My Software has a Vulnerability, should I worry?

Luca Allodi and Fabio Massacci
DISI - University of Trento
Trento, Italy
Email: lastname@disi.unitn.it

*Abstract***—Vulnerability studies usually rely on the NVD or 'proof-of-concept' exploits databases (Exploit-db, or OSVDB), while the individual vulnerability risk is measured by its CVSS score. A key issue is whether reported and evaluated vulnerabilities have been *actually exploited in the wild*, and whether the risk score do match the risk of actual exploitation.**

We compare the NVD dataset with two additional datasets, the EDBfor the white market of vulnerabilities, and the EKITS for the exploits traded in the black market. We benchmark them against Symantec's threat explorer dataset (SYM) of actual exploit in the wild. We analyze the whole spectrum of CVSS submetrics and use these characteristics to perform a case-controlled analysis of CVSS scores to test its reliability as a risk factor for actual exploitation. We conclude that EDB and NVD are the wrong databases to look at for studies that targets real exploits, (b) the CVSS score present high sensitivity (ruling in vulns for which we should worry) only for vulnerability traded in the black market, (c) we miss a metric with high specificity (ruling out vulns for which we shouldn't worry).

I. Introduction

Software vulnerabilities assessments usually rely on the National (US) Vulnerability Database ([1] (NVD for short). Each vulnerability is published with its "risk assessment" given by the Common Vulnerability Scoring System[2] (CVSS) which rate diverse aspects of the vulnerability [13].

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. For example, the US Federal government with QTA0-08-HC-B-0003 reference notice specified that IT products to manage and assess the security of IT configurations must use the NIST certified S-CAP protocol [20], which explicitly says: "*Organizations should use CVSS base scores to assist in prioritizing the remediation of known security-related software flaws based on the relative severity of the flaws.*"

The interest from industry is matched by many academic studies. On one side, Vulnerability Discovery Models [2], [12] try to predict the number of vulnerabilities that affect a software at a certain point in time, while empirical studies try to identify trends between open and closed source software [6], [24]. On the other, attack graphs [26] and attack surfaces [10] aim at assessing in which ways a system is "attackable" by an adversary and how easily he/she can succeed. Foundational to both approaches is calculating a) the number of vulnerabilities in the system and b) their individual "risk assessment".

[1]http://nvd.nist.gov
[2]http://www.first.org/cvss

Beside NVD, many datasets are used in vulnerability studies, but are they the right databases? 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 from the ExploitDB. 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. ExploitDB could just be used by security researchers to show off their skills (and obtain more contracts as penetration testers) but might not have a correlation with actual attacks by hackers. The same problem is faced by Shahzad et al. [24] who reported in the past ICSE that a large majority of "exploits" are zero-day[3]). The "exploit" time in OVSDB only measures the time when a proof-of-concept exploit becomes known. Unfortunately, security researchers normally submit proof-of-concept exploits to vendors and vulnerability white markets in order to prove that the vulnerability is worth the bounty [14]. So there is no surprise that there are a lot of zero-day exploit, but it doesn't mean that a bad hacker really exploited those vulnerabilities.

A. Our Contribution

We are interested in understanding if

1) all exploitable vulnerabilities are actually exploited in the wild (as most studies imply)?
2) are the CVSS (sub)scores a good predictor for actual exploitation (as NIST's S-CAP assumes)?

In other words, when new vulnerabilities are found, are we measuring the rate at which security researchers try to extract bounties from vendors (and should not worry)? or there is a concrete risk that bad guys end up exploiting our systems (and should worry)? This is particular interesting for the majority of *internet users at large* (individuals or corporations) who have not enough individual value to justify a targeted attack[4]. To this extent we analyzed three datasets:

- NVD, the benchmark universe of vulnerabilities;
- EDB (Exploit-DB), which contains information on the existence of proof-of-concept exploits, a good indicator of the white market of vulnerabilities;
- EKITS, our database containing vulnerabilities used in exploit kits sold in the black market.

[3]A zero-day exploit is present when the exploit is reported before or on the date that the vulnerability is disclosed.
[4]Obviously, for a nuclear power plan any proof-of-concept exploit is a problem as even a software crash may lead to a national emergency.

No previous study, to the best of our knowledge, extensively looked at CVSS *sub*scores throughout different datasets. We benchmark these DBs against the vulnerabilities exploited in the wild that we collected from Symantec's Threats and Attack Signatures databases (SYM). We have also carried out a case-controlled randomized experiment that have randomly sampled the NVD, EDB and EKITS datasets according to the SYM reported exploits; the goal is to understand the conditional probability that CVSS (sub)score would lead to an attack.

The conclusion of our analysis is the following: the NVD and EDB databases are not a reliable source of information for exploits in the wild, and the CVSS score doesn't help. The 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 no (sub)score has a high specificity, thus requiring further investigation.

The fact that EKITS vulnerabilities are actually exploited in the wild is interesting in its own sake. "Malware sales" are often scams for wanna-be scammers, such as credit-card numbers sold over IRC channels [9]. Surprisingly, while the final products (card numbers) sold on the black market are bad, the software tools to get them from the source looks good.

In the rest of the paper we introduces our four datasets (§II) and draws a first, observational comparison (§III). The core of the paper analyses the goodness of the CVSS global score as a test for exploitation (§IV), digs down over the submetrics (§V), and identify trade-off in the exploitations (§VI. Then we describes our randomized case-controlled analysis (§VII) and the (failed) attempt to find alternative association rules (§VIII). Next, we discuss the implication of our findings (§IX) and threats to valitity (§X). We finally discss related works (§XI) and concludes (§XII).

II. Datasets

NVD is the reference database for disclosed vulnerabilities held by NIST. It has been widely used and analyzed in previous vulnerability studies [11], [24], [22]. Our NVD dataset contains data on 49599 vulnerabilities.

The Exploit-db[5] (EDB) includes information on proof-of-concept exploits also represented in the Open Source Vulnerability Database (OSVDB). Both OSVDB[6] and EDB[7] derive data from Metasploit Framework. EDB references exploited CVEs by each entry in the db. Most notable studies relying on either EDB or OSVDB are [24], [3]. EDB has data on 8122 vulnerabilities for which a *proof-of-concept code* is documented and reported.

EKITS is our dataset of vulnerabilities bundled in Exploit Kits[8] sold on the black market. Given their popularity and their alleged efficacy [19], [25], they are a good starting point to investigate vulnerabilities of 'commercial interest' for attackers. After a long process of ethnographic research, starting from the "most popular" ones reported by Symantec in 2011 [25], we ended up with a list of almost 60 communities and, more importantly, 70+ Exploit Kits. We integrated the information from blogs and security reports with our direct observations, fixing and adding hundreds of entries in the database. We have 800+ entries and 103 unique CVEs. We cannot disclose the individual sources of the black-hat communities because this might hamper us from future studies.

In order to determine whether a vulnerability has been used in the wild we have collected information from Symantec's AttackSignature[9] and ThreatExplorer[10] public data. The SYM dataset contain all the entries identified as viruses (local threats) or remote attacks (network threats) by Symantec's commercial products at a given moment. It reports 1277 vulnerabilities. This has of course some limitation as direct attacks by individual motivated hackers against specific companies are not considered in this metric. To the best of our knowledge, no better public database exists because individual companies do not report attacks.

Table I summarizes the content of each dataset and the collection methodology. They are available upon request.

TABLE I
Summary of our datasets

DB	Content	Collection method	#Entries
NVD	CVEs	XML parsing	49599
EDB	Publicly exploited CVEs	Download and web parsing to correlate with CVEs	8122
SYM	CVEs exploited in the wild	Web parsing to correlate with CVEs	1277
EKITS	CVEs in the black market	ad-hoc analysis + Contagio's Exploit table	103

III. Exploratory analysis of datasets

We performed an exploratory analysis of the data in our four datasets: *Given a dataset (NVD, EDB, EKITS), what is the likelihood that a vulnerability it contains is going to be exploited in the wild?* i.e. occurs also in SYM?

Table II reports the likelihood of a vulnerability being a threat if it is contained in one of our datasets. Each row 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. Our first result confirms that vulnerabilities whose exploits are traded in the

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

[6] http://blog.osvdb.org/2012/08/15/august-2012-a-few-small-updates

[7] http://www.exploit-db.com/author/?a=3211&pg=1

[8] Exploit Kits are web sites that the attacker deploys on some public webserver he/she owns. When the victim is fooled in making an HTTP connection to the Exploit Kit, the latter checks for vulnerabilities on the user's system and, if any, tries to exploit them; eventually, it infects the victim machine with malware of some sort.

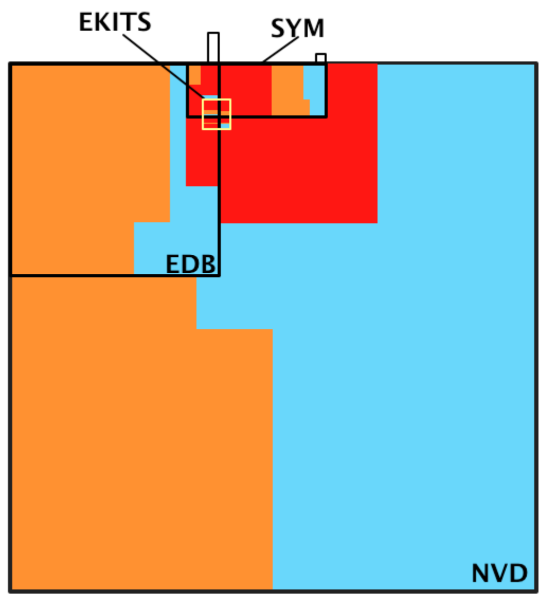
[9] http://www.symantec.com/security_response/attacksignatures/

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

TABLE II
CONDITIONAL PROB. OF VULN. FROM A DATASET BEING A THREAT

	vuln in SYM	vuln not in SYM
EKITS	75.73%	24.27%
EDB-EKITS	4.08%	95.92%
NVD-(EDB+EKITS)	2.10%	97.90%

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



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.

Fig. 1. Relative Map of vulnerabilities per dataset

black markets are actually monitored in the wild, and therefore do represent risk. This also implies that most vulnerabilities are likely to be not interesting to the attacker, and just counting vulnerabilities may overestimate actual cyber-attacks.

To visualize the potential issues arising with a large volume of irrelevant vulnerabilities, we present a Venn diagram in Figure 1 where size of the area is proportional to the number of vulnerabilities in each dataset and the color is an indication of the CVSS score (A detailed analysis of the CVSS scores will follow up in later sections).

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 75% of the time. As a minor note, NVD does not reference all vulnerabilities we found: the SYM and EDB datasets contain respectively 9 and 63 vulnerabilities that are not present in the NVD dataset. CVSS data on these

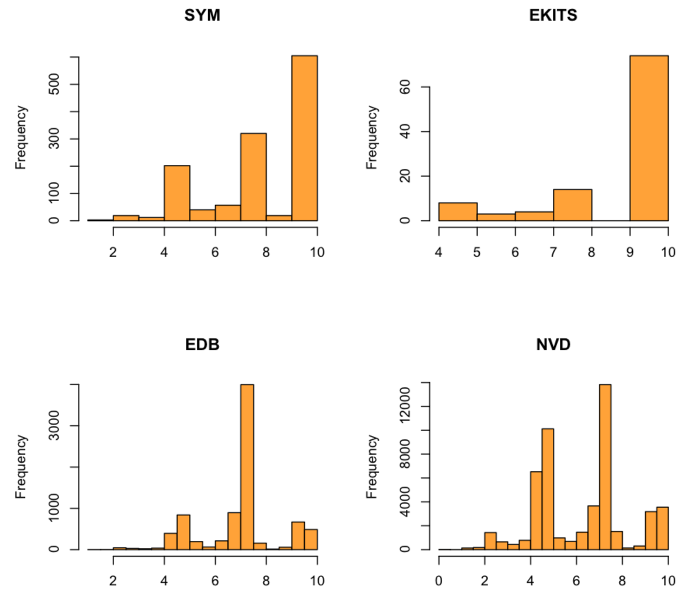


Fig. 2. Distribution of CVSS scores per dataset.

vulnerabilities is therefore missing.

A rushing conclusion might be that, if one sees a vulnerability affecting his/her software in the black market, there is roughly a 75% chance that it is exploited in the wild. The same cannot be said about EDB and NVD, for which the percentages is less than 5%. A possible counter observation would be that EDB and NVD include many low impact vulnerabilities and better results could be obtained if we eliminated the vulnerabilities with little chances of being exploited.

To address the above observation we further analyse the CVSS score and report the histogram distribution in Figure 2. It is definitely not normal across all datasets. There are essentially three clusters of vulnerabilities throughout all our datasets, with the 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 III reports the number of vulnerabilities with HIGH, MEDIUM, LOW score per each dataset. 52% of vulnerabilities

TABLE III
INCIDENCE OF CVSS SCORES PER DATASET

CVSS Score	EKITS	SYM	EDB	NVD
HIGH	74	612	1.209	7.026
MEDIUM	19	393	5.324	20.858
LOW	10	272	1.589	21.715
tot	**103**	**1.277**	**8.122**	**49.599**

TABLE IV
OBSERVATIONAL SPECIFICITY AND SENSITIVITY OF EACH DATASET.

test(v.CVSS) = H v M — SYM	EKITS	EDB	NVD
Sensitivity	97.4%	94.4%	78.7%
Specificity	32.0%	20.3%	44.4%

Sensitivity is the probability of the CVSS score being medium or high for vulnerabilities actually exploited in the wild. Specificity is the probability of the CVSS score being low for vulnerability not actually exploited in the wild.

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. The NVD totals do not coincide with Table I because 25 entries do not report CVSS score. MEDIUM and HIGH score vulnerabilities look interesting for exploitation.

Two issues hinders general conclusions: (a) HIGH, MEDIUM or LOW CVSS scores may not characterize correctly the vulnerabilities in SYM. (b) these results 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. To address (a) we look at two additional metrics, namely *sensitivity* and *specificity* (§IV). As for (b), we further explore the CVSS *sub*scores of vulnerabilities to underline statistically significant peculiarities of vulnerabilities in SYM (§V) and use these as control variables to random sample from EKITS, EDB, and NVD(§VII).

IV. SENSITIVITY AND SPECIFICITY

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 (the CVSS score) 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 according to the risk associated with the vulnerability[20].

Following the preliminary analysis in Section III we consider MEDIUM and HIGH CVSS scores as positive tests while LOW scores are negative tests. In formulae, Sensitivity=$Pr(v.score \geq 6 \mid v \in \mathsf{S}YM)$ while Specificity= $Pr(v.score < 6 \mid v \notin \mathsf{S}YM)$. Table IV reports the observational specificity and sensitivity for each dataset.

For the CVSS score to be a good indicator within a dataset, sensitivity and specificity should be both high, possibly over 90%. As shown in Table IV, EKITS is the only dataset that perform well in terms of sensitivity: out of 100 vulnerability exploited in the wild 97 would be predicted to be dangerous (H or M CVSS score). For NVD, an HIGH or MEDIUM CVSS score is *not* a good indicator that an exploit will actually show off in the wild: 22 vulnerabilities out of 100 which are actually dangerous would fail to get the HIGH or MEDIUM score (78% sensitivity). EDB scores well in terms of sensitivity: having an proof-of-concept exploit (being in EDB) *and* a CVSS score is good test (only 3 dangerous vulnerabilities out of 100 would turn negative tests). Unfortunately, all databases have poor specificity: basically more than 1 vulnerability out 2 *not* dangerous vulnerabilities would be wrongly tagged with a HIGH or MEDIUM score. Loosely speaking, the CVSS test would generate a medical unnecessary panic among otherwise healthy individuals.

This conclusion is only based on observational data: we report *all* data without random sampling. Therefore, these results should be used to draw statistical conclusions with care. We will build a case-controlled experiment in a later section.

TABLE V
POSSIBLE VALUES FOR THE EXPLOITABILITY AND IMPACT SUBSCORES.

Exploitability subscore		
Access Vector	Access complexity	Authentication
Undefined	Undefined	Undefined
Local	High	Multiple
Adjacent Net.	Medium	Single
Network	Low	None
Impact subscore		
Confidentiality	Integrity	Availability
Undefined	Undefined	Undefined
None	None	None
Partial	Partial	Partial
Complete	Complete	Complete

V. THE IMPACT AND EXPLOITABILITY SUBSCORES

The general CVSS score takes into consideration two subscores: *Impact* and *Exploitability*. The former is a measure of the potential damage that the exploitation of the vulnerability could cause to the victim system; the latter attempts at measuring the likelihood-to-be-exploited of the vulnerability [3]. They are calculated on the basis of further variables that are reported in Table V. Values of each column can be combined with values of the other columns in any possible way.

The impact metric distribution is plotted in Figure 3. Somewhat surprisingly, high impact score vulnerabilities are not by default preferred by attackers: data from SYM shows that attackers are also mildly interested in "low impact" vulnerabilities (256 - 20%) beside "high-impact" ones (663 - 50%). This effect is much reduced for the EKITS dataset: only 8 vulnerability (1 of them actually exploited in SYM) scores LOW (less than 8%). The HIGH or MEDIUM Impact score might therefore be a co-variate for the presence of an exploit in the market. As for EDB and NVD, the picture change completely: the greatest majority of vulnerabilities in EDB (5245, or 65%) have a medium score, and the remaining 35% is equally split between HIGH and LOW

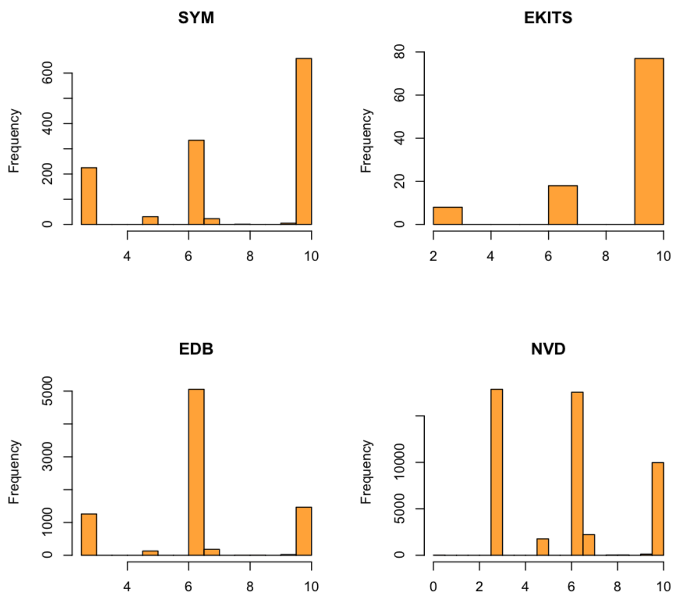


Fig. 3. Distribution of CVSS Impact subscores per dataset.

TABLE VI
INCIDENCE OF VALUES OF CIA TRIAD WITHIN THE SYM DATASET.

Confidentiality	Integrity	Availability	SYM	Negligible
C	C	C	51.53%	
C	C	N	0.08%	✓
C	N	C	0.08%	✓
C	N	N	0.23%	✓
P	P	P	26.16%	
P	P	N	1.64%	✓
P	N	P	0.16%	✓
P	N	N	7.67%	
N	C	C	0.23%	✓
N	P	C	0.08%	
N	P	P	0.63%	
N	P	N	3.68%	
N	N	C	1.57%	✓
N	N	P	6.26%	

Impact vulnerabilities. This might explain the low specificity for EDB: too many harmless vulnerabilities which just have a proof-of-concept exploit get a MEDIUM score. In NVD, the universe of vulnerabilities, only 20% (10101) have HIGH Impact score, while 40% (19.847) are scored MEDIUM. The remaining 19.651 are scored LOW.

The classification in Confidentiality, Integrity and Availability is a legacy of the classical view of security. Table VI shows the percentages of values assumed by three variables in the SYM dataset. Negligible configurations are represented by handful of vulnerabilities (e.g. the CCN case is represented by 1 vulnerability). It shows that the Availability variable almost always assume the same value as Integrity, apart from the case where both Integrity and Confidentiality are set to "None". The average variation of the Impact score if Availability was not to be considered at all is less than 1%. This is unsurprising: more convenient and reliable ways exist to perform a Denial-of-Service attack than mounting a remote exploit. Botnets led to the extinction of the "ping-of-death".

TABLE VII
COMBINATIONS OF CONFIDENTIALITY AND INTEGRITY VALUES PER DATASET.

Confidentiality	Integrity	SYM	EKITS	EDB	NVD
C	C	51.61%	74.76%	18.11%	20.19%
C	P	0.00%	0.00%	0.02%	0.04%
C	N	0.31%	0.97%	0.71%	0.88%
P	C	0.00%	0.00%	0.01%	0.01%
P	P	27.80%	16.50%	63.52%	37.84%
P	N	7.83%	0.97%	5.61%	10.62%
N	C	0.23%	0.00%	0.18%	0.22%
N	P	4.39%	2.91%	5.07%	16.52%
N	N	7.83%	3.88%	6.75%	13.69%

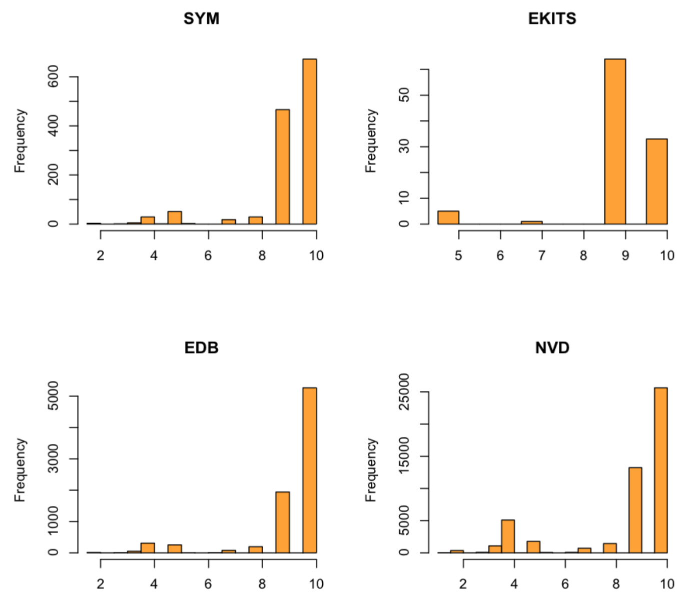


Fig. 4. Distribution of CVSS Exploitability subscores.

We proceed by analyzing the two remaining variables for the Impact subscore among all our four datasets. Results are reported in Table VII. Most vulnerabilities in the NVD dataset score "partial" in the three Impact sub-metrics. This effect is enhanced in the EDB dataset, where close to 70% of vulnerabilities score partial in at least one of either Confidentiality, Integrity or Availability. The scenario changes completely when looking at the SYM and EKITS datasets: most vulnerabilities (50%, 75%) score "complete" in the subscores. Across all databases mixed values (e.g. Confidentiality = Partial, Integrity = None) are of minor importance and do not evidence any intuitive trend.

Figure 4 shows the distribution of the Exploitability subscore per each dataset. This subscore has traditionally been used to represent 'exploitation-likelihood' [3]. Numbers are qualitatively identical among all datasets: most vulnerabilities have MEDIUM or HIGH Exploitability subscore, and almost none has LOW exploitability. Almost half of SYM entries (605) and two thirds (70) of EKITS's entries have an Exploitability subscore strictly lower than HIGH, while LOW

TABLE VIII
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%

TABLE IX
RELATIONSHIP BETWEEN ACCESS COMPLEXITY, IMPACT AND ACTUAL EXPLOITATION

		Impact	SYM	EKITS	EDB	NVD
Access Complexity	High	HIGH	1.33%	2.91%	0.58%	0.92%
		MEDIUM	1.88%	1.94%	2.34%	1.89%
		LOW	1.02%	0.00%	0.46%	1.89%
	Medium	HIGH	32.50%	55.34%	8.84%	7.65%
		MEDIUM	3.60%	4.85%	11.35%	7.69%
		LOW	2.43%	2.91%	5.29%	14.83%
	Low	HIGH	18.09%	16.50%	8.89%	11.80%
		MEDIUM	22.55%	10.68%	50.89%	30.43%
		LOW	16.60%	4.85%	11.36%	22.90%

scores vulnerabilities are a handful. On the other hand, 19.881 of the 40.574 non exploited vulnerabilities ($v \notin \mathsf{S}YM$ & $\notin$ $\mathsf{E}KITS$ & $\notin$ $\mathsf{E}DB$) are scored HIGH in the Exploitability submetric. These observations confirm Bozorgi et al.'s findings [3]: there is no direct relationship between Exploitability score and actual likelihood of exploitation. The EDB might still be a bad database, but the exploitability as a whole is a poorly discriminating score across all DBs.

Table VIII reports the total distribution of the exploitability variables. 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%). Authentication is another essentially boolean variable: most exploited vulnerabilities do not require any authentication.

VI. EXPLOITATION TRADE-OFFS

Among all subscores, access complexity present some interesting results: the percentage of "very difficult" vulnerabilities is equal (and very low) among all datasets but the percentage of "medium-complexity" vulnerabilities in the SYM and EKITS datasets is much higher than in EDB. Attacker are willing to put more effort in the process than security researchers! 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 capture easy vulnerabilities (71.14%).

To explain the higher average Complexity for vulnerabilities exploited in the wild we hypothesized a trade-off for the attacker: he/she is willing to put extra-effort in the exploitation only if it is worth it. Table IX reports the results of the analysis. The trade-off is particularly evident in the medium-complexity range of vulnerabilities: if an attacker is going to exploit a medium complexity vulnerability, most likely this will be a HIGH impact one (32.50%). This trend is even more evident in the EKITS dataset, in which this percentage increases to 55.34%. This supports the hypothesis that the extra effort required to write an exploit for a more complex vulnerability is to be weighted with a corresponding "return on investment". With LOW Complexity vulnerabilities, on the other hand, there is no clear difference between HIGH, MEDIUM and LOW impacts: as long as exploitation is easy, the attacker may be willing of exploiting it regardless of the Impact score.

In the SYM database, only 13 vulnerabilities (1%) exhibit HIGH complexity and LOW impact. They affect very popular software: Windows (2), Internet Explorer(6), MacOsX(2), Microsoft XML parser (1), Oracle and BEA enterprise software (2). This suggests that global market share or number of installations could be an interesting variable to add to those considered in the CVSS score.

VII. RANDOMIZED CASE-CONTROLLED STUDY

In order to obtain stronger statistical results we have generated a case-controlled study where the cases are the vulnerabilities in the SYM (loosely corresponding to cases of lung cancer), while the NVD, EDB, and EKITS correspond to patients from various sources and medical conditions. We are looking for a control variable (like smoking) that could overwhelmingly explain cancer. Our control variables for the generation of the samples are *access vector, authentication, access complexity, confidentiality, integrity, availability*. So we generated a random sample of vulnerabilities from the EKITS, NVD and EDB datasets with the same distribution of control variables present in the SYM database. The sampling was performed with the statistical tool R-CRAN [21]. Eventually, the samples include: 580 vulnerabilities for EKITS', 1272 vulnerabilities for EDB' and 1274 for NVD'. 3 vulnerabilities with "acc.vector== 'adjacent' " have been excluded from the sampling because of too low incidence (see Table VIII).

Table X shows the data for each of the datasets where we consider as a (tentative) explanatory variable the value of the CVSS and as response variable the presence of the vulnerability in the wild (in SYM). In order to understand whether this data is statistically significant we have run Fisher's exact test (because data is not normal) for each of the datasets. The p-values are reported in Table X. We recall that the p value does not measure the strength of an effect or an association (it is up to us to see it in the data), but only the certainty that the effect that we see in the data is not due to chance. A p value less than 0.05 is considered statistically significant because there is less than 5% chances that the data could exhibit the distribution by chance.

All p-values show statistical significance but the NVD' is, in contrast to EKITS' and EDB', not far from the $p < 0.05$ mark. showing that the evidence for statistical difference in the

TABLE X
CASE-CONTROLLED CONDITIONAL PROBABILITY

EKITS'

	v in SYM	v not in SYM	p-value
CVSS High or Med.	354 (79.37%)	92 (20.63%)	$p < 2.2^{-16}$
CVSS Low	43 (32.09%)	91 (67.91%)	

EDB'

	v in SYM	v not in SYM	p-value
CVSS High or Med.	158(15.58%)	856 (84.42%)	$p < 3.108^{-14}$
CVSS Low	3 (1.09%)	271 (98.91%)	

NVD'

	v in SYM	v not in SYM	p-value
CVSS High or Med.	61 (6.01%)	954 (93.99%)	$p < 0.022$
CVSS Low	7 (2.55%)	268 (97.45%)	

Case-controlled distribution among dataset of CVSS scores (explanatory variable) vs actual exploit in the wild as reported by SYM (response variable).

TABLE XI
RELATIVE RISK FOR CVSS SCORE

$v \in SYM$ vs $v \notin SYM$	Pr(H+M) - Pr(L)	Pr(H+M) / Pr(L)
EKITS'	+46.3%	2.4x
EDB'	+14.5%	14.3x
NVD	+3.5%	2.3x

Relative risk (by difference or ratio of proabilities) for a vulnerability to be exploited depending on the CVSS score and the database.

distributions of the scores among exploited and non-exploited vulnerabilities, here, is less strong than for the other datasets.

In this case, the effect that we are interested in seeing is the ability of CVSS scores (combined with the database) to predict the actual exploit in the wild (i.e. present in SYM). Figure Table XI shows both the difference among the probabilities and the ratio among the probabilities. Either approach can be used to evaluate the strength of an association.

If we consider vulnerabilities with the characteristics typical of exploited ones, such as network accessible, no authentication, medium complexity and high impact (see §V and §VI) each row in the table tells us which are the chances that a vulnerability with a MEDIUM or HIGH CVSS score is actually exploited in the while vs one with LOW scores.

So for EKITS'we see that a HIGH-MEDIUM vulnerability has around +46% more chances of being exploited (difference) and more than 2.4 times the chances of being exploited than a vulnerability with LOW (ratio). Both methods tell that the ending up in the black market is a bad sign. For EDB', the evidence is less strong. We only have +14% more chances albeit the ratio is 14.3 times higher. The reason for this conflicting results is due to the low prevalence rate of exploited vulnerabilities in EDB. Many of them are not exploited, even after contriolling for SYM-like characteristics and this dominate the difference of probability. NVD' has even weaker association for the same reasons: we only have +3.5% increase in chances and a ratio of 2.3 times. If we look at ratios only then vulnerabilities with the characteristics typical of exploited ones (network accessible, no authentication, medium complexity and high impact) and HIGH-MEDIUM CVSS have a much higher chances to be actually exploited and we should therefore fix them.

The higher ratio of NVD' and EDB' determines new values

TABLE XII
CASE-CONTROLLED SPECIFICITY AND SENSITIVITY.

CVSS H v M — Exploit	EKITS	EDB'	NVD'
sensitivity	89.17%	98.14%	89.70%
specificity	49.73%	24.39%	22.22%

Case-controlled sensitivity and specificity of the CVSS score being medium or high and the vulnerability being actually exploited in the wild (i.e. in SYM). Data has been random sampled from EDB and NVD according SYM's distribution of values for CVSS subscores.

for the specificity and sensitivity of the CVSS score. With sampled populations, the sensitivity of EKITS' drops by 8 percent points, while EDB' and NVD''s increases by 5 and 11 points respectively. This result is interesting in particular with respect to EDB', for whom HIGH CVSS scores might be a good test for exploitation. Yet, the CVSS score has a dramatically poor specificity for all datasets. Sampling SYM-like characteristics does not help in scoring vulnerabilities as "non-dangerous" ones.

Given our results on case-controlled specificity and sensitivity of CVSS, we conclude that the CVSS score is not a reliable test for not-exploitation of vulnerabilities; different results among different datasets evidence that its reliability varies depending on the reference dataset. This conclusion shows strong statistical significance throughout all of our datasets.

VIII. ASSOCIATION RULES FOR EXPLOITATION

The fact that CVSS shows to be an unreliable test against exploitation might be twofold:

1) CVSS is intrinsically correct, but the weights on variables are misplaced and do not represent risk correctly.
2) CVSS represents interesting characteristics of the vulnerability, but it is not sufficient to represent actual risk.

To resolve this issue we looked for association rules that imply the presence of the vulnerability in the SYM dataset.

WEKA [8] is a tool for data categorization. It also features *association rule mining* functionalities. Similarly to the approach adopted by Shahzad et al. in [24], we feed the tool with our dataset to see if any rule leads with sufficient confidence to the exploitation of the vulnerability (i.e. being featured in SYM). We run the tool on all our datasets, including the values for *access vector, access complexity, authentication, confidentiality, integrity, availability, v∈EDB, v∈EKITS, v∈NVD*. Temporal, software and vendor related information are not included as they do not belong to the CVSS score evaluation (see final discussion). We looked for 95% confidence association rules. Among the top 1 Million rules produced by WEKA, none predicted the vulnerability being featured in SYM (*symantec=yes*).

We tried then a manual approach: given our observations from Section V we build a model to fit the CVSS score evaluation with the exploitation of the vulnerability in SYM. However, we were unsuccessful in fitting the score to SYM *while* preserving statistical validity of the results.

The association rules result with WEKA shows that the presence of a vulnerability in the SYM dataset cannot be assessed via the CVSS scores and *sub*scores with any statistical significance. We therefore conclude that, according to our data, the *CVSS score is not representative of actual exploitation*. This is in accordance with our previous results on specificity and sensitivity presented in Section IV.

IX. Discussion and implications

Vulnerability assessment and patching has traditionally been a matter of great discussion within the community [4], [23], [24]. Here we summarize the main implications from our study.

Implication #1. Vulnerabilities exploited in the wild show specific patterns in the CVSS subscores; these observations can help to improve the sensitivity and specificity of the CVSS score. Some conclusions are more absolute (exceptions counted on one's fingers), while others are only statistically significant (hence the adverb "usually"), with a $p$vale lower than $< 2.2E - 16$ for Fisher's exact test.

1) *Actually exploited vulnerabilities are remotely exploitable and do not require multiple authentication.* Despite SYM containing local threats, only 3% of vulnerabilities are assessed as "only locally exploitable". Vulnerabilities exploitable from an adjacent network are even less interesting. 4% of vulnerabilities require a single instance of authentication; none of them require multiple authentication.
2) *Availability impact is irrelevant.* The impact of more than 96% of vulnerabilities in SYM can still be accurately assessed without taking into consideration the value of Availability. Therefore, when looking at broader datasets such as EDB and NVD, Availability represent almost only noise.
3) *Confidentiality and Integrity losses usually go hand-in-hand.* The overwhelming majority of vulnerabilities in SYM have complete or partial losses for both Confidentiality and Integrity: other combinations are less likely to be exploited. Only one value should therefore be considered.
4) *"Exploits" in EDB are usually for easy vulnerabilities.* Proof-of-concept exploits released in the EDB are for easier vulnerabilities than those actually exploited by attackers.
5) *Medium-complexity vulnerabilities are usually interesting only if they come along with high impact.* Either most attackers find high or medium complexity vulnerabilities too difficult or they seek an easier/more damaging one. In contrast Low-complexity vulnerabilities are exploited uniformly among all impact scores.

These observations boost up the sensitivity metric for both NVD' and EDB': it shrinks down the volume of 'uninteresting' vulnerabilities to manage.

Implication #2. The CVSS score is not capable yet of representing risk of actual exploitation: we used WEKA to try to map the whole set of variables (and relative values) to the presence of the vulnerability in SYM, but were unsuccessful. Unsurprisingly a second, manual approach didn't help either.

- The CVSS score underlines interesting characteristics of exploited vulnerabilities. However
- it is not expressive enough to reliably represent exploitation. Other factors such as software popularity, presence of the exploit in the market and existence of easier vulnerabilities for that software are all 'contextual factors' that might be worth exploring in future work.

Implication #3. The black market can be a good source to assess which vulnerabilities will represent risk. Exploits for vulnerabilities traded in the black market significantly overlap with those recorded in the wild: if an exploit is traded in the underground economy, it is going to be deployed in the wild. Of course, this conclusion is to be taken with a grain of salt: black markets are obviously not reliable in nature and a better understanding of their underlying trade dynamics and fairness are needed. However, we believe this paper presents some interesting preliminary evidence on the importance of *blackhat economics* in risk assessment - and could possibly make a starting point for future work.

X. Threats to validity

We identify a number of threats to validity. [18].

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. As for NVD and EDB, 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.

Because of the unstructured dataset of the original SYM dataset, to build SYM we needed to take some preliminary steps. We couldn't be sure about whether the collected CVEs were relevant to the threat. To address this issue, 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. This is highly prone to error and deeply influenced by our expertise; however, it seems that all the CVEs reported in our sample are relevant to the entry or to a variation of it. To double-check our 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 and threat-explorer datasets to use. Attack and infection dynamics are not always straightforward, and network and host-based threats often overlap. However, in this case, we are interested in a general evaluation of risk. Moreover, Exploit Kits enforce a drive-by download attack mechanism, therefore they are related to both the network and local threat scenario. We therefore can safely rely on both the datasets for our analyses.

Due to the shady nature of the tools, the list of exploited CVEs in EKITS may be incomplete and/or incorrect. We don't know any straightforward way to address this issue; to mitigate the problem, we crossed-referenced entries with knowledge from the security research community and from our direct observation of the black markets. We are planning to physically test a sample of tools which CVEs are in our dataset to check whether our list is sound. Moreover, our list of Exploit Kits may not be representative of actually deployed Exploit Kits. To address that we rely on databases of malicious urls such as Clean MX [11] and technical reports[12][25].

Internal validity is an issue when comparing different datasets. When building our NVD' and EDB'sample datasets, we considered as control variables those of the CVSS subscores only. However, other features of the vulnerabilities might be important to consider to build proper samples. 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.

External validity is concerned with the applicability of our results to real-world scenarios. As our bottom-line, 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 sample of real-world scenarios. Yet, our conclusion cannot be generalized to the risk due to targeted attacks. Targeted attacks in the wild of a specific platform or system are less likely to generate an entry into a general anti-virus product, and therefore less likely to be represented in the SYM database.

XI. Related works

Many studies before ours analyzed and modeled trends in vulnerabilities. Among all, Frei et al. [6] were maybe the first to link the idea of life-cycle of a vulnerability to the patching process. Their dataset was a composition of NVD, OSVDB and 'FVDB' (Frei's Vulnerability DataBase, obtained from the examination of security advisories for patches). The life-cycle of a vulnerability includes discovery time, exploitation time and patching time. They showed that, according to their data, exploits are often quicker to arrive than patches are. They were the first to look, in particular, at the difference in time between time of first "exploit" and time of disclosure of the vulnerability. This work have recently been extended by Shahzad et al. [24], 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. The difference with our paper, here, is that we look at a *sample of actual attack data* (SYM) and underline differences in vulnerability characteristics with other datasets. Importantly, we showed that looking at EDB (or OSVDB) might not be representative of actual vulnerability exploitation. An analysis of the distribution of CVSS scores and subscores has been presented by Scarfone et al. in [22] and Gallon [7]. However, while including CVSS subscore analysis, their results are limited to data from NVD and do not provide any insight on vulnerability exploitation. In this sense, Bozorgi et al. [3] were probably the first in looking at CVSS subscores against exploitation. They showed that the "exploitability" metric, usually interpreted as "likelihood to exploit" did not match with data from EDB: their results were the first to show that the interpretation of CVSS metrics might not be entirely straightforward. We extended their first observation with a in-depth analysis of subscores and of actual exploitation data.

On a slightly different line of research are studies concerned with the discovery of vulnerabilities. In [4] Clark et. al. underlined the presence of a 'honeymoon effect' in the discovery of the first vulnerability for a software, that is related with the "familiarity" of the product. In other words, the more popular the software the smaller the gap between software release and first vulnerability disclosure. This supports our conclusion that other factors apart from the CVSS score should be considered when analyzing risk associated with vulnerabilities.

Other studies focused on the modeling of the vulnerability discovery processes. Foundational in this sense are the works of Alhazmi et al. [2] and Ozment's [17]. The former fits 6 vulnerability models to vulnerability data of four major operative systems, and shows that Alhazmi's 'S shaped' model is the one that performs the better. However, as previously underlined by Ozment [17], vulnerability models often rely on unsound assumptions such as the independence of vulnerability discoveries. Current vulnerability discovery models are indeed not general enough to represent trends for all software [12]. Moreover, vulnerability disclosure and discovery are complex processes [16], and can be influenced by {black/white}-hat community activities [4], [6] and economics [14].

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 [5], Herley et al. [9] have show that IRC 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 cannot in any way assess the quality of the good (e.g. the amount of credit available) beforehand. On a folkloristic note, IRC markets are well known, in the underground community, to be markets for "newbies" and wanna-be scammers.

In contrast, Savage et al. [15] 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: sellers do re-use the same ID, the transactions are moderated, and reputation systems are in

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

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

place and seem to work properly. These observations coincide with our direct exploration of the black markets. 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.

XII. Conclusion

In this paper we presented our four datasets of vulnerabilities (NVD), proof-of-concept exploits (EDB), exploits traded in the black market (EKITS), and exploits recorded in the wild(SYM). We showed that, in general, the CVSS score and its submetrics capture some interesting characteristics of the vulnerabilities whose exploits are recorded in the wild but it is not expressive enough to be used as a reliable test for exploitation (with both high sensitivity and high specificity). We also traced some preliminary, novel line between *attacks in the wild*, *exploits in the white market*, and *exploits traded in the black markets*.

Alas, the bottom-line answer to the question set out in the title of this paper is not entirely satisfactory. *You should surely worry in few cases:*

- your vulnerability is listed by an exploit kit in the black market and have a medium-high CVSS score;
- your vulnerability has a proof of concept exploit (eg in EDB), requires no authentication, can be exploited over the network and have medium complexity but high-impact (with a medium-high CVSS score).

Unfortunately, nor CVSS subscores, nor the existence of exploits, nor the trading on the black market offer a statistically sound test for ruling out the 98% of the cases, for which users at large shouldn't worry.

Also our study do not apply to targeted attacks against individual companies. The SYM dataset might not cover this unique, individual exploit and therefore their actually exploited vulnerabilities would be marked by us as not exploited. To the best of our knowledge, there is no public evidence available in order to analyze these cases.

A robust claim can instead be made for the databases subject of this study: *using NVD, EDB (or consequently OVSDB) to assess software exploits in the wild is the wrong thing to do*. Those databases can only used to assess the upper hand in the race between software vendors and so-called security researchers.

An extension of this work is scheduled in October 2012, when in collaboration with Symantec's WINE project[13] we will gather additional data on exploited vulnerabilities. Another line of research we are following deals with the *economics* of attacker: we are investigating whether the trends in the black markets can be used to better assess risk.

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. No statement in this paper should be interpreted as endorsed by Symantec.

[13]http://www.symantec.com/about/profile/universityresearch/sharing.jsp

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] S. Clark, S. Frei, M. Blaze, and J. Smith. Familiarity breeds contempt: the honeymoon effect and the role of legacy code in zero-day vulnerabilities. In *Proc. of ACSAC'10*, pages 251–260, 2010.
[5] 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.
[6] S. Frei, M. May, U. Fiedler, and B. Plattner. Large-scale vulnerability analysis. In *Proc. of LSAD'06*, pages 131–138. ACM, 2006.
[7] L. Gallon. Vulnerability discrimination using cvss framework. In *Proc. of NTMS'11*, pages 1–6, 2011.
[8] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten. The weka data mining software: an update. *SIGKDD Exploration Newslett.*, 11(1):10–18, 2009.
[9] 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.
[10] M. Howard, J. Pincus, and J. Wing. Measuring relative attack surfaces. *Comp. Sec. in the 21st Century*, pages 109–137, 2005.
[11] F. Massacci, S. Neuhaus, and V. Nguyen. After-life vulnerabilities: A study on firefox evolution, its vulnerabilities, and fixes. In *Proc. of ESSoS'11*, LNCS, pages 195–208, 2011.
[12] F. Massacci and V. Nguyen. An independent validation of vulnerability discovery models. In *Proc. of ASIACCS'12*, 2012.
[13] P. Mell and K. Scarfone. *A Complete Guide to the Common Vulnerability Scoring System Version 2.0*. CMU, 2007.
[14] C. Miller. The legitimate vulnerability market: Inside the secretive world of 0-day exploit sales. In *Proc. of WEIS'07*, 2007.
[15] M. Motoyama, D. McCoy, S. Savage, and G. M. Voelker. An analysis of underground forums. In *Proc. of IMC'11*, 2011.
[16] A. Ozment. The likelihood of vulnerability rediscovery and the social utility of vulnerability hunting. In *Proc. of WEIS'05*, 2005.
[17] A. Ozment. Improving vulnerability discovery models. In *Proc. of QoP'07*, pages 6–11, 2007.
[18] 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.
[19] N. Provos, D. McNamee, P. Mavrommatis, K. Wang, and N. Modadugu. The ghost in the browser analysis of web-based malware. In *Proc. of HOTBOTS'07*, pages 4–4, 2007.
[20] 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.
[21] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.
[22] K. Scarfone and P. Mell. An analysis of cvss version 2 vulnerability scoring. In *Proc. of ESEM'09*, pages 516–525, 2009.
[23] G. Schryen. Is open source security a myth? *Comm. ACM*, 54, 2011.
[24] 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.
[25] 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.
[26] 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.