Estimating the Assessment Difficulty of CVSS Environmental Metrics: An Experiment

Luca Allodi[2], Silvio Biagioni[3(✉)], Bruno Crispo[3], Katsiaryna Labunets[1], Fabio Massacci[3], and Wagner Santos[3]

[1] TU Delft, Delft, Netherlands
[2] TU Eindhoven, Eindhoven, Netherlands
[3] University of Trento, Trento, Italy
silvio.biagioni@unitn.it

Abstract. [**Context**] The CVSS framework provides several dimensions to score vulnerabilities. The environmental metrics allow security analysts to downgrade or upgrade vulnerability scores based on a company's computing environments and security requirements. [**Question**] How difficult is for a human assessor to change the CVSS environmental score due to changes in security requirements (let alone technical configurations) for PCI-DSS compliance for networks and systems vulnerabilities of different type? [**Results**] A controlled experiment with 29 MSc students shows that given a segmented network it is significantly more difficult to apply the CVSS scoring guidelines on security requirements with respect to a flat network layout, both before and after the network has been changed to meet the PCI-DSS security requirements. The network configuration also impact the correctness of vulnerabilities assessment at system level but not at application level. [**Contribution**] This paper is the first attempt to empirically investigate the guidelines for the CVSS environmental metrics. We discuss theoretical and practical key aspects needed to move forward vulnerability assessments for large scale systems.

Keywords: CVSS metrics · Vulnerabilities · Environmental · Risk level · Controlled experiment

1 Introduction

Vulnerability management is a process of primary importance in most organizations [18]. A central activity to most mandated security practices is that of vulnerability assessment where the Common Vulnerability Scoring System (CVSS in short) [3], the worldwide *de facto* standard for vulnerability assessment and management, plays a central role. For example, the PCI-DSS standard for systems, involved in cardholder data processing, mandates that any vulnerability with a CVSS score greater than four should be fixed [22]. Several other NIST standards for critical infrastructures play offer similar recommendations.

T.K. Dang et al. (Eds.): FDSE 2017, LNCS 10646, pp. 23–39, 2017.
https://doi.org/10.1007/978-3-319-70004-5_2

Problem Statement. The CVSS 3.0 [3] provide extensive guidelines to asses the environmental context in which systems and their vulnerabilities are placed so that the final severity of a vulnerability is upgraded or downgraded given the context. The application of Environmental CVSS guidelines to real-world networks is generally considered impractical, as it involves time-consuming assessments of network topology, system roles, and configurations [8]. For this reason, CVSS Environmental assessments are seldom employed in real-world scenarios, resulting in overall lower final security and compliance levels [18].

However, the CVSS standard also includes guidelines to evaluate the *security requirements* of the affected system to increase or decrease the severity score of a vulnerability. Security requirements should be far easier to identify than the technical interactions between different systems, and their identification is at the basis of most security management best practices [22,24].

For example, PCI-DSS demands that any system that does not directly or indirectly operate on cardholder data is physically or logically separated from any system that does it. This operation produces the 'Scope' of the PCI-DSS compliance [22] and it is therefore called 'scoping'. This requirement is a simple and natural one and should yield a precise score irrespective of network complexity or system type.

Approach. We conduct an experiment run with 29 students from a MSc Cyber Security Risk Assessment course at University X where we ask students to perform CVSS assessments on the security requirements of systems in two realistic scenarios derived from a practitioner's guide book [22]. To evaluate how well does the assessment scale, we compare students' accuracy (with respect to an expert assessment) between assessments in a 'flat' and a 'segmented' network where components are segmented away following PCI-DSS guidelines.

Key Findings and Contribution. The experimental results show that the application of security requirements using CVSS is negatively impacted by the complexity of the network, even for moderately complex networks such as those presented in a textbook treatment. This is particularly important as we were not considering the revision of the security assessment of specific technical configurations but only the revision of the high level requirements due to the network compliance to PCI-DSS guidelines.

In other words, the CVSS environmental metrics assessments in the segmented network scenario are characterized by significantly higher error rates than assessments performed in the flat network scenario. Further the error rates are significantly higher for system-type vulnerabilities than for application vulnerabilities. This underlines that the complexities arising from the interaction of an increasing number of systems (and their related functionalities) in an enterprise network play a relevant role in the assessment. Hence, the manual appraisal of security requirements may be significantly enhanced by the support of automated procedures that are currently not commonly deployed in practice [18], nor well defined in the literature [8].

Ours is the first study that highlights the discrepancy between standard approaches for environmental vulnerability assessments and their practical application. Following our findings, we discuss the theoretical and practical implications of future research in this direction that is, at the present stage, still missing.

2 Background on CVSS

The CVSS framework specification is the result of the work of a First.org SIG, composed of security and domain experts from industry and academia. The CVSS framework provides three groups of metrics:

The Base Metric Group aims at evaluating the technical characteristics of the vulnerability; these characteristics are intrinsic to the vulnerability.
The Temporal Metric Group measures temporary conditions that characterize a vulnerability. These conditions may reflect, for example, the existence of a patch for the vulnerability, or the 'maturity' or reliability of known, public exploitation code.
The Environmental Metric Group reflects the company-specific environmental conditions in which the affected software is deployed. This accounts for alternative controls in place that mitigate the capabilities of an attacker in reaching and exploiting the vulnerability, and other organizational characteristics (e.g., how critical the vulnerable system is to the business).

The most commonly used metrics are the *Base Metrics* as they describe a technical system on its own, and their collection is supported by a number of automatic tools. Table 1 provides a summary description of Base metrics used in this study, and their possible values [3].

They are also widely studied in the literature (e.g., [1,4,9,11,12,20] and further in [2,7,17,21,23]).

The Environmental Metrics are less frequently used in practice because they require additional analyses, performed by an assessor internal to the organization, of the network topology, the role of systems in the company, and the relevant business activities (e.g., whether the system is central to a core business process). The environmental metrics enable the administrator to adjust the 'baseline' CVSS score on the specific company environment by evaluating the Security Requirements of the affected system, and any Modified Base metric value. For example, a vulnerability in a non-critical system might get a lower score than a vulnerability in a critical system. Administrator could have more difficulty in correctly assessing a vulnerability because it may be difficult to automatically collect and analyse this environmental information (see Holm et al. [8] for a discussion). As environmental metrics are seldom used in the literature, we provide here a brief introduction.

Security Requirements sub-metrics specify the relevance of the security properties of the affected system with respect to the normal operation of the organization. For example, a system that manages sensitive customer data may have high security requirements regarding the Confidentiality and Integrity of the data,

Table 1. Summary of CVSS v3 base metrics

ID	Metric	Description	Values
AV	Attack Vector	Reflects how remote the attacker can be to deliver the attack against the vulnerable component. The more remote, the higher the score	Not Defined(X), Physical(P), Local(L), Adjacent(A), Network(N)
AC	Attack Complexity	Reflects the existence of conditions that are beyond the attacker's control for the attack to be successful	Not Defined(X), High, Low(L)
PR	Privileges Required	Reflects the privileges the attacker need have on the vulnerable system to exploit the vulnerable component	Not Defined(X), High(H), Low(L), None(N)
UI	User Interaction	Reflects the need for user interaction to deliver a successful attack	Not Defined(X), Required(R), None(N)
S	Scope	Reflects when the vulnerability can affect resources beyond the authorization privileges intended by the vulnerable component	Not Defined(X), Unchanged(U), Changed(C)
C	Confidentiality	Measures the impact to the confidentiality of information stored on the impacted system	Not Defined(X), None(N), Low(L), High(H)
I	Integrity	Measures the impact to the integrity of information stored on the impacted system	Not Defined(X), None(N), Low(L), High(H)
A	Availability	Measures the impact to the availability of the impacted component	Not Defined(X), None(N), Low(L), High(H)

Table 2. Summary of CVSS v3 security requirements metrics (This table describes the metrics for Confidentiality (C), Integrity (I) and Availability (A). It is taken from [3, p. 15]).

ID	Metric	Description	Values
CR	Confidentiality Requirement	Measures how loss of Confidentiality is likely to have catastrophic, serious or limited effect on the organization or individual associated with the organization	Not Defined(X), Low(L), Medium(M), High(H)
IR	Integrity Requirement	Measures how loss of Integrity is likely to have catastrophic, serious or limited effect on the organization or individual associated with the organization	Not Defined(X), Low(L), Medium(M), High(H)
AR	Availability Requirement	Measures how loss of Availability is likely to have catastrophic, serious or limited effect on the organization or individual associated with the organization	Not Defined(X), Low(L), Medium(M), High(H)

but Availability may be less of a concern. Similarly, a router that bridges two enterprise networks may have high requirements on Availability (of the router) and Integrity (of the routing table) than on Confidentiality. This is of central importance in the management of large infrastructures where the criticality of the system for the business operation is at the focus of a correct vulnerability management practice [22], and is the focus of this paper.

Modified Base sub-metrics allow the assessor to tailor the baseline CVSS assessment of the vulnerability to the specific deployment conditions of the system. For example if the component is deployed beyond a firewall that allows SSH traffic only from within the same subnet, the 'Modified' CVSS assessment for `Attack Vector` for an SSH vulnerability should receive a lower score.

Generally, each metric may go either down or up depending on the environment (e.g., poorer or better local configuration than default one, higher or lower security requirements etc.) or the requirements (see [3, p. 16]). Table 2 provides a list of possible values used for all three metrics of the CVSS v3 Security Requirements metrics used in this study.

PCI-DSS and CVSS Environmental Scoring. PCI-DSS is the reference standard to which organizations that implement card payments or money transfers must be compliant to [13]. The standard focuses on the security of the data of credit-card holders, i.e. typically the organization's customers.

Vulnerability management is a central activity of PCI-DSS [18], and is a function of a broader criteria for network segmentation called, as already mentioned, 'scoping'. PCI-DSS identifies 'critical' systems by considering the (possibility of) interaction between systems in the organization and sensitive data (e.g. cardholder information). A system is said to be 'in scope' for PCI-DSS compliance if it either directly manages sensitive data, or if it can interact over the network with a system that manages it. The primary goal of a correct implementation of PCI-DSS guidelines is therefore to 'segmentate' a network such that systems that do not strictly need to communicate with 'sensitive' systems are isolated from those systems, and are therefore 'out of scope' [22]. This operation is meant to drastically decrease the complexity of managing the security of the organization's systems, and is in general considered best practice for network management also outside of PCI-DSS recommendations [19].

Following these guidelines, the vulnerability management process in PCI-DSS involves the prioritization of vulnerabilities affecting critical systems over vulnerabilities affecting 'out of scope' systems. It becomes therefore especially important to be able to correctly identify the 'security requirements' of the system affected by the vulnerability with respect to the business operation [22, Ch. 9, 'Vulnerability Management in PCI', p. 151]. To this aim, PCI-DSS indicates CVSS as the prioritization metric of choice for vulnerability management, and the CVSS Environmental metric *security requirements* naturally matches PCI-DSS' specification of 'in scope' (i.e. high security requirements) and 'out of scope' (i.e. low security requirements) systems. In this study we therefore focus on the implementation of CVSS Environmental directives for the sub-metric

security requirements on two textbook PCI-DSS case studies provided by the authors of [22].

3 Research Design

Study Design and Planning. According to [3] the environment metrics enables security analysts to customize the CVSS score depending on the area where the component is located in the organization's infrastructure. To investigate this question we chose a within-subject design which requires the participants to score vulnerabilities for different environments, namely networks of different type (flat and segmented) and the presence of security countermeasures in form of compliance state ("before compliance" and "after compliance" with PCI Data Security Standard (PCI-DSS) [22]). To mitigate the learning effect we used different vulnerabilities and application scenarios.

Table 3 provides the list of features for a scenario and how the network evolve once it complies with PCI-DSS scoping guidelines. Each feature shown represents a component of the network that can be affected by two specific type of vulnerability: application (APP) and system (SYS). For example vulnerability CVE-2016-0036 of MS Windows Server is present on the scenario before the PCI-DSS compliance whilst vulnerability CVE-2016-1619 of Chrome is present on the scenario after compliance.

Table 3. List of features and corresponding vulnerabilities

Network	Feature	Software	CVE ID	Vuln. Type	CVSS Score
Initial Net	System Managing Customer Data	MS Office	CVE 2016 0126	APP	7.8
	Register to Shop's Mailing List	Chrome	CVE 2016 5167	APP	8.8
	Customers' Computers	MS Windows	CVE 2016 0019	SYS	8.1
	POS terminals for credit card and debit card transaction	POS Terminal System	CVE 2016 0067	SYS	7.8
PCI-DSS compliant Net	Corporate network appliances (web servers, bck servers, etc.)	MS Windows Server	CVE 2016 0036	SYS	8.8
	POS systems	POS Terminal System	CVE 2016 0469	SYS	5.5
	Administrative area in store	MS Office	CVE 2016 3234	APP	5.5
	Administrative area in store	Chrome	CVE 2016 6792	APP	9.8
	Wireless area with legacy and customer systems	MS Office	CVE 2016 0012	APP	4.3
	Wireless area with legacy and customer systems	Chrome	CVE 2016 1619	APP	7.6
	Core switch implementing ACLs on top of each VLAN	IOS Core Switch	CVE 2016 6441	SYS	9.8
	Core switch implementing ACLs on top of each VLAN	IOS Core Switch	CVE 2016 6428	SYS	7.8

Experimental Protocol. The experiments consist of three main phases:

Training phase: All participants attended the tutorial on the application scenarios describing the organization of the networks before and after compliance with PCI-DSS. The participants received an introduction into the CVSS Base and Environmental Metrics (metrics definitions, their impact on the score) and explanation how to complete task questionnaire.
Application phase: The participants were asked to assess the environmental metrics information on the security requirements (CR, IR, AR) for each of vulnerability in two scenarios (flat and segmented) before compliance, and revise their evaluation after the network has been changed to comply with the PCI-DSS standard. The participants had 2 h to complete the task. At the beginning of the task they received all necessary materials (e.g., tables and tutorial slides) in electronic format.
Evaluation phase: The participants' assessments for each vulnerability were validated by comparing it with the evaluation produced by an expert member of the CVSS SIG Scoring Group.

Variables and Hypotheses. The main objective of our study is to evaluate how the use of environmental metrics helps security analysts to score different types of vulnerabilities in different context. The *independent variables* are:

- the network type in the scenario ("flat" and "segmented"),
- the compliance of the scenario with PCI-DSS ("before" and "after"), and
- the type of vulnerabilities ("APPlication" and "SYStem" level).

The *dependent variable* is the correctness of the CVSS security requirements sub-metrics calculated, which are based on the participants' assessment of the related vulnerabilities score. Table 4 presents our hypotheses.

Table 4. Experimental hypotheses

	Null	Alternative
H1	The type of vulnerability does not affect the correctness of the environmental metric score	There is a difference in the correctness of the environmental metric score for different types of vulnerabilities
H2	The type of the network does not affect the correctness of the environmental metric score	There is a difference in the correctness of the environmental metric score for networks of different type

Data Collection and Analysis. To test our hypotheses we computed the vulnerabilities' score ([3, Sect. 8.3]) based on the environmental metrics assessed by the participants. We collected participants' assessments using an online questionnaire. The questionnaire was organized following the structure from [3, Sect. 1.1]. The detailed task is described in Sect. 4.

To assess the correctness of participants' evaluation of the vulnerabilities, we compared students' assessments with an expert assessment performed by one of the authoring members of the First.org Special Interest Group for CVSSv3. The expert's evaluation is our reference value against which to compare the participants' assessments. We only consider the presence of a change and its sign, i.e. direction of change. For each vulnerability and network type, we evaluate:

$$\begin{aligned} \Delta(p|v \cap cxty) = score(p|v \cap before \cap cxty) \\ - score(p|v \cap after \cap cxty) \end{aligned} \tag{1}$$

where Δ is the change $before$ and $after$ compliance of the network with PCI-DSS in the *score* of vulnerability v evaluated by the participant p, $cxty$ is the factor of the network type (flat/segmented). We then compare each participant assessment against the agreed-upon value identified by the expert. We can now evaluate our hypotheses by aggregating results by v and $size$.

To test our hypotheses we can use Fisher's exact test as our metric is binomial in nature ("correct" or "incorrect" in comparison with expert's evaluation).

To investigate the possible interaction of scenario and vulnerability types and co-factors (e.g. participants' background in security and working experience) on the correctness of participants' evaluation of the vulnerabilities, we use the permutation test for two-way ANOVA, which are suitable for not normally distributed samples.

4 Study Realization

Table 5 summarizes the demographics of the study. The experiment was conducted at the UniversityX in November 2016. The participants were 29 MSc students in Computer Science. The experiment took place in a single computer laboratory. The experiment was presented as a laboratory activity and only the high-level goal of the experiment was mentioned. The experimental hypotheses were not revealed so as not to influence the participants but they were informed about the procedure. The material used during the experiment is available online[1].

Application Scenario. To test the effectiveness of the CVSS guidance we considered two scenarios (flat and segmented networks) and how their environmental metrics should change after security metrics are deployed. The first scenario features four vulnerabilities and the second eight, reflecting the increased numerosity of the involved systems. In our study we used two scenarios described in the *"PCI Compliance: Understand and Implement Effective PCI Data Security Standard Compliance"* book [22].

First we provided participants with the flat network scenario then the segmented network scenario where the critical appliances are segregated from the public parts of the network (see Fig. 1).

[1] Removed for anonymity.

Table 5. Demographic statistics (The participants were 29 international MSc students attending Security Course at the UniversityX. More than half of the participants reported that they had a working experience which means that they may understand better the environments of different type).

Variable	Scale	Mean/Med.	Distribution
Age	Years	24.8 (mean)	31% were 21–22 yrs old; 41% were 23–25 yrs old; 14% were 26–34 yrs old
Gender	Sex		86% male; 14% female
Work experience	Years	3.4 (mean)	44% had no experience; 7% some experience; 28% had 1–3 yrs; 21% had 4–7 yrs
Expertise in security	0(Beginner)–4(Expert)	1 (median)	24.1% novices; 55.2% beginners; 10.3% competent users; 6.9% proficient users; 3.4% experts

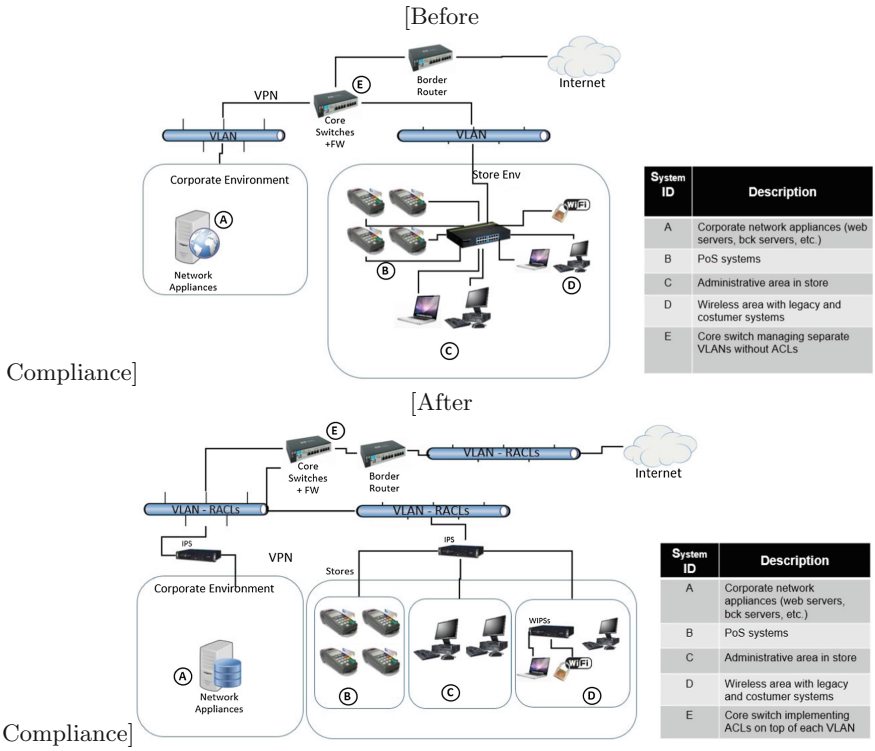


Fig. 1. Segmented network – before and after compliance with PCI-DSSs

Table 6. List of vulnerabilities in flat network

ID	Aff-Sw (NVD)	CVE ID	Description
A	MS Office	CVE 2016 0126	Microsoft Office 2013 SP1, 2013 RT SP1, and 2016 allows remote attackers to execute arbitrary code via a crafted Office document, aka "Microsoft Office Memory Corruption Vulnerability"
B	Chrome	CVE 2016 5167	Multiple unspecified vulnerabilities in GoogleChrome before 53.0.2785.89 on Windows and OS X and before 53.0.2785.92 on Linux allow attackers to cause a denial of service or possibly have other impact via unknown vectors

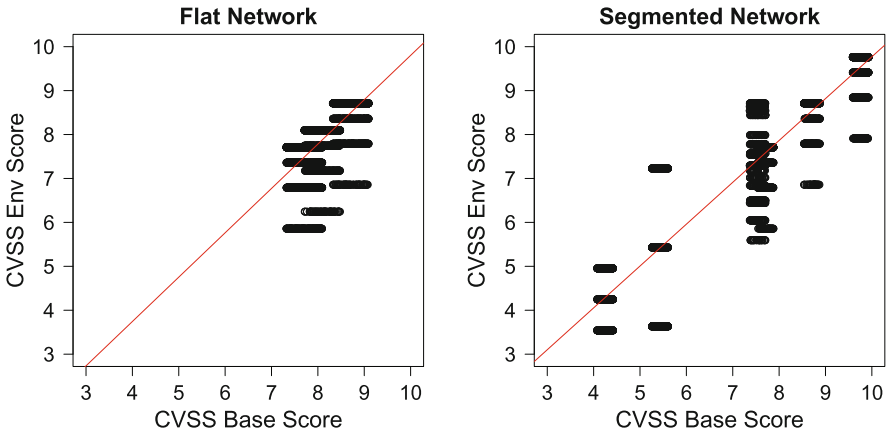
Below we present excerpts of descriptions of the segmented network scenario. Table 7 presents examples of vulnerabilities present in the segmented network.

> **Before:** "Christine's company has recently become a Level 1 merchant, it discovers that its internal assessors have underestimated the scope of PCI due to their flat corporate network. There are legacy system not involved in card processing on its corporate network, and many of those are no longer maintained and cannot meet PCI-DSS requirements. [· · ·]" [22, p. 77].
> **After:** "The cardholder environment will be segmented form the rest of the core network; IT and Management staff requiring access to those systems (both internally and remotely) are provided two-factor authentication tokens and have VPN software installed on their laptops; [· · ·]" [22, p. 77].

Table 7. List of vulnerabilities in segmented network

ID	Aff-Sw (NVD)	CVE ID	Description
A	MS Windows Server	CVE 2016 0036	The Remote Desktop Protocol (RDP) implementation in Microsoft Windows 7 SP1, Windows 8.1, Windows Server 2012 Gold and R2, and Windows 10 allows remote authenticated users to execute arbitrary code via crafted data, aka "Remote Desktop Protocol (RDP) Elevation of Privilege Vulnerability"
B	POS Terminal System	CVE 2016 0469	Unspecified vulnerability in the Oracle Retail MICROS C2 component in Oracle Retail Applications 9.89.0.0 allows local users to affect confidentiality via vectors related to POS

We ran Monte-Carlo simulation of 10000 times of CVSS environmental scores that the participants could obtain for the proposed vulnerabilities in flat and segmented networks. Figure 2 presents the distribution of the simulated scores (Table 6).



The distribution of CVSS environmental scores obtained by a Monte Carlo simulation of all possible values of security requirements is greater in the segmented scenario than in the flat one. This may increase the chances to score vulnerabilities incorrectly.

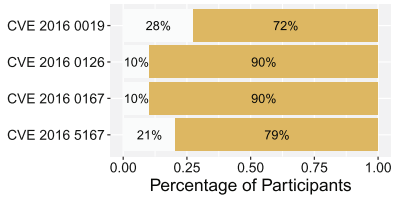
Fig. 2. Distribution of possible environmental CVSS scores for the scenarios

Task. The task included 4 vulnerabilities for the flat network scenario and 8 vulnerabilities for the segmented network scenario. The vulnerabilities were randomly selected from the CVSS 3.0. The participants were asked to assess, for each network scenario, the environmental metrics information on the security requirements (CR, IR, AR) for each vulnerability two times: for the company's environment *before* and *after* compliance with PCI-DSS.

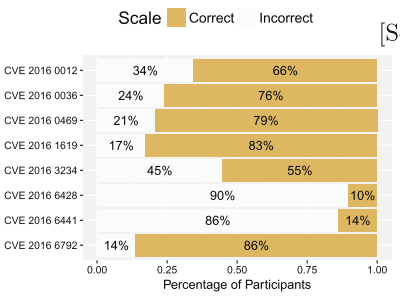
Table 2 reports the possible values for the security requirements sub-metrics. The questionnaire also allowed participants to justify their evaluation.

5 Our Results

Figure 3 presents the percentages of correct and incorrect responses given by participants for the flat network scenario. For all vulnerabilities we can see that majority of the participants correctly scored the vulnerabilities. In case of the segmented network (see Fig. 3) a significant part of participants were able to correctly assess the vulnerabilities. For *CVE 2016 0012* and *CVE 2016 3234* the participants were mostly correct, and only for vulnerabilities *CVE 2016 6428* and *CVE 2016 6441* the vast majority of the participants did a mistake.



[Flat Network]



[Segmented Network]

Most of the participants were able to score correctly the vulnerabilities in the flat network. In the segmented scenario we have a higher number of incorrect responses, e.g., CVE 2016 6428 CVE 2016 6441 vulnerabilities related to the system level were difficult to score.

Fig. 3. Responses by correctness and CVE

Vulnerability Type. Table 8 compares the number of correct and incorrect scores by vulnerability type correspondingly for flat and segmented networks. We can see that for flat network there is no difference between application and system vulnerabilities in the number of correct and incorrect responses (Fisher's test returned $p = 0.81$, $CI = [0.43; 3.82]$, odds ratio 1.3). In case of segmented network we can observe a significant difference between two vulnerability types by the number of correct scores. The results of Fisher's test confirmed the statistical significance of this difference ($p = 3.2 * 10^{-5}$, $CI = [1.81; 5.81]$, odds ratio 3.2). This means that for a segmented scenario it is easier to correctly score the application vulnerabilities than the system ones. Therefore, we can reject the null hypothesis $H1_0$ for the segmented scenario, but not for the flat one.

Network Type. There is a significant difference in the number of correct scores between networks in favor of the flat one (see Fig. 3). To investigate this difference, we summed the number of correct scores for each participant for flat and segmented network scenario. Table 9 reports the number of the correct and incorrect scores given by the participants by network type. The results of the Fisher's exact test confirmed that the number of the correct scores is differ by network type in favor of the flat network and this is statistically significant ($p = 4.2 * 10^{-6}$, $CI = [1.91, 6.18]$, odds ratio 3.4).

Hence, we can reject $H2_0$ as it is more difficult to score vulnerabilities correctly in case of segmented network. This can be explained by the fact there are

Table 8. Responses by correctness and vulnerability type (The difference in the number of correct scores between application and system vulnerabilities in flat network is negligible. In segmented network this difference is greater in favor of the application vulnerabilities).

[Flat Network]

	Correct	Incorrect	Total
APP	49	9	58
SYS	47	11	58
Total	96	20	116

[Segmented Network]

	Correct	Incorrect	Total
APP	84	32	116
SYS	52	64	116
Total	136	96	232

Table 9. Responses by correctness and network type (The participants were better in scoring vulnerabilities in the flat network (83% of responses were correct) over the segmented one (59% correct responses)).

Network	Correct	Incorrect	Total
Flat	96	20	116
Segmented	136	96	232
Total	232	116	348

more options how to score vulnerabilities in the segmented network than in the flat one and, therefore, the participants have more chances to do a mistake.

Co-factor Analysis. We investigated the possible interaction between the scenario and vulnerability type, and co-factors (e.g. participants' level of knowledge in security ad working experience) on the correctness of participants' scores of the vulnerabilities. The results of permutation test for two-way ANOVA did not reveal any statistically significant interaction of the main factors and co-factors on the experimental results.

6 Threats to Validity

Construct validity. The information received by the participants may affect the realism of the experiment in correctly representing an environmental vulnerability assessment. To mitigate this, we gave the participants vulnerabilities and system information following best-practice guidelines, as reported for example in ISO/IEC 30111:2013 and [22]. Further, the assignment of software and vulnerabilities to each system in the application scenarios may affect the validity of our experiment. Software was manually assigned to each system considering its function (e.g., backend server vs kiosk vs POS) as described in [22] and vulnerabilities were randomly picked from the set of existing vulnerabilities for that software reported in the National Vulnerability Database (NVD).

Internal validity. The interpretation of the CVSS metrics may depend not only on the assessor but also on received training. In our experiment, all participants received the same training class on CVSS ahead of the exercise, and received the same informative material during the experiment. To mitigate the relative effect of possible *learning effects* whereby later vulnerability assessments may be influenced by anterior evaluations, the system forced answering all questions in a fixed order for all participants.

External validity. To assure generalizability of our conclusions, we considered two real case-studies reported in the literature [22] and derived network schemas and system configurations from their description. Additionally, students may not suitably represent professionals for the evaluation of vulnerability environments. However, as indicated by [10] students (and in particular MSc students [16]) are suitable subjects when the research hypotheses evaluate relative differences between subjects as opposed to absolute assessments.

7 Discussion and Conclusions

Several standards, among which the PCI-DSS [13] and NIST's SCAP [15] are notable examples, mandate the adoption of CVSS guidelines to guide vulnerability risk mitigation. For example, the PCI-DSS standard uses the concept of 'segmentation' to identify the scope of the compliance within an organization's network: vulnerable systems within the compliance scope are more critical than identical vulnerable systems outside the scope. In this scenario, the CVSS Environmental metrics should provide a guideline for assessors to better fit the CVSS severity score to the organization's environment.

Current best practices, as well as expert opinion, emphasize the importance of considering additional information in the vulnerability assessment process on top of the baseline information provided by the vulnerability description [8,18,19]. Whereas it is obvious that the vulnerability assessment process can benefit from additional information on the vulnerable system, it is less clear whether current standard implementations result in better assessments when that information is added to the process. Human processing of information is known to significantly correlate with judgment errors (for example, [14] reports on the effects of 'information overload' on risk assessment practices in software engineering), but no account of the 'scalability' of existing security measurement best practices currently exists in the literature.

This study is the first to investigate the scalability of contextual security assessments on software vulnerabilities. Our results (see Table 10) indicate that a correct implementation of Environmental security requirement assessments may be impractical, even when limited to the sole evaluation of network segmentation and disregarding the complex technical interactions of different systems. This effect is particularly evident for systems vulnerabilities, which suggests that the operational inter-relation between systems is more difficult to capture using the CVSS specifications than for applications. For example, the network segmentation of a network appliance and the payment server may induce the assessor

to evaluate the operation of the two systems to be subject to different security requirements. Furthermore, the added complexity of the network naturally creates a greater variance in the scoring of the vulnerability (see Fig. 2), which may ultimately result in increased chances of error.

Table 10. Summary of results

	Null hypothesis	Result
H1	The type of vulnerability does not affect the correctness of the environmental metric score	**Rejected** for segmented network. The results showed that for the segmented network it is harder to correctly evaluate the changes due to security requirements in *system vulnerabilities* in comparison to *application vulnerabilities*
H2	The type of the network does not affect the correctness of the environmental metric score.	**Rejected**. The results showed that it is significantly easier to correctly score vulnerabilities based on security requirements in the context of *flat network* in comparison to the *segmented network*

Our experimental results indicate that contextual security assessments do not scale well with complexity of the environment even when limited to simple tasks involving the identification of 'segmented' areas of a network. This may lead to high error rates in the assessment, ultimately resulting in decreased overall security and compliance to regulation. This provides an independent confirmation of the previous claims based only on expert interviews that manual assessment of vulnerabilities and their relation with the overall infrastructure presents a significant challenge for the assessor [8].

These findings identify two key points in improvement of security assessment practices at the organizational level:

Automation: as the complexity of networks and IT infrastructure is not bound to decrease in the future, it seems natural to deduce that environmental assessments need some level of process automation. Whereas some tools exist in industry to help assessors in this direction (Rapid 7, Symantec, Qualys are only some of the more notorious vendors), the bulk of the environmental work needs to be performed manually by the assessor [18]. Current methodologies employed to integrate attack graphs with vulnerability information [5] could be integrated to consider environmental factors such as network dependencies, presence and configuration of mitigating controls, and the interaction between deployed systems as suggested by Zhuang and Aberer [25].

Data Integration: the variables included in the automation process are implicitly defined at the standard level. For example, CVSS' Security Requirements can be mapped to internal assessments performed in the Business Impact Analysis most companies perform [6]. Unfortunately, other measures may be harder to

consistently measure. For example, different security tools output event logs and alarms following different structures, which may render the correlation between a firewall configuration and an IDS alarm hard to infer. The NIST's SCAP standard [15] aims at achieving this standardization, but its large scale adoption is currently unclear.

References

1. Allodi, L., Massacci, F.: Comparing vulnerability severity and exploits using case-control studies. ACM Trans. Inf. Syst. Secur. **17**(1), 1:1–1:20 (2014)
2. Beck, A., Rass, S.: Decision-support by aggregation and flexible visualization of risk situations. In: Proceedings of ECCWS 2016, p. 313. Academic Conferences and Publishing Limited (2016)
3. CVSS-SIG. Common vulnerability scoring system v3.0: Specification document. Technical report (2015). First.org
4. Frei, S., May, M., Fiedler, U., Plattner, B.: Large-scale vulnerability analysis. In: Proceedings of LSAD 2006, pp. 131–138. ACM (2006)
5. Gallon, L., Bascou, J.J.: Using cvss in attack graphs. In: Proceedings of ARES 2011, pp. 59–66. IEEE (2011)
6. Giacalone, M., Mammoliti, R., Massacci, F., Paci, F., Perugino, R., Selli, C.: Security triage: a report of a lean security requirements methodology for cost-effective security analysis. In: Proceedings of ACM/IEE ESEM 2014, pp. 25–27 (2014)
7. Hamid, T., MacDermott, Á.: A methodology to develop dynamic cost-centric risk impact metrics. In: Proceedings of DeSE 2015, pp. 53–59. IEEE (2015)
8. Holm, H., Afridi, K.K.: An expert-based investigation of the common vulnerability scoring system. Comput. Secur. **53**, 18–30 (2015)
9. Holm, H., Ekstedt, M., Andersson, D.: Empirical analysis of system-level vulnerability metrics through actual attacks. IEEE Trans. Dependable Secur. Comput. **9**(6), 825–837 (2012)
10. Höst, M., Regnell, B., Wohlin, C.: Using students as subjects-a comparative study of students and professionals in lead-time impact assessment. Empir. Soft. Eng. **5**(3), 201–214 (2000)
11. Houmb, S.H., Franqueira, V.N., Engum, E.A.: Quantifying security risk level from cvss estimates of frequency and impact. J. Sys. Soft. **83**(9), 1622–1634 (2010)
12. Liu, Q., Zhang, Y., Kong, Y., Wu, Q.: Improving VRSS-based vulnerability prioritization using analytic hierarchy process. J. Sys. Soft. **85**(8), 1699–1708 (2012)
13. PCI. PCI (2010)
14. Pennington, R., Tuttle, B.: The effects of information overload on software project risk assessment. Decision Sci. **38**(3), 489–526 (2007)
15. Quinn, S.D., Scarfone, K.A., Barrett, M., Johnson, C.S.: SP 800–117: Guide to adopting and using the security content automation protocol (SCAP) version 1.0. Technical report, NIST (2010)
16. Runeson, P.: Using students as experiment subjects-an analysis on graduate and freshmen student data. In: Proceedings of EASE 2003, pp. 95–102 (2003)
17. Singh, U.K., Joshi, C.: Quantitative security risk evaluation using CVSS metrics by estimation of frequency and maturity of exploit. In: Proceedings of the WCECS 2016, vol. 1, pp. 19–21 (2016)
18. Verizon. PCI compliance report. Technical report, Verizon Enterprise (2015)

19. Wang, L., Zhang, M., Jajodia, S., Singhal, A., Albanese, M.: Modeling network diversity for evaluating the robustness of networks against zero-day attacks. In: Kutyłowski, M., Vaidya, J. (eds.) ESORICS 2014. LNCS, vol. 8713, pp. 494–511. Springer, Cham (2014). doi:10.1007/978-3-319-11212-1_28
20. Wang, R., Gao, L., Sun, Q., Sun, D.: An improved CVSS-based vulnerability scoring mechanism. In: Proceedings of MINES 2011, pp. 352–355. IEEE (2011)
21. Wen, T., Zhang, Y., Dong, Y., Yang, G.: A novel automatic severity vulnerability assessment framework. J. Commun. **10**(5) (2015)
22. Williams, B.R., Chuvakin, A.: PCI compliance: understand and implement effective PCI data security standard compliance. Syngress (2014)
23. Younis, A.A., Malaiya, Y.K.: Comparing and evaluating CVSS-based base metrics and microsoft rating system. In: Proceedings of QRS 2015, pp. 252–261. IEEE (2015)
24. Zhang, M., Wang, L., Jajodia, S., Singhal, A., Albanese, M.: Network diversity: a security metric for evaluating the resilience of networks against zero-day attacks. IEEE Trans. Inf. Forensics Secur. **11**(5), 1071–1086 (2016)
25. Zhuang, H., Aberer, K.: A non-intrusive and context-based vulnerability scoring framework for cloud services. arXiv preprint arXiv:1611.07383 (2016)