploit kits). Figure 3 reproduces his diagram of CVSS scores stacked against Exploit-DB, EKITS, and Symantec's Threat Exchange (to be meaningful, this figure must be viewed online at: https://www.usenix.org/publications/login/december-2013-volume-38-number-6). Dimensions are proportional to data size; vulnerabilities with CVSS ≥9 are red, vulnerabilities with 6≤CVSS<9 are orange, and vulnerabilities with CVSS<6 are cyan. The two rectangles outside of NVD space are vulnerabilities not present in NVD.

There are many entries with CVSS≥9 but with no exploit nor even any live exploit traffic. Conversely, a large portion of Exploit-DB and Symantec's intelligence go unflagged by CVSS scoring; however, this is still a definitional analysis. Visually, it is easy to see that currently adopted strategies—namely, the pervasive use of CVSS to direct remediation [5]—yield undesirable false negative rates (false positives rates are commonplace and widely accepted in remediation strategy). What is of greater interest, however, are the false positive and false negative rates of remediation strategies based on live vulnerability analysis.

Two terms of art from diagnostic testing are *predictive value positive* (PVP), the proportion of positive test results that are true positives, and *sensitivity*, the proportion of true positives that test positive. Using the same data set as above, in Figure 4 we can now really see the value of measuring vs. modeling.

Not everyone has the kind of large scale data we have here, so what is a CISO to do? First, remember that a model is a model—understand the implications of that by collecting some data on yourself, and make a commitment to long-term longitudinal data collection. Assess how well your remediation strategy is performing against your adversaries—adversaries do this all the time; they will implement different exploit kits or simply target others if the success rates of their kits decrease. Some black-market exploit kits offer SLAs to their customers with refunds if the attacks are detected or unsuccessful. A good way to do your assessment is to use an incident response team as a way to obtain the kind of predictive value positive metrics you see above. Use more than one indicator for whether to spend the labor to remediate a particular vulnerability (as we also illustrated above). For the C-suite, being able to show a metric about the level of effectiveness of a program is important, but more important is being able to claim a reduction in the volume of

data that the security team has to sift through to get to similar results. In our data set, while the intersection of ExploitDB and Metasploit yields a marginally better sensitivity, the predictive value positive is far higher, indicating that to get the same results, the "cost" is reduced. This is a metric that is useful in practice and accessible to the C-level.

This column suggests a few measures for an efficient, impactful security practice. It is probable that there are other attributes of a vulnerability which are better indicators of breach or which increase operational efficiency. The 28% PVP we obtain here is relatively inefficient even if much better than prior art. Identifying these attributes and using them to generate better predictive metrics is key to more effective security practices.

References

[1] "Trend Analysis of the CVE for Software Vulnerability Management": dc239.4shared.com/doc/JAFW1G95/preview.html (tinyurl.com/k57gxqe).

[2] Steve Christey and Brian Martin, "Buying Into the Bias: Why Vulnerability Statistics Suck":

www.attrition.org/security/conferences/2013-07-BlackHat-Vuln_Stats-draft_22-Published.pptx (tinyurl.com/ksalk3z).

[3] Appendix D: CVSS 2.0 Base Score Equation: www.first.org/cvss/history#c8 (tinyurl.com/mex8a2x).

[4] Luca Allodi, "Risk Metrics for Vulnerabilities Exploited in the Wild": securitylab.disi.unitn.it/lib/exe/fetch.php?media=seminar-unimi-apr-13.pdf (tinyurl.com/p252aa2).

[5] Guide to Adopting and Using the Security Content Automation Protocol (SCAP) Version 1.0: csrc.nist.gov/publications/nistpubs/800-117/sp800-117.pdf (tinyurl.com/m3famtj).