A Preliminary Analysis of Vulnerability Scores for Attacks in Wild

The EKITS and SYM Datasets

Luca Allodi
University of Trento
via Sommarive 14
Povo (Tn), Italy
luca.allodi@unitn.it

Fabio Massacci
University of Trento
via Sommarive 14
Povo (Tn), Italy
fabio.massacci@unitn.it

ABSTRACT

NVD and Exploit-DB are the de facto standard databases used for research on vulnerabilities, and the CVSS score is the standard measure for risk. On open question is whether such databases and scores are actually representative of attacks found in the wild. To address this question we have constructed a database (EKITS) based on the vulnerabilities currently used in exploit kits from the black market and extracted another database of vulnerabilities from Symantec's Threat Database (SYM). Our final conclusion is that the NVD and EDB databases are not a reliable source of information for exploits in the wild, even after controlling for the CVSS and exploitability subscore. An high or medium CVSS score shows only a significant sensitivity (i.e. prediction of attacks in the wild) for vulnerabilities present in exploit kits (EKITS) in the black market. All datasets exhibit a low specificity.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection — Unauthorized access (e.g., hacking, phreaking)

Keywords

Vulnerability datasets, CVSS, security metrics

1. INTRODUCTION

Evaluation of software security has traditionally been a matter of vulnerability assessment. This approach is twofold: on one side, Vulnerability Discovery Models (VDMs) [2, 9] try to predict the number of vulnerabilities that affect a software. On the other, attack graphs [19] and attack surfaces [8] aim at assessing in which ways a system is more likely to be attacked by an adversary. Foundational to both these approaches is calculating a) the number of vulnerabilities in the software/system and b) the "risk assessment" of the vulnerabilities [5]. The most common source of vulnerabilities is the National Vulnerability Database[1] (NVD for short) which is a public and almost exhaustive historical record of disclosed vulnerabilities. Each vulnerability is published alongside with a "risk assessment" given by its CVSS score (Common Vulnerability Scoring System), a composition of sub-scores that exemplify diverse aspects of the vulnerability. Version 2 of the standard has been released few years ago [10] and a new version is being developed [2]. The intuition is that the more vulnerabilities affecting a system are reported in NVD and the higher their CVSS score is, the higher the risk assessment of a system will be. However, the figures do not add up as expected. For example, Bozorgi et al. [3] showed (as a side result) that the exploitability CVSS subscore distribution do not correlate well with existence of known exploit (as reported in Exploit-db[3], or EDB for short). Yet, there are two ways to interpret this result: the exploitability of CVSS is the wrong metric, or Bozorgi and his co-authors used the wrong DB. For example, EDB could just be used by security researchers to show off their skills, in order to obtain more lucrative contracts as penetration testers, but might not have a correlation with the actual attacks by hackers.

1.1 Our Contribution

To understand this problem more completely we need to address the following research question: *is there a good database for estimating the risk of actual attacks in the wild?* To address this question we have

1. constructed a database EKITS from the vulnerabilities used in exploit kits from the black market with 103 distinguished unique vulnerabilities;

2. extracted another database of vulnerabilities from Symantec's Threat Database (SYM) with more than one 1000 vulnerabilities.

These databases of attacks in the wild are 1-2 orders of magnitude smaller than both EDB and NVD, and the picture that they offer is radically different. The conclusion of our analysis can be summarized as follows: the NVD and EDB databases are not a reliable source of information for exploits in the wild, even after controlling for the CVSS and

BADGERS'12, October 15, 2012, Raleigh, North Carolina, USA.
Copyright 2012 ACM 978-1-4503-1661-3/12/10 ...$15.00.

[1] http://nvd.nist.gov

[2] http://www.first.org/cvss

[3] http://www.exploit-db.com/

exploitability subscore. An high or medium CVSS score shows only a significant sensitivity (i.e. prediction of attacks in the wild) for vulnerabilities present in exploit kits in the black market (EKITS). Unfortunately the latter has still a low specificity, thus requiring further investigation. The statistical significance of our conclusions is supported by a case-controlled study performed on vulnerability characteristics, here included as a mitigation to internal validity threats and briefly described in Section 5.

In the next section we present the datasets used for the analysis. Then we compare at first their high-level characteristics (§3) and then specifically with reference to the CVSS score (§4) and the exploitability subscore (§4.1). Finally, we discuss related works (§6) and conclude (§7).

2. DATASETS TO CAPTURE EXPLOITS IN THE WILD

Security studies are often concerned with assessing software security [8, 5] or exploitation trends [17]. These assessments are, most of the time, based on public vulnerability and exploit databases such as NVD (vulnerabilities) and OSVDB or EDB(exploits). However, it is not clear to what point these datasets may be representative of actual software security (i.e. are all vulnerabilities in NVD important to assess risk?) or exploitation (i.e. are all exploits in EDB a threat, or nobody is using them?).

2.1 The universe of vulnerabilities

NVD is an almost exhaustive database for disclosed vulnerabilities held by NIST[4]. It contains all CVEs currently disclosed and confirmed by software vendors. The number of vulnerabilities affecting a software is often associated with its intrinsic security. Many vulnerability discovery models have been developed on this observation [2]. However, looking at all the vulnerabilities affecting a software may prove to be a wide over-estimation of the actual risk associated with it. For example, some vulnerabilities may be of too high complexity for the attacker to be convenient to actually exploit them. Or the impact on the exploited machine too low for the exploitation being of interest for the attacker. To partially address this issue, to each vulnerability is associated a "risk score", namely the CVSS score. The CVSS score is a metric tailored around the usual *likelihood* × *impact* definition of risk. This score is the *standard-de-facto* for vulnerability risk assessment, to the point that the US Federal Government asks that IT products to manage and assess the security of IT configurations must use the NIST certified S-CAP protocol [14] to prioritize patching according to the CVSS score. To date, however, no extensive study validating the CVSS score against actual attack data has been done. Some preliminary doubts on its fitness have been shaded by Bozorgi et al. [3] that showed, as a side result, that no correlation exists between the CVSS Exploitability subscore (i.e. likelihood of exploitation) and the presence of an exploit for that vulnerability.

2.2 The "white hat" exploits market

A vulnerability is arguably more interesting when an exploitation code for it exists. The security community acknowledged this issue [3, 17] and reacted by relying on additional datasets such as EDB and OSVDB. Both these datasets cooperate with the Metasploit framework to gather data on exploits. However, it is important to note that, if an exploit is featured in EDBor OSVDB, it is not evidence that some company or individual actually reported to have suffered the exploitation in the wild. It only means some proof-of-concept exploitation code is known to exist. Moreover, proof-of-concept exploitation code may be hardly capable of crashing the vulnerable application, rather than allowing the attacker to actually exploit the vulnerability. Security researchers autonomously submit the exploitation code to the Exploit-DB team, and no justification for the submission is to be provided. We can not therefore know whether the code was submitted only after the exploitation was seen in the wild. EDB might be a "*white hat* market" for exploits, where security researchers show off their skills by publishing the exploitation code of the vulnerabilities they have discovered (and whose exploitability must be proven via some proof-of-concept code disclosed to the vendor/third party handling the vulnerability [11]). There is no evidence to conclude that exploits in EDB and exploits in the wild correlate. Studying which vulnerabilities are interesting for the attacker on the basis of data in EDB can therefore be misleading.

2.3 The black markets for exploits

A step forward in this sense is our EKITS dataset, that features vulnerabilities whose exploits are traded in the black markets. These vulnerabilities are traded as bundled in "exploitation tools", namely Exploit Kits. Given the popularity of these tools and of their alleged efficacy [18], we believe that we can consider the exploits they bundle as actual ones (as opposed to EDB's proof-of-concept/unreliable exploits). Exploit Kits are, basically, websites that the attacker deploys on some public webserver he/she owns. First, the victim is fooled in making an HTTP connection to the Exploit Kit; then, this checks for vulnerabilities on the user's system and, if any, tries to exploit them; eventually, it typically infects the victim machine with malware of some sort. Among the exploit kits considered for our study, we have the "most popular" ones as reported by Symantec in 2011 [18]. After a long process of ethnographic research, we ended up with 800+ entries and 103 unique CVEs traded in the black market. However, this dataset has many limitations. First of all, the presence of an exploit in an Exploit Kit is *not* evidence, by itself, of actual attacks. We cannot rely on any evidence of the efficacy of Exploit Kits: we cannot trust the black markets to sell proper goods. IRC black markets have already been shown to be a complete scam [7], where fake goods are sold to wanna-be-scammers (that get ultimately scammed themselves). However, observational studies support the hypothesis that these tools actually work [18]. A second limitation of the EKITS dataset is that exploits bundled in Exploit Kits are targeting only client-side and consumer applications running on Windows. EKITS is therefore in no sense to be intended as representative of exploitation trends in the wild.

2.4 Records of exploits in the wild

Obtaining reliable data on exploits in the wild is challenging. Companies are not prone to release data on the cyberattacks they suffered from, for obvious commercial and reputation reasons. To the best of our knowledge, no reliable or reputable source for attacks against corporations exists

[4] http://www.nist.gov

Table 1: Summary of our datasets

DB	Content	Collection method	#Entries
NVD	CVEs vulnerabilities	XML parsing	49624
EDB	Publicly exploited CVEs	Direct download and web parsing (for correlation with CVEs)	8189
SYM	CVEs exploited in the wild	Web parsing	1289
EKITS	CVEs in the black market	manual exploration + Contagio's Exploit pack table[11]	103

yet. On the contrary, more reliable data can be found for non-targeted attacks. Symantec keeps two public datasets of signatures for local and network threats: the AttackSignature[5] and ThreatExplorer[6] datasets. These datasets contain all the entries identified as viruses or network threats by Symantec's commercial products at a given moment. Our SYM database is directly derived from these sources. We trust it can be considered a sort of ground-truth for vulnerabilities exploited in the wild. However, it must be pointed out that this dataset is, by construction, limited to threats that Symantec identifies. These therefore mainly include threats directed against home systems, which are not, in general, victims of targeted attacks. At this point, we are not aware of any more reliable and complete source for vulnerabilities actually exploited in the wild than Symantec's Attack Signature and Threat Explorer datasets. SYM will therefore be our "ground-truth" for exploits in the wild. We don't, however, consider SYM as comprehensive or complete in any way.

Table 1 summarizes the content of each dataset and the collection methodology. The datasets are available for the scientific community upon request. We cannot disclose the name of the black-hat communities and the fora because this might hamper us from future studies. It is to be noted that in this study we are not concerned with the *effects* of vulnerability exploitation (as, for example, honeypot studies do). We are not looking at what the attackers do after, say, a privilege escalation attack. Here we are instead focusing on the coverage and significance of datasets for vulnerabilities and exploits: are they representative of actual attacks? Additional details on the collection of data are discussed in our replication guide in Appendix.

3. COMPARING DATASETS FOR ATTACKS

We performed an exploratory analysis of the data in the four datasets we obtained; we are in particular interested in understanding the coverage of the datasets and how current vulnerability metrics are distributed among the datasets. Our starting point is the SYM dataset for attacks: it contains attacks that are seen in the wild, to the point that protection is needed from an anti-virus product. We consider this as our bottom-line. Our first preliminary question is therefore the following: *Given a dataset (NVD, EDB, EKITS), what is the probability that a vulnerability it contains is going to be exploited in the wild?.*

Table 2: Conditional probability of vulnerability from a dataset being a threat

	EKITS	EDB-EKITS	NVD-(EDB+EKITS)
SYM	75.73%	4.08%	2.10%
all-SYM	24.27%	95.92%	97.90%

The table shows the conditional probability that a vulnerability v is listed by Symantec as threat knowing that it is contained in a dataset, i.e. $P(v \in SYM \mid v \in dataset)$.

NVD is essentially the entire universe of vulnerabilities, so answering the question tells how many vulnerabilities are actually dangerous. EDB identifies the vulnerabilities for which exploitable code exists, but they might not be seen in the wild because their actual exploitation is inconvenient. As we have already remarked maybe EDB is just engulfed by show-off security researchers, and therefore exploits there may not be representative of those actually deployed by attackers. EKITS is the database of vulnerabilities used by exploits kits. It has the potential to have actually dangerous vulnerabilities. Table 2 reports the likelihood of a vulnerability being a threat if it is contained in one of our datasets. Each column represents a dataset from which the intersection with the smaller ones has been ruled out: this is to avoid data overlapping that would falsify the results. The vulnerabilities whose exploits are sold in the market (EKITS) are a remarkably better predictor than those featured in the other two datasets: 75.73% of vulnerabilities in EKITS are actually monitored as actively exploited in the wild. This percentage drops dramatically when looking at the other datasets: EDB-EKITS has only 4% of actually exploited vulnerabilities, and the remaining vulnerabilities in NVD-(EDB+EKITS) are only 2% of the total. Of course, these percentages are strongly influenced by the volume of the datasets: NVD contains almost 50.000 vulnerabilities, while those monitored in the wild are less than 1.300. However, this also implies that most vulnerabilities are not interesting to the attacker, and that huge databases of vulnerabilities and exploits are not addressing the problem of actual cyber-attacks.

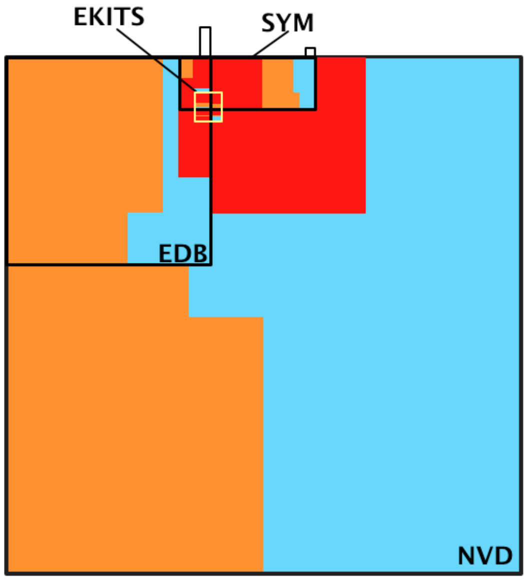
To better understand the issue we have depicted in Figure 1 the relative relations among the different datasets. The size of the area is proportional to the number of vulnerabilities and the color is an indication of the CVSS score (we discuss this more in detail in the next section). As one can see from the picture many vulnerabilities in the NVD are not exploited. The EDB is not overly better in terms or representativeness of actual exploitability in the wild: EDB and SYM share 393 vulnerabilities only. This means that EDB does not contain 75% of the threats measured by Symantec in the wild. In contrast, our EKITS dataset of vulnerabilities whose exploits are advertised in the black market overlaps with SYM for 75% of the time.

CONCLUSION 1. *The presence of a vulnerability in the EDB, i.e. if there exists a proof of concept code for the exploit, is* not *a good indication that an exploit will actually show in the wild.*

[5] http://www.symantec.com/security_response/attacksignatures/

[6] http://www.symantec.com/security_response/threatexplorer/

Figure 1: Relative Map of vulnerabilities per dataset



Dimensions are proportional to data size. In red vulnerabilities with CVSS≥9 score. Medium score vulnerabilities are orange, and cyan represents vulnerability with CVSS lower than 6. The two small rectangles outside of NVDspace are vulnerabilities whose CVEs are not present in NVD.

However, a possible counter observation would be that those databases include lots of low impact vulnerabilities; to address this we follow a further analysis on the CVSS score.

4. OBSERVATIONAL ANALYSIS OF CVSS

The histogram distribution of the CVSS scores, not depicted here, is definitely not normal across all datasets. There are essentially three clusters of vulnerabilities throughout all our datasets. We identify three corresponding categories of scores:

1. HIGH: CVSS ≥ 9
2. MEDIUM: $6 \leq$ CVSS < 9
3. LOW: CVSS < 6

In Figure 1, red, orange and cyan areas represent HIGH, MEDIUM and LOW score vulnerabilities respectively. The amount of MEDIUM and LOW vulnerabilities in the NVD dataset is disproportionally high with respect to the others. One cannot simply ignore vulnerabilities with CVSS score MEDIUM or LOW because it would miss half of the vulnerabilities that are actually exploited in the wild (SYM dataset). EDB performs better with regards to the distribution of scores: almost none of the vulnerabilities with LOW score in EDB are contained in the SYM dataset. By looking only at HIGH and MEDIUM score vulnerabilities in EDB one would deal with about 94% false positives (6140 entries out of 6533). False positives decrease to 79% (955 out of 1209) if one considers vulnerabilities with HIGH scores only. Table 3 reports the incidence of CVSS scores within each range in the dataset. 52% of vulnerabilities in the SYM dataset have a CVSS score strictly lower than 9 (665 out of 1277), and 21% are strictly lower than 6 (272): 1 out of 5 vulnerabilities exploited in the wild are ranked as "low risk vulnerabilities", and 1 out of 2 as "non-high risk" ones. At a superficial analysis, the CVSS score does not seem to be a good predictor of the actual exploitation of the vulnerability. Both the EKITS and SYM datasets feature many vulnerabilities whose CVSS score is well below HIGH. We therefore proceed with a more detailed analysis of the predictive ability of the CVSS scores.

Table 3: Incidence of CVSS scores per dataset. NVD totals do not coincide with Table 1 because 25 entries do not report CVSS score.

CVSS Score	EKITS	SYM	EDB	NVD
HIGH	74	612	1209	7026
MEDIUM	19	393	5324	20858
LOW	10	272	1589	21715
tot	**103**	**1277**	**8122**	**49599**

Table 4: Observational Specificity and Sensitivity of each dataset.

v.CVSS = H v M | v Expl.	EKITS	EDB	NVD
Sensitivity	97.4%	94.4%	78.7%
Specificity	32.0%	20.3%	44.4%

Observational sensitivity of the CVSS score being medium or high and the vulnerability being actually exploited in the wild. Specificity is the probability of the CVSS score being low and the vulnerability not being exploited in the wild.

In the medical domain, the sensitivity of a test is the conditional probability of the test giving positive results when the illness is present. The specificity of the test is the conditional probability of the test giving negative result when there is no illness. In our context, we want to assess to what degree our current test (CVSS score being HIGH or MEDIUM) predicts the illness (the vulnerability being actually exploited in the wild and tracked in SYM). This is particularly relevant because many customers and software vendors decide whether to fix the vulnerability (treat the symptom) according to the probability of the vulnerability being exploited. In this case we chose to split the scores in two: CVSS scores above 6 (MEDIUM,HIGH) are considered as positive tests; those below 6 (LOW) as negative tests. In formulae, Sensitivity=$Pr(v.score \geq 6 \mid v \in \mathsf{SYM})$ while Specificity= $Pr(v.score < 6 \mid v \notin \mathsf{SYM})$. Table 4 reports the observational specificity and sensitivity of the CVSS score for each dataset. For the CVSS score to be a good indicator within a dataset, sensitivity and specificity should be both high, at least over 90%. 90% sensitivity means that in 1 out of 10 cases when the vulnerability is exploited in the wild, it is reported as a "non-dangerous" one in the database (and vice-versa for specificity). As shown in Table 4, EDB and especially EKITS perform well in terms of sensitivity: if a vulnerability is exploited in the wild and is in one of the datasets, most of the time it was predicted to be a dangerous one. NVD performs far worse, scoring only 78.7%, meaning that the CVSS test often fails in predicting actual risk. Specificity, on the contrary, performs badly throughout all the datasets: it is not true that, if a

Table 5: Possible values for the Exploitability and Impact subscores metrics.

Exploitability subscore

Access Vector	Access complexity	Authentication
Undefined	Undefined	Undefined
Local	High	Multiple
Adjacent Net.	Medium	Single
Network	Low	None

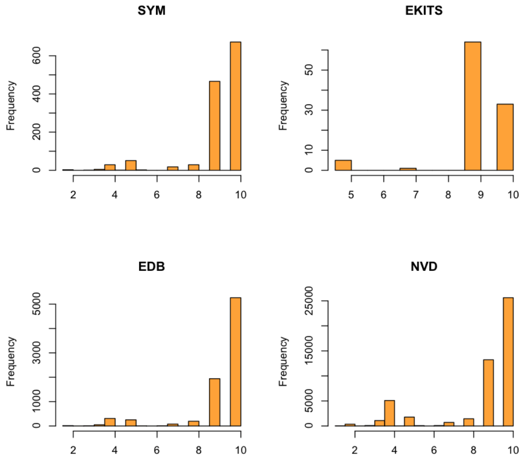


Figure 2: Distribution of CVSS Exploitability subscores.

vulnerability is not going to be exploited, then it has a low "risk score" associated. Notice that these conclusions are only based on observational data: we report *all* data without random sampling and selection. To assure statistical validity of our conclusions, we checked for control variables to sample the EDB,EKITS,NVD populations and re-run the tests. All the results show strong statistical validity. The sampling procedure and the results are briefly discussed in Section 5.

Conclusion 2. *An HIGH or MEDIUM CVSS score is* not *a good indicator that an exploit will actually show off in the wild for NVD vulnerabilities. EDB and EKITS actually show a good sensitivity, meaning that CVSS score is successful in representing exploitation. However, all the databases perform really bad in terms of specificity. The EDB database is actually the one that scores the worst: therefore, few vulnerabilities in EDB that are not showing in SYM have LOW score, showing that the CVSS score is definitely not suitable to represent non-exploitation. This scenario doesn't change with the NVD and the EKITS datasets.*

4.1 Analysis of Exploitability

As Bozorgi et al [3] already noted, the Exploitability subscore is intended to be an indicator of "likelihood to be exploited" of a vulnerability. Here, we perform an in-depth analysis to check for the significance of (a) the Exploitability subscore and (b) of the Exploitability subfactors of vulnerabilities. Table 5 reports the possible metrics for the exploitability subscore. Figure 2 shows the distribution of the Exploitability subscore per each dataset. On a first note, most vulnerabilities do turn out to have medium-high Exploitability subscore. In NVD, 84.29% of vulnerabilities (41089 out of 49599) score at least MEDIUM ($\geq$ 6), and 51.66% (25625 out of 49599) score HIGH ($\geq$9). The standard deviation for NVD is 2.2, with average 8.5. Therefore, almost the whole population of vulnerabilities is assessed as "likely to be exploited". Here, the population is basically splitted in two: the vulnerabilities with HIGH Exploitability score, and those with LOW or MEDIUM Exploitability score. One could therefore consider only HIGH Exploitability vulnerabilities to be likely to be exploited. However, this doesn't fit either: 47% (605) of SYM entries have an Exploitability subscore strictly lower than 9. The average Exploitability score for SYM is 8.99, with standard deviation 1.49. The figures are qualitatively equivalent for EKITS and EDBas well. From these observations, we conclude that

Table 6: Exploitability Subfactors for each dataset.

	metric	value	SYM	EKITS	EDB	NVD
Exploitability	Acc. Vec.	local	2.98%	0%	4.57%	13.18%
		adj.	0.23%	0%	0.12%	0.35%
		net	96.79%	100%	95.31%	87.31%
	Acc. Com.	high	4.23%	4.85%	3.37%	4.54%
		medium	38.35%	63.11%	25.49%	30.42%
		low	57.24%	32.04%	71.14%	65.68%
	Auth.	multiple	0%	0%	0.02%	0.05%
		single	3.92%	0.97%	3.71%	5.35%
		none	96.08%	99.03%	96.27%	95.45%

Conclusion 3. *The Exploitability subscore is not a suitable predictor for exploitation.*

To better investigate why Exploitability show such low variance and , we look at the details of the Exploitability subscore. Table 6 reports the total distribution of the Exploitability factors. The greatest share of actual risk comes from vulnerabilities that can be remotely exploited; despite including the host-based attacks in Symantec's threat-explorer dataset, just 3% of vulnerabilities are only locally exploitable. Moreover, the great majority of discovered vulnerabilities is network-based (87.31%). Access complexity shades some light on the willingness of the attacker in engaging in a demanding task to exploit the vulnerability; the percentage of "very difficult" vulnerabilities is equal (and very low) among all datasets. Interestingly, the ratio of "medium-complexity" vulnerabilities in the SYM and EKITS datasets is much higher than in EDB: medium-complexity vulnerabilities in the EKITS and SYM datasets are respectively 63.11% and 38.35% of the totals. As a comparison, only 25.49% of vulnerabilities in the EDB dataset have medium-complexity. Exploits in EDB are indeed mainly for very easy vulnerabilities (71.14%). Finally, Authentication turns out to be not a discriminating variable: most vulnerabilities do not require any authentication to the system.

Conclusion 4. *The CVSS Exploitability score is computed on the basis of three factors of which only one might be discriminating: almost all vulnerabilities do not require any authentication and almost all of them can be accessed remotely. Only the access complexity score shows variability within and across datasets.*

5. THREATS TO VALIDITY

During our analysis we identified a number of threats to validity [13].

Construct validity affects mainly the building process of our datasets, i.e. we need to be sure that the data we collect is meaningful and do represent the scenario we want to study. The collection mechanism is quite straightforward and no particular threat can be identified. By definition, NVD collects data on disclosed vulnerabilities and EDB collects data on public exploits. However, SYM and EKITS were much more complicated to collect.

SYM is by nature a rather unstructured, undocumented dataset. To be sure of collecting the right CVE data, we proceeded in two steps. First, we manually analyzed a random selection of about 50 entries to check for the relevance of the CVE entries in the "description" and "additional references" sections of each entry, and manually check for their relevance to the threat. To double-check our error-prone evaluation, we questioned Symantec in an informal communication: our contact confirmed that the CVEs are indeed relevant. Another issue is what data from Symantec's attack-signature (network threats) and threat-explorer (local threats) datasets to use. However, Exploit Kits enforce a drive-by download attack mechanism, and EDB and NVD do are not explicitly selective against network or local threats. There is therefore no reason to exclude the Attack signature or Threat explorer datasets from the analysis.

Similarly, EKITS was complicated to build: due to the shady nature of the tools, the list of exploited CVEs may be incomplete and/or incorrect. To mitigate the problem, we crossed-referenced entries with knowledge from the security research community and from our direct observation of the black markets. To check for the representativeness of our Exploit Kit data, we relied on databases of malicious urls such as Clean MX [7] and technical reports[8][18].

Internal validity is an issue when comparing different datasets. For example, the systems affected by the vulnerabilities in each dataset may vary in between the datasets: SYM might feature vulnerabilities for, say, Windows only, and NVD for Unix, Windows, and many others. Therefore the populations of the sampled vulnerabilities would not be comparable. However, we checked the affected systems in our datasets: SYM features vulnerabilities from all the major operative systems (Linux, Windows, MacOsX, Unix, BSD, Solaris and others) and both client and server side software. To check for validity of our results, we performed a case-controlled study with population sampling.

Vulnerability populations sampling. We looked at the CVSS subscores of the vulnerabilities in SYM and sampled EDB, NVD, EKITS to reproduce an identically distributed population. We used the *access vector, access complexity, authentication, confidentiality, integrity, availability* CVSS variables as control variables for the sampling. The samples included 580 vulnerabilities for EKITS, 1275 for EDB and 1277 for NVD (because not every possible combination was present among all dataset). Our previous conclusions remain qualitatively unchanged. To check for statistical significance, we have run Fisher's Exact test on the data (because the data is not normal). The P value for the Fisher's exact test of the EKITS sample is $p < 2.2e^{-16}$, while the P value of the EDB's and NVD's samples are respectively $p<0.0359$ and $p<0.0006$. A more detailed description of CVSS scores and subscores is scheduled for a future publication.

External validity is concerned with the applicability of our results to real-world scenarios. As our ground truth, we rely on Symantec's dataset of signatures and threats. Symantec is a world-wide diffused company and a leader in the security industry. We are therefore confident is considering their data representative of real-world scenarios, and our analysis generalizable to other settings/data. However, because of the nature of the SYM dataset, our data cannot be considered in any way representative of targeted attacks, of attacks against corporations and/or of exploits targeting vulnerabilities affecting systems not of commercial interest for Symantec. As a final note, nothing in this paper should be seen as an endorsement by Symantec of the results or conclusions.

6. RELATED WORKS

We identify two main currents in security research when it comes to vulnerability studies.

Vulnerability studies. Many studies before ours analyzed and modeled trends in vulnerabilities. Among all, Frei et al. [5] were maybe the first to link the idea of life-cycle of a vulnerability to the patching process. They showed that, according to their data, exploits are often quicker to arrive than patches are. This work have recently been extended by Shahzad et al. [17], which presented a comprehensive vulnerability study on NVD and OSVDB datasets (+ Frei's) that included vendors and software in the analysis. Many interesting trends on vulnerability patching and exploitation are presented, and support Frei's conclusion. However, they basically looked at the same data: looking at EDB or OSVDB may say little about actual threats and exploitation of vulnerabilities. An analysis of the distribution of CVSS scores and subscores has been presented by Scarfone et al. in [16] and Gallon [6]. Bozorgi et al. [3] were probably the first to look at CVSS subscores against exploitation. They showed that the "exploitability" metric, usually interpreted as "likelihood to exploit" did not match with data from EDB.

Metrics for system security. Vulnerability databases are often used to assess software and system security. *Attack surfaces* [8] rely on the number of vulnerabilities affecting a system to assess the "potential exposure" of the system to cyber-attacks. *Attack graphs* [19], similarly, try to foresee the "attack path" followed by an attacker by estimating the likelihood of him/her exploiting a vulnerability of the system. Both these approaches rely on the whole population of vulnerabilities affecting a system.

Our results show that both *Vulnerability studies* and *Metrics for system security* rely on the wrong datasets. The former shouldn't rely on broad, comprehensive dataset such as NVD and EDB because (a) the CVSS metrics are unreliable and (b) the datasets are not representative of actual threats. The latter, on the other hand, could be greatly improved by considering a narrower set of (actually exploited) vulnerabilities.

Our analysis of the vulnerabilities marketed in exploit-kits is also interesting because it confirms that the market for exploits is significantly different than the IRC markets for credit cards and other stolen goods. Indeed, dismantling some previous analysis [4], Herley et al. [7] have show that IRC black markets feature all the characteristics of a typical "market for lemons" [1]: the vendor has no drawbacks in scamming the buyer because of the complete absence of a unique-ID and of a reputation system. Moreover, the buyer

[7] http://support.clean-mx.de/clean-mx/viruses.php

[8] http://www.securelist.com/en/analysis/204792160/Exploit_Kits_A_Different_View

cannot in any way assess the quality of the good (i.e. the validity of the credit card and the amount of credit available) beforehand. In contrast, Savage et al. [12] analyzed the private messages exchanged in 6 underground forums. Most interestingly, their analysis shows that these markets feature the characteristics typical of a regular market. The results reported in this paper show that by buying exploit kits one buys something that might actually work: the exploits in exploit kits are actually seen in the wild.

7. CONCLUSIONS

NVD and Exploit-DB are the de facto standard databases used for research on vulnerabilities, and the CVSS score is the standard measure for their risk profile. In this paper we have addressed the question whether the datasets that are routinely used to asses vulnerabilities are the most appropriate to represent the risk of actual exploit in the wild. The results of our analysis are summarized as follows:

1 The existence of a proof of concept code for the exploit is *not* a good indication that an exploit will actually show in the wild.

2 For the CVSS score to be suitable as a test for exploitation, it should represent well both exploitation (sensitivity) and *non*-exploitation (specificity). However, the CVSS scores shows high sensitivity only for the EDB and EKITS datasets; very low specificity throughout all datasets show that CVSS does not represent non-exploitation.

4 The CVSS Exploitability score is computed on the basis of three factors of which only one might be discriminating: almost all vulnerabilities do not require any authentication and almost all of them can be accessed remotely. Only the access complexity score shows variability within and across datasets.

Our final conclusions are the following: *the NVD and EDB databases are not a reliable source of information for exploit in the wild, even after controlling for the CVSS and exploitability subscore. An high or medium CVSS score shows only a significant sensitivity (i.e. prediction of attacks in the wild) for vulnerabilities present in exploit kits in the black market (but unsatisfactory specificity).*

8. REFERENCES

[1] G. A. Akerlof. The market for "lemons": Quality uncertainty and the market mechanism. *The Quarterly Jour. of Econ.*, 84:pp. 488–500, 1970.

[2] O. Alhazmi and Y. Malaiya. Application of vulnerability discovery models to major operating systems. *IEEE Trans.*, 57(1):14 –22, march 2008.

[3] M. Bozorgi, L. K. Saul, S. Savage, and G. M. Voelker. Beyond heuristics: learning to classify vulnerabilities and predict exploits. In *Proc. of SIGKDD'10*, pages 105–114. ACM, 2010.

[4] J. Franklin, V. Paxson, A. Perrig, and S. Savage. An inquiry into the nature and causes of the wealth of internet miscreants. In *Proc. of CCS'07*, pages 375–388, 2007.

[5] S. Frei, M. May, U. Fiedler, and B. Plattner. Large-scale vulnerability analysis. In *Proc. of LSAD'06*, pages 131–138. ACM, 2006.

[6] L. Gallon. Vulnerability discrimination using cvss framework. In *Proc. of NTMS'11*, pages 1–6, 2011.

[7] C. Herley and D. Florencio. Nobody sells gold for the price of silver: Dishonesty, uncertainty and the underground economy. *Springer Econ. of Inf. Sec. and Priv.*, 2010.

[8] M. Howard, J. Pincus, and J. Wing. Measuring relative attack surfaces. *Comp. Sec. in the 21st Century*, pages 109–137, 2005.

[9] F. Massacci and V. Nguyen. An independent validation of vulnerability discovery models. In *Proc. of ASIACCS'12*, 2012.

[10] P. Mell and K. Scarfone. *A Complete Guide to the Common Vulnerability Scoring System Version 2.0.* CMU, 2007.

[11] C. Miller. The legitimate vulnerability market: Inside the secretive world of 0-day exploit sales. In *Proc. of WEIS'07*, 2007.

[12] M. Motoyama, D. McCoy, S. Savage, and G. M. Voelker. An analysis of underground forums. In *Proc. of IMC'11*, 2011.

[13] D. E. Perry, A. A. Porter, and L. G. Votta. Empirical studies of software engineering: a roadmap. In *Proc. of ICSE'00*, pages 345–355. ACM, 2000.

[14] S. D. Quinn, K. A. Scarfone, M. Barrett, and C. S. Johnson. Sp 800-117. guide to adopting and using the security content automation protocol (scap) version 1.0. Technical report, 2010.

[15] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[16] K. Scarfone and P. Mell. An analysis of cvss version 2 vulnerability scoring. In *Proc. of ESEM'09*, pages 516–525, 2009.

[17] M. Shahzad, M. Z. Shafiq, and A. X. Liu. A large scale exploratory analysis of software vulnerability life cycles. In *Proc. of ICSE'12*, pages 771–781. IEEE Press, 2012.

[18] Symantec. *Analysis of Malicious Web Activity by Attack Toolkits.* Symantec, Available on the web at http://www.symantec.com/threatreport/topic.jsp?id=threat_activity_trends&aid=analysis_of_malicious_web_activity, online edition, 2011. Accessed on June 1012.

[19] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia. An attack graph-based probabilistic security metric. In *Proc. of DAS'08*, volume 5094 of *LNCS*, pages 283–296. Springer, 2008.

Acknowledgement

This work as been partly supported by the project EU-FP7-SEC-CP-SECONOMICS. We would like to thank Tudor Dimitras and Viet H. Nguyen for many useful discussion. Viet actually wrote the second script used for cross-checking the SYM dataset.

APPENDIX

A. REPLICATION GUIDE

NVD and EDB are public datasets that are freely available for download.

attack_id	type	CVE	Name
aid	REF	{CVE,-}	string
aid	PAGE	{CVE,-}	string
aid	NOREF	{CVE,-}	string

attack_id	link	ref_id
aid	NONE	-
aid	THREAT	tid
aid	ASIG	aid

Table 7: Top table reports the structure of the CVE data from the attack-signatures dataset. Bottom table reports the structure for the "referenced threats" for the attack-signatures dataset. Both structures are replicated for the threat-explorer dataset.

The SYM dataset is a composition of two different datasets on Symantec's website. Both of them are non-structured non-downloadable archives, available only as webpages. We are interested in the CVEs reported in these datasets. The *attack-signatures* website contains network attack signatures that are detected by their products. The *threat-explorer* website is updated daily with host-based threats (e.g. viruses, worm, trojan horses..). Entries in this dataset provide a general description of the threat, and related vulnerabilities, if any. Each attack signature report can be divided in two parts of interest: 1) High-level description of the threat; 2) "Additional references". The first contains human-readable description of the signature. The "Additional references" section provides information about related vulnerabilities, if any, and additional links to related entries. Each attack signature is identified by an id, called *asid*. We observed that vulnerabilities are mainly referenced in the "Additional references" section. However, some of them are reported in the description of the threat. We decided to distinguish the two cases: on a first, exploratory analysis this was fundamental to assess the quality of the data, i.e. if the vulnerabilities reported in each section were relevant to the attack. Additionally, some entries in the "Additional references" of the "attack-signatures" dataset point to other attack-signatures or threats. We keep track of those as well. In some cases, an attack-signature without any CVE pointed toward another one that reports a CVE: the CVE reported in the second attack- signature might be relevant to the first. We therefore decided to keep track of those links as well. Starting from these observations, we parsed the content of both datasets on Symantec's website. We decided to write two independent parsers, one in Python language and the other in Bash, in order to double-check the results. The results were identical. Our final dataset contains 15644 entries overall, which reference 1289 unique CVEs. Table 7 reports the structure for the "CVEs" and "Links" tables of the attack-signatures datasets. Both structures are replicated, with swapped AIDs-TIDs, for the threat-explorer dataset. In the CVE table, we keep track whether the CVE was referenced in the description of the attack-signature entry (PAGE), in the "Additional references" section of the entry (REF) or if there was no "Additional references" section at all (NOREF). We keep track of the entries without a "reference" section because that may be index of a less reliable entry. The column "name" keeps track of the name of the signature. The second table reports the structure of the linked threats attack_signatures in the attack-signatures dataset. Each time a link to a second attack_signature or threat is found in the "Additional references" section of the entry, it is reported as an ASIG or THREAT link accordingly. The "ref_id" column reports the ID of the linked attack_signature threat.

To collect data for our EKITS dataset, we needed to both collect information from the community and directly explore the black markets. However, most of them are in Russian language, and are difficult to find by nature. Therefore, it was difficult to start from scratch without any particular knowledge of a) the language b) the communities of interest and c) the names of the tools. We therefore started with reading many popular security-related web-blogs to get acquainted with the problem. Unfortunately (but comprehensibly), these blogs rarely publish the names of the communities they report from, unless the communities were, in the meanwhile, closed or moved somewhere else. We have also conducted interviews with non-academic security researchers (They wish to maintain anonymity) to gather as much information as possible. Moreover, Contagio's table of exploit kits[9] has been of great help for the information gathering process: it contained a lot of precious information about most of the Exploit Kits we have already identified, and many more.

To create the sampled populations we relied on the statistical tool R [15]. Sampling was done with replication, as there is no evidence of exploitation events not being independent. On a final note, due to the small dimension, with respect to the others, of the EKITS dataset, this does not feature all the identified values of the control variables. The sample was therefore created by taking into consideration only the existing combinations of the values.

[9] http://contagiodump.blogspot.it/2010/06/overview-of-exploit-packs-update.html