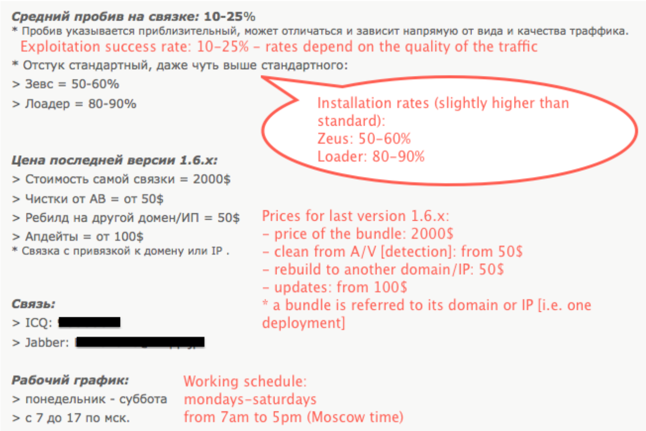
2013 IEEE Security and Privacy Workshops

Quantitative assessment of risk reduction with cybercrime black market monitoring.

Luca Allodi and Woohyun Shim and Fabio Massacci
DISI - University of Trento, Italy
name.surname@unitn.it

***Abstract*—Cybercrime is notoriously maintained and empowered by the underground economy, manifested in black markets. In such markets, attack tools and vulnerability exploits are constantly traded. In this paper, we focus on making a quantitative assessment of the risk of attacks coming from such markets, and investigating the expected reduction in overall attacks against final users if, for example, vulnerabilities traded in the black markets were all to be promptly patched. In order to conduct the analysis, we mainly use the data on (a) vulnerabilities bundled in 90+ attack tools traded in the black markets collected by us; (b) actual records of 9×10^7 attacks collected from Symantec's Data Sharing Programme WINE. Our results illustrate that black market vulnerabilities are an important source of risk for the population of users; we further show that vulnerability mitigation strategies based on black markets monitoring may outperform traditional strategies based on vulnerability CVSS scores by providing up to 20% more expected reduction in attacks.**

***Index Terms*—black markets; cybercime; vulnerabilities; exploits;**



Средний пробив на связке: 10-25%
* Пробив указывается приблизительный, может отличаться и зависит напрямую от вида и качества траффика.
Exploitation success rate: 10-25% - rates depend on the quality of the traffic
* Отстук стандартный, даже чуть выше стандартного:
> Зевс = 50-60%
> Лоадер = 80-90%

Installation rates (slightly higher than standard):
Zeus: 50-60%
Loader: 80-90%

Цена последней версии 1.6.x:
> Стоимость самой связки = 2000$
> Чистки от АВ = от 50$
> Ребилд на другой домен/ИП = 50$
> Апдейты = от 100$
* Связка с привязкой к домену или IP .

Prices for last version 1.6.x:
- price of the bundle: 2000$
- clean from A/V [detection]: from 50$
- rebuild to another domain/IP: 50$
- updates: from 100$
* a bundle is referred to its domain or IP [i.e. one deployment]

Связь:
> ICQ: [redacted]
> Jabber: [redacted]

Рабочий график:
> понедельник - суббота
> с 7 до 17 по мск.

Working schedule:
mondays-saturdays
from 7am to 5pm (Moscow time)

Fig. 1. Sample advertisment for a popular exploit kit in 2011- mid 2012, "Eleonore".

I. Introduction

Vulnerability exploitation is a major attack vector and threatens a relevant part of the population of internet users [19], [12], [3].

While vulnerability risk estimation is one of the biggest concerns in the software security community, there has been little success in developing quantitative estimation methods for vulnerability risk and effectiveness measures for risk reduction strategies. For example, although the Common Vulnerability Scoring System (CVSS) score [15][1] is used as a standard-de-facto risk metric for vulnerabilities and is widely recommended as a patch-prioritization metric in protocols, guidelines and best practices for vulnerability mitigation (e.g. U.S. Government recommended SCAP protocol [18]), it is not clear how well the CVSS score correlates with attack data [5], [1]. A complementary, or even new, approach might be required to tackle this issue.

This motivates us to conduct an analysis for this matter. According to Google, 70% of the threats to users are represented by automated web attacks [19]. As reported in security blogs[2], reports from the security industry (e.g., [22]) and academic publications (e.g., [12]), these infection mechanisms are mainly driven by ready-to-go attack tools traded in cybercrime black markets; these tools are often referenced as *exploit kits*. Exploit kits are, basically, websites deployed and maintained by an attacker (or somebody to whom he outsources the job [2]); when an unfortunate user connects to an exploit kit, it checks for vulnerabilities on the victim machine. If the user's system is vulnerable to any of the attacks the exploit kit supports, the vulnerability is exploited and *shellcode* is executed on the victim machine. At this point, the shellcode typically downloads a piece of malware chosen by the attacker and, if successful, infects the machine. These tools are advertised and traded in forum-like black markets. An example of such advertisement is given in Figure 1.

We believe that if, as Google reports [19], a relevant proportion of threats for final users comes from these tools, analyzing the risk coming from the cybercrime black markets against real attack data may provide useful insights into (a) estimating and mitigating risk for final users, and (b) developing effective defense strategies. To this purpose, we collected and analyze data on (1) vulnerabilities traded in the black markets and (2) real attack data as reported by Symantec's WINE Data Sharing Programme[3].

In this paper we give two closely related contributions:

1) We perform a preliminary analysis of attacks delivered by means of vulnerabilities traded in the black markets and bundled in exploit kits. In particular, we aim at understanding to what degree these vulnerabilities represent

[1] CVSS developed by NIST is a scoring system that intends to rate the relative severity of a vulnerability. It ranges from 0 to 10.

[2] For example, http://www.securelist.com/en/analysis/204792056/Drive_by_Downloads_The_Web_Under_Siege

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

DOI 10.1109/SPW.2013.16

a risk in the overall attack scenario. As a result, we show that black market vulnerabilities are responsible for a relevant fraction of overall attacks.

2) Motivated by these results, we hypothesize that the presence of a vulnerability in the black markets may be a good risk indicator for that vulnerability. We introduce the *effectiveness* [9] of a remediation strategy as a measure of "expected diminishment in attacks if a group of vulnerabilities were all to be patched". We then test our hypothesis and compare its performance with results for the current state-of-the-art approach, CVSS. As a result, we show that not only CVSS generally performs poorly as a metric for vulnerability remediation, but that monitoring the cybercrime black markets may result in up to 20% more effective patching policies.

This paper proceeds as follows. Section II describes our datasets and collection methodology. In Section III we report a first, observational analysis of attack trends and ratio of attacks driven by vulnerabilities traded in the black markets. We then introduce in Section IV the *effectiveness* metric as the expected reduction in overall attacks toward the end user after some mitigation strategy is enforced and test our data against it. In Section V we discuss the implications of our results. Section VI outlines possible threats to validity to our study, and Section VII concludes the paper.

II. Data

To explore the conjectures established in the introduction, we base our analysis on three different sources: EKITS, WINE-DB and NVD.

EKITS is our dataset of tools traded in the black markets. It contains detailed information on vulnerabilities bundled in exploit kits, services provided by the vendors (e.g. hosting on their own domain, discounted trail rates, etc), prices and release dates. The dataset is a substantial expansion on Contagio's Exploit Pack Table[4]. We retrieve the data in EKITS directly from various black markets. For the moment, the retrieving infrastructure is *semi*-automated, meaning that we manually verify the retrieved data before committing the update to the dataset. In order to lower our visibility in the markets, we are currently monitoring a limited set of communities. We periodically check our dataset with other external sources (e.g. security reports and security news press) to "fill the gaps" with data not reported in the communities we monitor. However, this circumstance proved to seldom occur. After more than 1.5 years of investigations and data collection we ended up monitoring more than 90 different exploit kits attacking overall 126 unique vulnerabilities. Data in EKITS spans from July 2007 (for Icepack Exploit kit) to February 2013 (when the exploit kit Whitehole was released).

The second source, WINE-DB is our ground truth dataset. It is a composition of publicly available data on attacked CVEs (SYM) and real attack data as collected by Symantec sensors worldwide and shared with researchers through the WINE data sharing programme. In the analysis, we use the association *attack signature - vulnerability* reported in SYM to map attack signatures recored in WINE to the vulnerabilities these attacks exploit.

Category	Type of software	Examples
1. BROWSER	Browser software	Internet Explorer, Firefox,
2. PLUGIN	Browser plugins	Acrobat reader, Adobe Flash Player
3. DEV	Software intended as support for developers	Visual C++
4. BUSS	Software used mainly in business environment	Lotus Notes, Dreamweaver
5. SERVER	Server side software	Apache, Ftp daemons
6. WINDOWS	Microsoft Windows releases	Windows XP, Windows Vista
7. OTH_OS	Operative systems other than Microsoft Windows	Solaris, OpenBSD
8. COMM	"Common-usage" software	Microsoft Office, Eudora

TABLE I

Categories for vulnerability classification. Categories are reported in descending order of specialization, meaning that a software falling in a lower-number category is excluded from any other category with a more general "specialization". For example, according to this rule Windows Server 2008 will fall in the SERVER category rather than the WINDOWS category.

1) Data in SYM is a collection of attack signatures and vulnerabilities reported as exploited in Symantec's Attack Signature and Threat Explorer public databases[5]. If a vulnerability is reported in SYM, this is evidence of the actual exploitation of the vulnerability in the wild. However, this dataset provides no information on volumes and time of the attacks.
2) The second data set, WINE-DB, fills this gap. By joining Symantec's WINE programme, we collected data on volumes of attacks per attack signature per month and attacks against different platforms (i.e. Windows {XP, Vista}). The collection of attacks in WINE-DB dates back to October 2009 up to November 2012. After the join with SYM, WINE-DB includes data on 9×10^7 attacks targeting more than 600 unique vulnerabilities.

Lastly, the third data set, NVD, refers to the data set collected from National Institute of Standards and Technology (NIST) National Vulnerability Database. NIST has been collecting information on vulnerabilities since 2004. We use this database as the population of the analysis in Section IV, since it contains all identified vulnerabilities that are plausible to be attacked.

A. Data categorization

NVD also reports information on the software that the vulnerability affects. We employ this information to categorize vulnerabilities into seven categories, as reported in Table I. Vulnerability software categorization is important here since it allow us to assess confounding influences on the probability of exploitation of different vulnerabilities. However,

[4] http://contagiodump.blogspot.it/

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

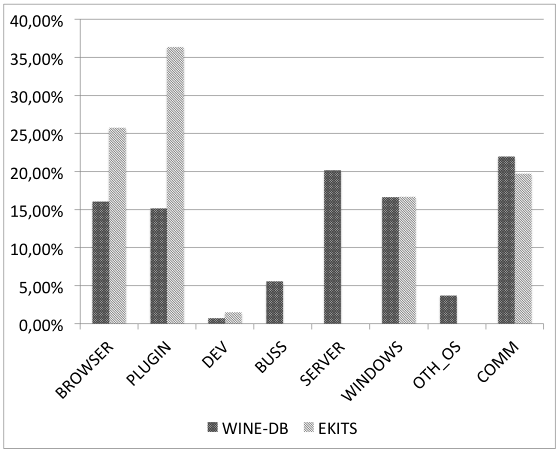


Fig. 2. Distribution of vulnerabilities by category.

the categorization task proved to be tough to accomplish: software data on NVD is not always consistently reported, meaning that often names for the same software are spelled differently across different entries or are incomprehensible (e.g. 3crwe554g72t). We therefore chose to use as a baseline for classification solely the vulnerabilities in SYM. While this is certainly a limitation to the overall classification of vulnerabilities, the diversity of software affected by vulnerabilities in SYM allowed us to categorize, on its basis, more than 40% of NVD[6]. With this approach we classified 95% of vulnerabilities in EKITS as well.

Figure 2 reports a bar diagram of vulnerability distribution for WINE-DB and EKITS by category. Vulnerability categories in WINE-DB are overall well distributed, evidencing the representativeness of the vulnerabilities it samples. However, we do not include all categories in our study. In particular, we exclude the categories that are not closely related to a "typical" home system. For example, measuring SERVER vulnerabilities would be unrealistic in assessing the security for the final regular user. We therefore exclude vulnerabilities classified in the categories $\{$BUSS, SERVER, OTH_OS$\}$. Moreover, due to the scarce prevalence of vulnerabilities in DEV among both WINE-DB and EKITS, we exclude that category as well. We end up with 542 vulnerabilities overall.

III. Observational analysis of attacks

In order to better understand the background and the trend of vulnerability exploitation, we start the analysis by conducting an observational exploration of the data. Vulnerability risk is typically assessed by means of the CVSS methodology. We therefore first look at the CVSS scores of vulnerabilities in our WINE-DB and EKITS datasets.

Figure 3 reports the ratios of total attacks driven by means of HIGH, MEDIUM and LOW score vulnerabilities. HIGH CVSS is identified by a score ≥ 9, $6 \leq MEDIUM < 9$ and

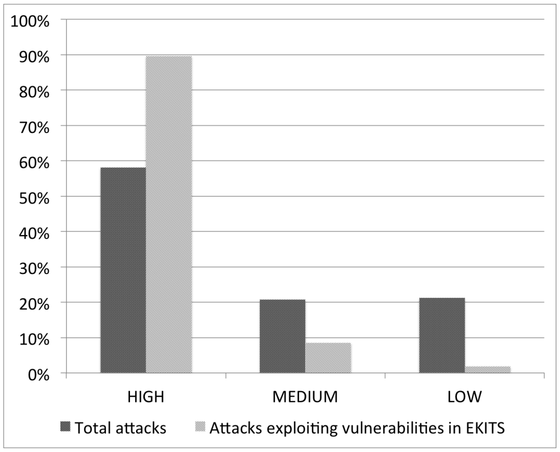


Fig. 3. CVSS score distribution expressed in ratio of delivered attacks per dataset. HIGH CVSS correspond to a score ≥ 9, $6 \leq MEDIUM < 9$, $LOW < 6$.

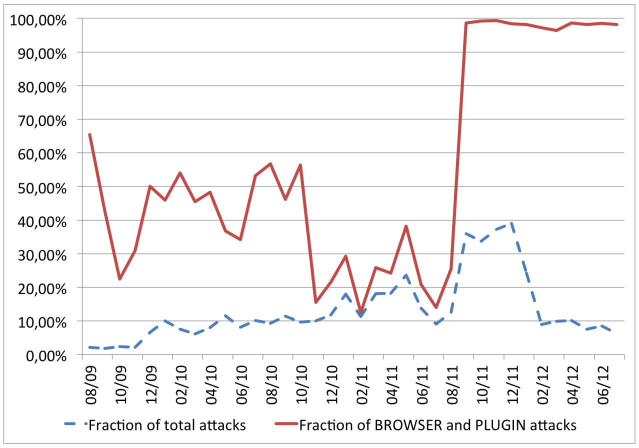


Fig. 4. Fraction of overall attacks driven by exploitation of vulnerabilities in EKITS. In september 2011 the prevalence of attacks targeting Browser and Plugin vulnerabilities in exploit kits peaked to almost 100% of the total.

$LOW < 6$. While the greatest majority of attacks are driven by means of HIGH score vulnerabilities in both WINE-DB and EKITS, MEDIUM and LOW score vulnerabilities still represent more than 40% of the volume of attacks delivered in the wild: this is further evidence that CVSS scores and vulnerability exploits do not correlate well [5], [1] ¿From this figure, it can be seen that CVSS might not be a good marker for actual exploitation, as opposed to what is often suggested by academic studies [21] and government reports [18].

The question raised from the observation of Figure 3 is then whether the observation of black markets can perform better as a marker for actual exploitation than the use of CVSS. Figure 4 is generated to explore this question. This figure illustrates the trends in the ratios of attacks targeting vulnerabilities traded in the black markets against overall attack volumes. The lower line can be interpreted as the conditional probability of being attacked by means of a vulnerability traded in the black markets ($Pr(v \in \text{E}KITS|Attacked)$). This trend peaks at

[6] Note that, this is far from being a full classification, the vulnerabilities classified in NVD are those representative (by construction) of the software reported in SYM and attacks are reported in WINE-DB.

about 40% of overall attacks. Remember that EKITS features 120 vulnerabilities, and WINE-DB runs at 540. Despite representing only 20% of all attacked vulnerabilities, vulnerabilities traded in the black markets are responsible for up to 40% of the final attacks for the user. We believe this evidences that black markets monitoring may be an effective way to avoid a big chunk of risk for the final user.[7] More generally, the trend of attacks driven by means of vulnerabilities in EKITS is monotonically positive in time. In other words, the prevalence of attacks against vulnerabilities in the black markets seem to be increasing. The drop in attack ratios in the last reported months of 2012 can be attributed to two factors: (a) Attackers became very good at avoiding signature detection by antivirus products; (b) Symantec may have needed additional time to update certain attacked CVEs in their signature descriptions. Not having any evidence to support (a), we consider the decreasing trend at the end of the time series an artifact of data censorship (right) for certain vulnerabilities [17].

Being web browsers and browser plugins typically the most exposed software to attacks for the final users [19], we further analyzed the incidence of attacks exploiting vulnerabilities in these categories. The red continuos line in Figure 4 depicts the trends in the ratio of attacks against BROWSER and PLUGIN vulnerabilities for the EKITS dataset. Overall, the greatest majority of attacks against these categories of software seem to be driven by means of black market vulnerabilities. After September 2011, these attacks are almost entirely targeting vulnerabilities in EKITS. This may explain the peak in overall attacks driven by black market vulnerabilities in September 2011.

To better understand these dynamics we further investigate attack trends per attack platform. Our WINE-DB dataset reports attacks against Windows Xp, Vista and Seven. This information is reported first-hand by Symantec sensors. We ignore the data on service packs and build numbers, as already proved to be unreliable [4]. Figure 5 reports ratios of attacks exploiting black market vulnerabilities per platform. Surprisingly, while attacks against Windows XP and Windows Vista remain under 10% of the total, Windows 7 seems to be the most targeted by exploit kits. Exploit kits have already been shown to choose whether to deliver the attack or not according to the system configuration of the victim machine [14]. This may therefore be evident that volume of attacks may increase despite higher security expectations from newer versions of operative systems.

Figure 6 plots the trends in difference of attacks driven by means of vulnerabilities in EKITS against vulnerabilities not in EKITS (WINE-DB / EKITS). Again, vulnerabilities in EKITS seem to always represent a higher risk for the users running Windows Vista or Windows 7 (despite Windows XP being the most popular platform detected in the WINE dataset [7]). Here, the negative values of the difference in the ratios represent periods of time for which the overall probability of being targeted by an attack *not* from the black markets is *higher* than one from the black markets. The positive values above the red horizontal line represent periods of time during which this probability is reversed: given that you were attacked in the period between August 2011 and January 2012 and was running Windows Vista or Windows 7, the attack was 10% more likely to be delivered by means of vulnerabilities traded in the black markets. This result in particular shows that black market monitoring could be an important proxy to measure final risk from cyber attacks for users.

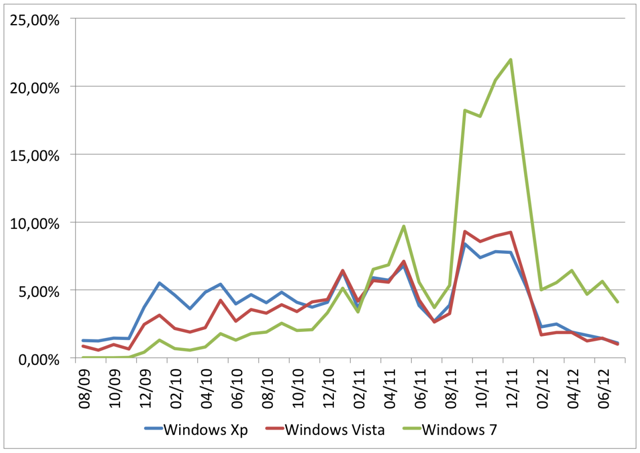


Fig. 5. Trends of attacks driven against vulnerabilities in EKITS as opposed to victim platforms.

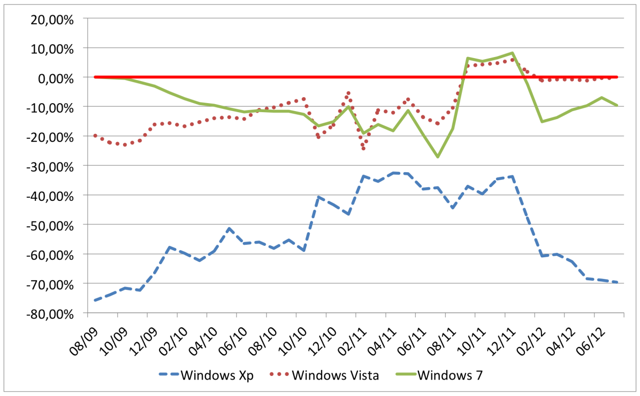


Fig. 6. Difference in the realitve ratios of attacks driven against vulnerabilities in EKITS ($Pr(v \in EKITS, Platform|Attack)$ and attacks driven by vulnerabilities not in EKITS ($Pr(v \notin EKITS, Platfom|Attack)$.

In conclusion, this preliminary analysis of our datasets underlines three aspects of attack trends that may be useful to better understand final risk for users:

1) vulnerabilities in the black markets are responsible for an important chunk of the overall volume of cyber-attacks affecting the final user.
2) higher security measures (e.g. those introduced in new versions of the operative system) do not necessarily discourage attackers from delivering attacks.

[7] As a speculation, we observe that September 2011 registers a peak in relative risk coming from the black markets. According to our data, this coincides with the release date of Blackhole 1.2.0, a major release of a very popular exploit kit [12].

3) depending on your system configuration and the moment in time, risk of attacks coming from the black markets can be up to 10% greater than risk coming from other sources.

These results show the high impact of the cybercrime black markets in terms of volumes of attacks against internet users. This is good reason, we believe, to ask the following question: "*If you were to patch all vulnerabilities which exploits are traded in the black markets, how much will your final risk of being attacked diminish?*"

IV. Effectiveness of patching a vulnerability

In order to answer the above question, this section conducts a more formal analysis for determining the effectiveness of different patching strategies in reducing the overall risk of attacks. The precise meaning of the effectiveness of non-patched vulnerability removal is based on asking how many of the vulnerabilities in the population would have been exploited in the wild if, instead of all being not patched, they had all been patched, all other pre-exploitation conditions remaining identical. In this paper, we run our analysis against two different strategies. Firstly, we test current best practices based on the prioritization of HIGH CVSS score vulnerabilities in the patching process [18]. Secondly, we apply the same approach to the case where, instead of CVSS score, the presence of a vulnerability in a market is the risk indicator for a vulnerability. Given the generality of the assessment presented in this work, we rely on two assumptions: 1) NVD includes all vulnerabilities that can be exploited; and 2) SYM includes all vulnerabilities that are actually exploited in the wild.

A. The Method

The method used in this section is based on a parallelism with various studies on estimating the "effectiveness" of seat belt usage in preventing fatalities in car crashes (e.g., [6], [20], [9], [8]). In our case, using a seat-belt is paraphrased by the installation of a patch, and the presence of a fatality is the actual exploitation of a vulnerability. For example, in [9], Evans concludes that, by using seat belts, chances of survival in a car accident increase by 43%; similarly, we want to assess to what degree chances of being attacked *decrease* if a certain group of vulnerabilities were to be patched. For the illustrative purpose, here, we present the detailed procedures following [6], [20], [9] and [8][8].

Let us assume that the probability that an exploit for a high risk vulnerability exists in the wild is $p_{h,e}$ and the probability that one exists for a low risk vulnerability is $p_{l,e}$. Then we can define the ratio of actual exploitation of high and low vulnerabilities, R, as:

$$R = \frac{\frac{\text{No. of low risk } v \text{ exploited in the wild (a)}}{\text{No. of low risk } v \text{ (b)}}}{\frac{\text{No. of high risk } v \text{ exploited in the wild (c)}}{\text{No. of high risk } v \text{ (d)}}} = p_{l,e}/p_{h,e}, \qquad (1)$$

where v denotes vulnerabilities[9]. R therefore gives the probability that low risk vulnerabilities are actually exploited compared to the corresponding probability that high risk vulnerabilities are actually exploited.

Subtracting equation (1) from $p_{h,e}/p_{h,e}$ and multiplying by 100 gives the "percent effectiveness", E. That is,

$$E = 100\left[\frac{(p_{h,e} - p_{l,e})}{p_{h,e}}\right] = 100(1 - R). \qquad (2)$$

More precisely, the effectiveness of patching vulnerabilities can be defined as the percent reduction in the expected level of actual exploitation in the wild that would occur if all considered non-patched vulnerabilities are patched, all other factors remaining same. Therefore, the effectiveness can be interpreted as the *percent of risk reduction* a final user gains when patching certain vulnerabilities of his/her system.

While the above application can provide an overall effectiveness, we calculate R for the different categories mentioned previously (i.e., categories for vulnerabilities) in order to take into account possible confounding effects in calculating the effectiveness estimates with respect to reducing the risk of exploitation. In particular, a confounding factor is a variable in the data that may influence the observed outcome (i.e. exploitation of a vulnerability) alongside the value of the *explanatory variable* (i.e. in our case, CVSS scores or $(v \in \mathsf{E}KITS) \in \{0, 1\}$). By "controlling" confounding variables it is therefore possible to isolate the explanatory variable and measure its real influence on the observed effect. Including a confounding factor in the analysis can provide us with a less biased estimate and with insights regarding the effectiveness for removing vulnerabilities from different categories. We now denote the ratio and the effectiveness for each category as R_x and E_x, respectively, where x is the category.

Since each calculation of R_x causes an error regarding the exploitation, we need to calculate the standard error of R_x, ΔR_x, which can be given by

$$\Delta R_x = R_x\sqrt{\sigma_\mu^2 + 1/a + 1/b + 1/c + 1/d}, \qquad (3)$$

where σ_μ is a value of the unpredictable fluctuations caused by confounding influences. Following [9] and [8], we assume that $\sigma_\mu = 0.1$; that is, due to unpredictable confounding interactions, the accuracy of the estimation of R_x is limited to $\pm 10\%$. The estimate of the effectiveness, when the standard error is taken into account, therefore yields $(E_x \pm 100\Delta R_x)\%$.

Since each R_x is a ratio, it is undesirable to compute the avarage value of R_x using the arithmetic mean calculation. Based on the corresponding estimates for different categories, we therefore compute the weighted average value, $\bar{R}$, expressed as

$$\bar{R} = exp\left[\sum_x (w_x \times log(R_x)) / \sum_x w_x\right], \qquad (4)$$

[8] We particularly apply the procedures used in [6] and [20] to estimate the effectiveness of patching vulnerabilities, and use the calculation used in [9] and [8] to take into account errors and weighting.

[9] (a), (b), (c) and (d) in equation (1) correspond to the followings in [6] and [20], respectively: the number of belted drivers injured, the number of car crashes with belted drivers, the number of unbelted drivers injured and the number of car crashes with unbelted drivers.

Remediation by	R	E	ΔR	$E \pm 100\Delta R$
CVSS	0.179	82.118	0.021	(82.12 ± 2.06)%
EKITS	0.032	96.801	0.006	(96.80 ± 0.58)%

TABLE II
ESTIMATED PATCHING EFFECTIVENESS WITHOUT CONFOUNDING INFLUENCES MEASURED BY CVSS

Category	R_x	E_x	ΔR_x	$E_x \pm 100\Delta R_x$
BROWSER	0.267	73.286	0.051	(73.29 ± 5.15)%
COMM	0.254	74.620	0.048	(74.62 ± 4.81)%
PLUGIN	0.411	58.852	0.104	(58.85 ± 10.35)%
WINDOWS	0.306	69.357	0.060	(69.36 ± 6.04)%
Weighted average values using CVSS:				(70.69 ± 2.99)%

TABLE III
ESTIMATED AVERAGE PATCHING EFFECTIVENESS MEASURED BY CVSS. BUSS, DEV, OTH_OS AND SERVER CATEGORIES ARE REMOVED FROM THE CALCULATION

Category	R_x	E_x	ΔR_x	$E_x \pm 100\Delta R_x$
BROWSER	0.061	93.935	0.020	(93.94 ± 2.00)%
COMM	0.096	90.393	0.036	(90.39 ± 3.55)%
PLUGIN	0.116	88.371	0.036	(88.37 ± 3.58)%
WINDOWS	0.104	89.598	0.051	(89.60 ± 5.09)%
Weighted average values using EKITS:				(90.96 ± 1.62)%

TABLE IV
ESTIMATED AVERAGE PATCHING EFFECTIVENESS MEASURED BY EKITS. BUSS, DEV, OTH_OS AND SERVER CATEGORIES ARE REMOVED FROM THE CALCULATION

where w_x is an assigned weight for each category and equals to $(R_x/\Delta R_x)^2$. Since $\bar{R}$ is also affected by confounding influences, the standard error of $\bar{R}$ is computed by

$$\Delta\bar{R} = \bar{R}/\sqrt{\sum(R_x/\Delta R_x)^2}. \qquad (5)$$

Therefore, the overall estimate of the effectiveness can be expressed as:

$$E = 100(1 - \bar{R} \pm \Delta\bar{R}). \qquad (6)$$

In the next section, we provide the results of the analysis using this application.

B. The Results

We fist estimate the effectiveness of fixing vulnerabilities on the reduction of expected attacks.In the analysis, we use CVSS and EKITS as proxies for "riskiness of the vulnerability". In particular, as for CVSS, we regard a vulnerability as high risk if it has $CVSS \geq 9$, while we consider a vulnerability as low risk if it has $CVSS < 9$. This is consistent with common practices in vulnerability prioritization [18]) and CVSS score distribution among vulnerabilities [1]. Similarly, we assume that a vulnerability included in EKITS is high risk whereas a vulnerability not included in EKITS is regarded as low risk.

Table II displays the results of the calculations. ¿From the estimated average patching effectiveness (82.12 ± 2.06 for the CVSS case and 96.80 ± 0.58 for the EKITS case) over 10% of difference in the patching effectiveness is identified. This indicates that a patching strategy based on CVSS might be at least 10% less effective in reducing attacks than a strategy based on the observation of the black markets. This is consistent with the results gained from Figure 3.

As explained previously, however, since confounding interactions can introduce serious biases in the estimations of the effectiveness, we further re-calculate our estimates using "software category" as a confounding factor. Tables III and IV report the results of such investigations. Each table provides the details of how category-specific final effectiveness is calculated. In the tables, each row reports the result for the corresponding category; the last row displays the average effectiveness of patching. More specifically, Table III indicates that patching effectiveness is highly dependent on categories. For example, when patching strategies are based on CVSS, the effectiveness for some categories such as BROWSER and COMM is higher than 70% while the effectiveness for PLUGIN is less than 60%. In addition, the results show that, if confounding interactions are considered, the estimated effectiveness is even lower than the effectiveness calculated without confounding influences. Two things should be noted here: first, the high value of the standard error in each category might limit the certainty of the estimated effectiveness. Second, the estimated weighted average of the effectiveness, $(70.69 \pm 2.99)\%$, indicates that the result in Table II might be biased.

As for the EKITS case, Table IV shows that overall around 90% of cyber attacks can be avoided by using a patching strategy based on the observation of black markets. While lower than the effectiveness measured without taking categories into account, we consider the estimated weighted average of the effectiveness, (90.96 ± 1.62) unbiased and more reliable compared to the result displayed in Table II. Moreover, the standard errors of the estimated effectiveness for the categories fluctuate less than those for the estimation with CVSS. This implies that using EKITS as a baseline for creating patching strategies can increase the certainty of the strategy effectiveness as compared to the case of CVSS.

V. DISCUSSION AND RELATED WORKS

Overall, our results highlight that much room for improvement in current approaches is available. In particular, we believe that our observations may influence vendors and users strategies when it comes to vulnerability remediation. Vulnerability patching is an expensive and high-uncertainty process, in which an accurate vulnerability risk measurement is key to a proper remediation strategy. Our results are here twofold: First, *CVSS score measures poorly when it comes to actual exploitation of HIGH score vulnerabilities*. Our results show that, overall, by following a high-score-first-patch policy the expected reduction in overall attacks against a final user is 70%. By considering BROWSER and PLUGIN vulnerabilities, which represent the most common vector of attacks for final users [19], the effectiveness of this strategy drops as low as 58%, with a ± 10% error margin.

Secondly, *measurement of risk by means of black markets monitoring may prove to be a good proxy for risk management and prioritization.* The effectiveness of a potential patching policy that considers the black markets for attacks as a proxy for "risky vulnerabilities" is expected to be 20% higher than the current CVSS policies. The margin of error allowed by this estimation is also very limited, meaning that no good reason to expect a much lower reduction in final overall attacks is found.

We can therefore conclude the followings:

1) Our empirical analysis confirms that patching strategies based on the observation of black markets can be much more effective than those based on the traditional CVSS score. Remediation strategies based on the CVSS score may represent a not cost-effective approach to vulnerability remediation.
2) The effectiveness estimates that ignore the confounding influences of the categories are biased upwards by large amounts particularly in the CVSS analysis; practically speaking, this means that a remediation strategy can be deeply affected by contextual variables. This observation is particularly useful for policy makers that need to build effective remediation strategies.
3) The high standard errors in the analysis using CVSS highlight the high uncertainty in the applicability of current remediation strategies to real-world scenarios.
4) In the analysis, it is confirmed that the estimates of the effectiveness for BROWSER and COMM categories are higher than other categories. If software vendors have very limited resources for patching, prioritizing the patching for vulnerabilities in BROWSER and COMM categories would provide the greatest reduction in final attacks for the users and may therefore result in a better investment.

A. Related works

Frei et al. [11] were maybe the first to thoroughly analyse vulnerability and exploitation dynamics. This work have recently been extended by Shahzad et al. [21], which included vendors and software in Frei's analysis. These studies relied on publicly available data on vulnerabilities and existence of public exploits (NVD and Exploit-DB or OSVDB databases). Only very recently, vulnerability studies using real attack data emerged [7], [4]. These studies looked at the vulnerability scenario in general, and did not focus on a particular source of threats such as exploit kits and black markets are in our study. Exploit kits relevance as a threat vector have been addressed (albeit only very recently) by Grier et. al in [12] and more generally by Provos et. al [19]. A comprehensive overlook of exploit kits was provided by Symantec in [22]. General characteristics of vulnerabilities in exploit kits and in the wild have already been analyzed by Allodi et al. [1], but no quantitative assessment of the actual volume of attacks these vulnerabilities drive has been reported in the literature so far. Black market analyses have been proposed mainly from a market-internal economic/observational point of view by Savage et. al [16], Franklin et. al [10] and Herley et. al [13]. Unlike these studies, we use our observations from the black markets to measure the effects of the black-hat economy in the ordinary everyday world.

VI. Threats to validity

In this Section we discuss the threats to validity of our study.

Construct validity regards the collection methodology of the data and the representativeness of the final dataset of the studied scenario. In our case, we collected data for vulnerabilities recorded in the wild and vulnerabilities traded in the black markets. WINE-DB is an aggregation of attack data recorded by one of the security industry leaders worldwide. The representativeness of the data depends on the host selection methodology adopted by Symantec, which states the sample is *representative* of the whole population. The representativeness of the EKITS dataset depends on (a) the representativeness of the black-hat communities we monitored and (b) our data collection mechanism. While it may be impossible to prove that we are monitoring the communities that report *all* the information we are interested into, we periodically check with third party resources such as security blogs and Contagio's Exploit Pack table and check for missing information in our dataset. However, we often end up having more details or even more exploit kits that those resources do, meaning that the collection mechanism is at least *on par* if not better than the industry public state-of-the-art.

Internal validity is concerned with the inter-relation between variables within the analyzed samples. In our case, as described in Section II, it must be underlined that we are *not directly* measuring exploitation against vulnerabilities, while rather the relevance of different attack signatures in the general attack scenario. Only subsequently we map attack signature data into vulnerability data. In a few cases, this means mapping the same measured volume of attacks against a signature to more than one vulnerability. In reality, it is therefore not necessarily true that each vulnerability has been attacked $\$volume$ times. In order not to introduce further noise to the analysis, we did not artificially modified these volumes (e.g. by assuming uniformity in the distribution of attacks per vulnerability per attack signature). While this multi-mapping problem only seldom occurs in our dataset, it must be identified as a potential source of noise for the final results.

External validity regards the applicability of the results to other scenarios. In particular, our study and our conclusions on patching policies apply *only* to the home-user threat scenario. Without further refinement, our conclusions cannot be applied to server / IT facility management or other general business environments. Because of the data collection methodology, the external validity of our conclusions may be limited by a number of factors. For example, volumes of attacks against vulnerabilities may change by geographical area (as an instance, exploit kits may attack fewer users in ex-USSR states[10]). Moreover, it is possible that some vulnerabilities attacked only

[10] http://krebsonsecurity.com/2012/01/citadel-trojan-touts-trouble-ticket-system/

in particular areas or affecting only particular systems of lower commercial interest for Symantec may not appear or are under-represented in our datasets. Further refinements in population control may therefore be needed to safely narrow the scope of our conclusions down to specific user populations.

VII. Conclusion

The scope of this paper is twofold. As a first contribution, we make a first exploratory analysis of volumes of attacks coming from vulnerabilities in the black markets. It must be noted that this analysis is *not* by itself evidence of the importance of the actual exploits and tools traded in the black markets: we do not have any proof that the actual attacks recorded in the WINE-DB dataset are delivered by means of exploit kits. Differently, our analysis provides, we believe, strong evidence that the black markets can be used as a *proxy* to estimate the final risk for the user: independently of the exploit delivery mechanism, a vulnerability in the black markets represents *a priori* important risk for a regular user. Our second contribution is the quantification of the performance of standard-de-facto approaches to vulnerability remediation against the potential of black-market monitoring. As a result, black markets monitoring results as a, on average, 20% more effective strategy than those currently enforced.

Acknowledgments

This work was partly supported by the projects EU-IST-NOE-NESSOS and EU-SEC-CP-SECONOMICS and TENACE PRIN Project (n. 20103P34XC) founded by the Italian Ministry of Education, University and Research. Our research wouldn't have been possible without Symantec's attack data sharing platform WINE. We also would like to thank Tudor Dumitras of Symantec research labs for the many helpful pointers. Interested researchers can reproduce our results by accessing the WINE dataset 2012-008.

References

[1] L. Allodi and F. Massacci. A preliminary analysis of vulnerability scores for attacks in wild. In *ACM Proc. of CCS BADGERS'12*, 2012.

[2] L. Allodi, W. Shim, and F. Massacci. Crime pays if you are only an average hacker. In *Proc. of IEEE ASE CyberSec'12*, 2012.

[3] W. Baker, M. Howard, A. Hutton, and C. D. Hylender. 2012 data breach investigation report. Technical report, Verizon, 2012.

[4] L. Bilge and T. Dumitras. Before we knew it: an empirical study of zero-day attacks in the real world. In *Proc. of CCS'12*, pages 833–844. ACM, 2012.

[5] 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*, July 2010.

[6] F. M. Council and W. W. Hunter. *Seat belt usage and benefits in North Carolina accidents*. University of North Carolina, Highway Safety Research Center, 1974.

[7] T. Dumitras and P. Efstathopoulos. Ask wine: are we safer today? evaluating operating system security through big data analysis. In *Proc. of LEET'12*, LEET'12, pages 11–11, 2012.

[8] L. Evans. Double pair comparison—a new method to determine how occupant characteristics affect fatality risk in traffic crashes. *Accident Analysis & Prevention*, 18(3):217–227, 1986.

[9] L. Evans. The effectiveness of safety belts in preventing fatalities. *Accident Anal. & Prev.*, 18(3):229–241, 1986.

[10] 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.

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

[12] C. Grier, L. Ballard, J. Caballero, N. Chachra, C. J. Dietrich, K. Levchenko, P. Mavrommatis, D. McCoy, A. Nappa, A. Pitsillidis, N. Provos, M. Z. Rafique, M. A. Rajab, C. Rossow, K. Thomas, V. Paxson, S. Savage, and G. M. Voelker. Manufacturing compromise: the emergence of exploit-as-a-service. In *Proc. of CCS'12*, pages 821–832. ACM, 2012.

[13] 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.

[14] V. Kotov and F. Massacci. Anatomy of exploit kits. preliminary analysis of exploit kits as software artefacts. In *Proc. of ESSoS 2013*, 2013.

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

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

[17] 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.

[18] 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.

[19] M. Rajab, L. Ballard, N. Jagpal, P. Mavrommatis, D. Nojiri, N. Provos, and L. Schmidt. Trends in circumventing web-malware detection. Technical report, Google, 2011.

[20] L. S. Robertson. Estimates of motor vehicle seat belt effectiveness and use: implications for occupant crash protection. *American Journal of Public Health*, 66(9):859–864, 1976.

[21] 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.

[22] 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.