CONTEXTUAL REMEDIATIONS PRIORITIZATION SYSTEM DESIGNED TO IMPLEMENT THEORETICAL PRINCIPLES OF CVSS V4

Octavian GRIGORESCU[1], Laurențiu BOTEZATU[2], Andrei MUTU[3], Dinu ȚURCANU[4]

Technological advancements have led to an unprecedented expansion in the number and complexity of device interconnections. This brisk pace of adoption, while indicative of progress, often results in improper or incomplete configurations, thereby escalating the susceptibility of cyber infrastructures to vulnerabilities. In 2023 alone, approximately 30.000 vulnerabilities were identified, culminating in over 8 billion data breach records. Vulnerabilities prioritization is a problematic subject, consequently, VRM solutions should change their approach. This paper introduces a novel vulnerability prioritization method that proposes to cover new CVSS v4 version challenges. This approach leverages dynamic scoring and contextual information of vulnerabilities, aiming to provide a more nuanced and effective prioritization strategy.

Keywords: VRM, CVE, CVSSv4, Vulnerability Prioritization

1. **Introduction**

The evolution of computer networks has led to a necessity for cybersecurity to fight against the increasing number of intrusions, attacks, and disruptions. Surveillance is an essential element of cybersecurity, as a continuous process of monitoring can expose suspicious activities. Network administrators analyze ongoing events and proactively allocate resources to prevent potential attacks. Nevertheless, a significant challenge is arising with the increasing number of devices within networks: monitoring becomes difficult to perform.

In the contemporary landscape, discussions about standalone devices or systems are already obsolete. A relevant example of an increasing number of

[1] PhD Student, Faculty of Computer Science, University “Politehnica” of Bucharest, Romania, e-mail: octavian.grigorescu@upb.ro

[2] Engineer, Faculty of Computer Science, University “Politehnica” of Bucharest, Romania, e-mail: laur.botezatu@codaintelligence.com

[3] Engineer, Faculty of Computer Science, University “Politehnica” of Bucharest, Romania, e-mail: andrei.mutu@stud.acs.upb.ro

[4] Professor, Technical University of Moldova, Chișinău, Moldova, e-mail: dinu.turcanu@adm.utm.md

network components is the Internet of Things (IoT), a technological concept that describes a system of different devices with sensors connecting and exchanging data. As per a Statista report [1], the number of IoT devices almost doubled in 2023 compared to 2019. Despite the advantages and large number of industries where this technology has a positive impact, the growth of the IoT has raised concerns about security risks. A security report [2] indicates that many of these IoT systems exhibit medium or high-risk vulnerabilities that malicious actors could exploit. These vulnerabilities, combined with the rapidly increasing number of connected devices, have created the ideal environment for malicious cyber activities.

There are two known approaches for ensuring security: reactive and proactive, the first one aims to detect and block an ongoing attack, while the second one promptly identifies vulnerabilities that could be exploited. A study conducted by The Economist Intelligence Unit [3] on a two-year timespan has shown that companies using proactive security solutions have registered 53% fewer cyberattacks compared to those who do not. Network security auditing relies on human effort, being conducted through conformity assessments and penetration testing. However, growing networks and limited resources of a person introduce the need to use an automated solution to prevent these kinds of incidents. Another study [4] guided by PDQ, a software deployment and management platform, emphasizes the importance of proactive solutions that reduce the risk alongside reactive strategies such as disaster recovery.

Risk evaluation procedures, aligned with a proactive approach to ensure cyber security, require a scoring mechanism for vulnerabilities identified within an infrastructure. These mechanisms should take into consideration both external factors, such as patch availability or vulnerability prevalence, but also internal ones, including specific aspects of the vulnerabilities and targeted environment. Most of these considerations are already introduced in standard risk assessment procedures that use the Common Vulnerability Standard System (CVSS)[5].

An efficient risk evaluation process should take into consideration both the probability of exploiting a vulnerability and the potential impact this exploit could have on the given infrastructure. Common Vulnerability Scoring System handles impact estimation as an inherent vulnerability trait, computing scores based on intrinsic features such as attack complexity, necessary privileges, and potential damage caused to the vulnerable device. However, these values often neglect the contextual factors of the targeted device. For example, an attack conducted on a device with a low CVSS scored vulnerability could have a significant impact if the targeted device plays a key role within its infrastructure, outscoring the impact on an isolated device with a high CVSS scored vulnerability. Therefore, introducing device context in risk evaluation associated with vulnerability exploit becomes crucial, especially in the current landscape of interconnectivity exponential growth.

This solution proposes a scoring system that aims to address certain enduring issues and challenges that are still present in the new CVSS v4 proposal [6]. Our solution emphasizes the technical attributes and threat intelligence data that are present in the specification document.

The next section presents the state-of-the-art techniques for scoring systems, as well as similar approaches. Afterwards, the third section introduces our proposed solution. Following the discussion section, the paper concludes with ideas for further improvement.

2. State of The Art and Related Work

Vulnerability Risk Management (VRM) encompasses the systematic procedure of recognizing, evaluating, ranking, and reducing the impact of weaknesses present in the systems, networks, applications, or processes of an organization. Implementing this procedure is essential in safeguarding the integrity and security of the digital assets and infrastructure of an organization. VRM confronts the difficulties inherent in vulnerability management within the context of contemporary cybersecurity. Vulnerability management in contemporary cybersecurity is difficult due to the growing complexity of IT environments. Implementing continuous monitoring and risk-based prioritization of vulnerabilities are critical strategies. Effective vulnerability management requires that the IT and security teams conduct joint efforts. Automation is an indispensable component in optimizing operations and bolstering productivity. It is imperative for organizations to consistently modify their strategies in order to mitigate security risks within ever-changing threat environments [7].

The most recent iteration, CVSS v4.0, which was introduced by FIRST [8], is a substantial update that provides a scoring system that is more straightforward, adaptable, and precise. To rectify the shortcomings of its predecessor, CVSS v3.1, this iteration offers a more accurate depiction of potential hazards. CVSS v4.0 offers several enhancements in comparison to v3.1. In essence, version 4.0 provides an improved description of real-world risk, increased flexibility and adaptability, and enhanced clarity and simplicity. Furthermore, it incorporates novel notions like "Automatable," "Recovery," and "Mitigation Effort," which contribute to a more in-depth comprehension of every vulnerability [9].

Although CVSS 4.0 introduces several modifications and enhancements to the metric groups and resolves some persistent issues, obstacles continue to exist. Although CVSS may recommend using more than the base metrics for maximal efficacy, those are the ones that are ultimately populated in sources such as the NVD; downstream consumers are responsible for parsing the other metric groups. This is one of the most significant issues and the elephant in the room. It is widely acknowledged that relying solely on baseline metrics is inadequate for vulnerability prioritization; supplementary considerations include the business or environmental context, as well as the actual exploitation and exploitability of a given vulnerability.

As downstream CVSS consumers and organizations are responsible for populating a number of the metric categories with their own organization-specific context, a substantial amount of effort remains to develop precise vulnerability prioritization scoring and actions. This situation could be mitigated, even though research indicates numerous organizations are grappling with vulnerability backlogs amounting to hundreds of thousands and are unable to maintain pace with the exponential growth of discovered and disclosed vulnerabilities.

Regarding exploited vulnerabilities, sources have emerged, such as the US Cybersecurity and Infrastructure Security (CISA) Known Exploited Vulnerabilities catalog [10], which is related to federal agencies but effective for commercial organizations as well. Additionally, there is also the Exploit Prediction Scoring System (EPSS)[11], which is starting to get industry traction and is also headed by FIRST organization.

The CISA Known Exploited Vulnerabilities Catalog is an informational resource containing details regarding software vulnerabilities that have been identified as having been exploited by malicious entities. It provides organizations and individuals with a point of reference for identifying and prioritizing the patching or mitigation of vulnerabilities that present an imminent threat to their networks and systems [10].

The Exploit Prediction Scoring System (EPSS) is an initiative that utilizes data to forecast the probability or likelihood, that a real-world scenario will involve the exploitation of a software vulnerability. Our endeavor aims to assist network defenders in efficiently allocating resources towards remediating vulnerabilities. Although current industry standards serve as a valuable tool for identifying inherent attributes of vulnerabilities and delivering severity metrics, they are not entirely adequate for evaluating the true level of threat. By integrating real-world exploit data and up-to-date threat information from CVE, EPSS effectively bridges this knowledge gap. A probability score between 0 and 1 (or 0% to 100%) is generated by the EPSS model; higher scores indicate a larger likelihood that a vulnerability will be exploited [11].

The SSVC framework, an acronym for Stakeholder-Specific Vulnerability Categorization, operates under the premise that prioritization in the vulnerability management process is significantly influenced by decisions pertaining to vulnerabilities, as opposed to their severity. Utilizing decision trees to improve the decision-making process, particularly in vulnerability management, is the fundamental principle underlying SSVC. Various types of decision trees can be accommodated within the framework, contingent upon their function involved in the process. As an illustration, the SSVC framework incorporates decision trees to support patch appliers, patch deployers, and a patch coordinator decision tree, which was introduced with the release of version 2.0. Recognizing the variety of responsibilities associated with vulnerability management, this design is role-based, which is a benefit of the framework [12].

Jung et al. [13] present the Context-Aware Vulnerability Prioritization (CAVP) model as a revolutionary approach to tackle the growing intricacy and quantity of software security vulnerabilities. The CAVP model represents a notable progress in the domain of vulnerability management by incorporating a temporal aspect into the evaluation of Common Vulnerability Exposures (CVEs), an element that conventional models frequently fail to consider. By utilizing heuristic principles, this model automates the process of deriving temporal metric values of CVEs, enabling a context-sensitive and dynamic approach to vulnerability prioritization. As described by Jung et al., the implementation of CAVP in two organizations demonstrates its efficacy and applicability in augmenting the workflows of organizational risk management. The capacity of the model to visually represent vulnerabilities and rank them according to their significance in the particular setting of an organization is critical for effective cybersecurity administration.

Ahmadi Mehri et al. [14] tackle the pivotal issue of Vulnerability Risk Management (VRM) within the continuously evolving domain of cybersecurity in their groundbreaking research. The researchers' investigation, titled "Automated Context-Aware Vulnerability Risk Management for Patch Prioritization," presents a novel methodology for contextualizing and automating the VRM procedure with an emphasis on patch evaluation and prioritization. By tailoring patch prioritization to the specific context of an organization, this methodology enables the selection of vulnerability management modes and the adjustment of weights for evaluation criteria. In contrast to traditional tools such as Rudder's CVE-plugin [15], which provides a generic ranking of vulnerabilities, the Automated Context-Aware VRM (ACVRM) system developed by Ahmadi Mehri et al. prioritizes vulnerabilities according to the specific criteria of an organization, including its risk appetite. Additionally, it sorts vulnerabilities with identical patch scores according to their age, thereby granting priority to older, publicly known vulnerabilities. This methodology signifies a substantial advancement in the pursuit of enhancing the effectiveness, efficiency, and congruence of VRM with the unique security requirements and risk profile of an organization.

Our vision is to create a solution that is customized in an organization's context by allowing them to select certain parameters, with a focus on ease of use. We propose a pragmatic solution that implements the theoretical components from the CVSS v4 such as Threat and Environmental Group. Moreover, it goes on the same direction as the related work described above. Our solution has already a simplistic implementation that is fully operational, and in this paper, we present its extended version.

3. Proposed Solution

The proposed solution presents a method for prioritizing vulnerability remediation using several indicators. This approach constructs a contextual risk score and a remediation effort level that are calculated based on the following 3 components:

- static vulnerability indicators
- dynamic vulnerability indicators
- vulnerability context indicators

During our research and analysis, we used test data created in our laboratory followed by experiments that were conducted in association with the vulnerability assessment and management platform, CODA Footprint [16], in a much more advanced version. A relevant component in our experiments is the Collaborative Threat Remediation Workflows which is a vulnerability patches tracking system implemented in the platform. One of the features that is used the most in our work is the remediation learning module that offers insights about necessary efforts in certain contexts.

The input for the first two components used in our calculations is retrieved from the Yggdrasil Threat Intelligence service [17]. The vulnerability context indicators component is derived from two subcomponents: scanning results and user configuration. The scans are performed by the CODA intelligence system of agents that retrieves the requested data, running predominantly on the Windows operating system. Among collected data we are listing helpful data in our research such as: installed applications, IP addresses, running processes, list of open ports, installed patches, missing patches, connections, security products, and CIS Benchmarks[18] results run by a Security Content Automation Protocol(SCAP)[19] tool such as OpenSCAP [20].

Data collected by the agent system is used to determine existing vulnerabilities of the applications running on the devices, vulnerabilities generated by the lack of patches installed, the state of the system in terms of CIS Benchmarks (which is related to OpenSCAP) as well as the risk of the applications according to their state. Also, based on the data regarding the security products' status: whether installed and up to date or not, we establish a score.

In Figure 1 the components used for calculating the contextual risk scores and remediation efforts within an infrastructure are presented.

3.1. Static Vulnerability Indicators

The data used as input for Static Vulnerability Indicators component is taken from the Threat Intelligence service that detects early vulnerabilities [21]. Details about CVEs are taken from several sources, the severity of non-impact ones is predicted according to [22], and tactics and techniques will be

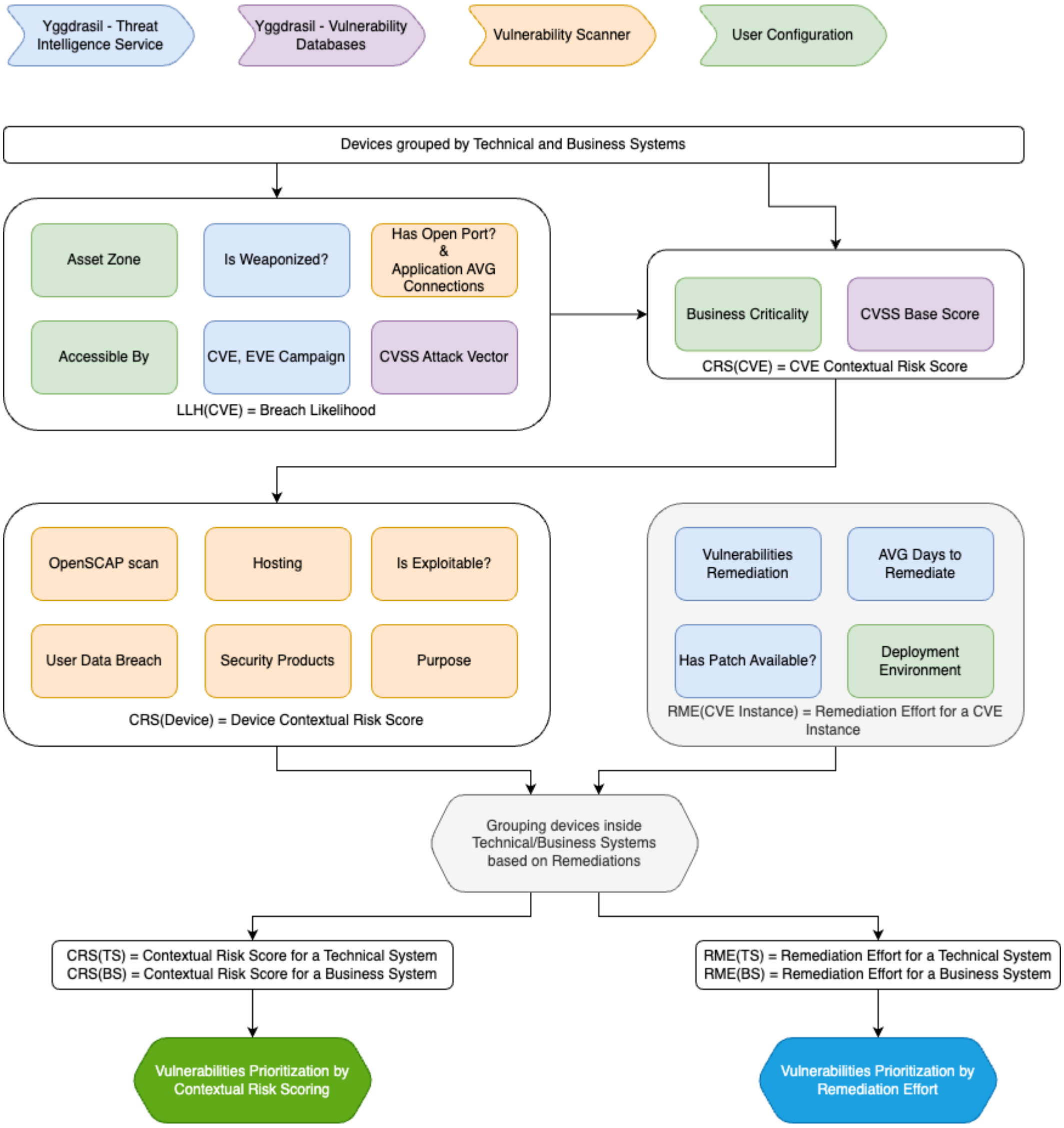
Yggdrasil - Threat Intelligence Service
Yggdrasil - Vulnerability Databases
Vulnerability Scanner
User Configuration
Devices grouped by Technical and Business Systems
Asset Zone
Is Weaponized?
Has Open Port? & Application AVG Connections
Accessible By
CVE, EVE Campaign
CVSS Attack Vector
LLH(CVE) = Breach Likelihood
Business Criticality
CVSS Base Score
CRS(CVE) = CVE Contextual Risk Score
OpenSCAP scan
Hosting
Is Exploitable?
User Data Breach
Security Products
Purpose
CRS(Device) = Device Contextual Risk Score
Vulnerabilities Remediation
AVG Days to Remediate
Has Patch Available?
Deployment Environment
RME(CVE Instance) = Remediation Effort for a CVE Instance
Grouping devices inside Technical/Business Systems based on Remediations
CRS(TS) = Contextual Risk Score for a Technical System
CRS(BS) = Contextual Risk Score for a Business System
RME(TS) = Remediation Effort for a Technical System
RME(BS) = Remediation Effort for a Business System
Vulnerabilities Prioritization by Contextual Risk Scoring
Vulnerabilities Prioritization by Remediation Effort

Fig. 1. Scoring and Prioritization System Components

associated according to their prediction model. Also, details about zero-day vulnerabilities that do not yet have an associated CVE will be taken from the Threat Intelligence service, as EVEs [23].

Early Vulnerability Exposures (EVEs) is a temporary ID for a zero-day vulnerability that eventually will get a CVE ID. These are predicted according to [23] using NLP models that were trained on multiple cyber security news from several news platforms.

The data used in the risk calculation is:

- CVE / EVE CVSS Base Score
- CVE / EVE Attack Vector

Is Weaponized - this information retrieved from the Threat Intelligence Service represents whether the CVE or EVE have a public exploit available. The proposed values with associated scores are:

- Yes - 100
- No - 50

Has Patch Available - the information on whether a patch is available or not is based on output provided from [21]. It influences the remediation effort necessary to fix a vulnerability, a patch available reduces the time consumed in finding a mitigation. The proposed values with associated scores are:

- Yes - Decrease the effort with 20%
- No - Do not influence the score

AVG Days to Remediate - this average is calculated based on the historical information that is sent periodically from the vulnerability management platform to the Yggdrasil. This information contains details from the remediation workflow component that helps in vulnerability remediation tracking. The most two important parameters are the number of days spent and the number of devices fixed for a certain remediation.

3.2. Dynamic Vulnerability Indicators

The data used as input for Dynamic Vulnerability Indicators component is taken from the Threat Intelligence service where a trend score and patch details for each vulnerability is established. The data used in the risk calculation is:

- CVE, EVE Campaign
- CVE Patch
- CVE Remediation Efforts

The term named Campaign discussed in [24] is a score that indicates whether a CVE or EVE is discussed worldwide because it is used by the attackers in the moment of calculating the scores.

3.3. Vulnerability Context

This information comes from both scanning and user-provided configuration. This score is computed from the following:

- **User Perspective settings**: Business Criticality, Asset Zone, Accessible by, Deployment Environment

- **Environment configuration**: Hosting, Purpose, Security Products, User Data Breach, Has Open Port, AVG Application Connections, OpenSCAP tests, Is Exploitable

3.**3**.1. *Perspective Settings.*

Business Criticality - a very important parameter that is set as input from the user to understand what is critical and what is less important for the respective organizations. It represents a prioritization based on the impact of potential security incidents on an organization's services and the proposed values are between:

- 0 - No business criticality
- 100 - The highest business criticality

Asset Zone - following the analysis, we considered it necessary to take into account the positioning of devices in the network for the risk calculation. The likelihood is higher that a device exposed to the Internet will be compromised than a device that is not exposed to the Internet. The proposed values with associated scores are:

- Internet - 100
- WAN - 80
- Isolated - 60
- Offline - 0

Accessible by - the risk increases with the number of entities that have access to the device under analysis. Thus, the highest risk is encountered when both other servers, services and users have access to the device. The proposed values with associated score are:

- Servers & Users - 100
- Only Users - 80
- Only Servers - 60
- Nobody - 0

Deployment Environment - the environment type is relevant in calculating the remediation effort of fixing vulnerabilities. In most cases, the involved effort in a production environment and the impact of failure deployment is higher than the others. The proposed values with associated scores are:

- Production - increase by 20%
- Staging - do not influence the remediation effort
- Development - decrease by 20%

3.**3**.2. *Environment Configuration.*

Hosting - Devices running in the cloud are at greater risk even though there are more automated security checks in the cloud, as IP addresses belonging to cloud providers are scanned every day by automated bots. In addition, if a system is compromised, you don't have complete control over it like you would in your own datacenter. The proposed values with associated risk are:

- Cloud - increase by 5%
- Datacenter - do not influence

Purpose - in risk analysis, it is important to consider what type of device we are looking at because a vulnerability running on a server will have a greater impact in most cases than a vulnerability running on a workstation or personal laptop. The values it may take are:

- Server - increase by 5%
- Workstation - do not influence

Security Products - is related to security products installed on the analyzed device. Their status is indicated by either of the following values:

- Up to date - do not influence
- Out of date - increase by 5%
- None - increase by 10%

User Data Breach - users are the weak point when it comes to IT security. Check if users' data has been compromised in data breaches. The service aggregates data from various publicly disclosed breaches and allows users to search for their email addresses or usernames to see if they have been involved in any known breaches. Either of the following values indicates their status:

- Yes - increase by 5%
- No - do not influence

Has Open Port - the exploitation likelihood is increased if the application that has the vulnerability opens a port that listens for connections. The probability will be decreased if no port is open related to the CVE even if the attack vector is Network. Either of the following values indicate their status:

- Yes - 1
- No - 0

AVG Application Connections - as the number of application connections increases the exploitation probability increases. Furthermore, if the vulnerability has the Attack Vector on the highest score, Network, the risk is boosted.

OpenSCAP tests - the OpenSCAP scan evaluates configuration information, and device state, and outputs a comprehensive assessment report. For the proposed solution, we use the compliance score that indicates the weight of total compliant findings which represents the security status.

Is Exploitable - this score is calculated based on the vulnerability prerequisites check provided by the agents' system and the CVE Attack Vector value. If the prerequisites are not fulfilled the score is 1, otherwise, the values are dependent on AV:

- Physical - 1
- Local - 1.1
- Adjacent - 1.15
- Network - 1.2

3.4. Contextual Risk Scoring System

In equation 1 we calculate the breach likelihood and the impact of a breach considering the Asset Zone, Is Weaponized, Campaign, Accessible By, Attack Vector, and the Connections number on the application that has the CVE.

$$\begin{aligned} LLH(CVE) = 0.3 * AssetZone + \; 0.4 * IsWeaponized \\ + \; 0.1 * Campaign + \; 0.1 * AccessibleBy \\ + \; 0.1 * HasOpenPort * AttackVector * (1 - \frac{1}{2^{max(1, AVGAppConn)}}) \end{aligned} \tag{1}$$

In equation 2 we calculate the contextual risk score of the CVE starting from the CVSS Base Score, considering the likelihood of a breach and introducing the business criticality factor to indicate the importance inside the organization.

$$\begin{aligned} CRS(CVE) = MIN(100, IsExploitable \\ *(0.5 * CVSSBaseScore_{0-100scale}) \\ +0.25 * LLH(CVE) \\ +0.25 * BusinessCriticality)) \end{aligned} \tag{2}$$

In equation 3 the application CRS is calculated based on the sum of critical and severity vulnerabilities CRS and the average for medium and low. In equations 4, 5, 6 we calculate the sum of multiple CRS scores considering the CVEs severity.

$$\begin{aligned} CRS(Application) = Sum(CRS(CVE\_critical)) \\ +Sum(CRS(CVE\_high)) \\ +AVG(CRS(CVE\_medium)) \\ +AVG(CRS(CVE\_low)) \end{aligned} \tag{3}$$

$$CRS\_SUM = SUM(CRS(Application)) \tag{4}$$

$$\begin{aligned} CRS(Device) = CRS\_SUM * \ ((3 - OpenSCAPScan)/2) \\ * \ UserDataBreach * \ SecurityProducts \\ * \ Hosting * \ Purpose \end{aligned} \tag{5}$$

$$\begin{aligned} CRS(TS) = SUM(CRS(Device)) \\ CRS(BS) = SUM(CRS(TS)) \end{aligned} \tag{6}$$

3.5. Remediation Effort

In general, in prioritizing vulnerabilities same remediation can be applied for multiple CVE instances on multiple devices from multiple Business and Technical Systems. Therefore, an important point of view is to group vulnerabilities by remediation and analyze both Contextual Risk Scoring and Remediation Efforts per collection. Formulas related to remediation efforts are presented in equation 7. The average CVE remediations in man-days are retrieved from the Threat Intelligence service which calculates them based on historical data and collected data from different sources. Historical data contains periods tracked in the CODA Footprint platform from the beginning of vulnerability remediation to its resolution.

$$\begin{aligned} RME(CVE) = AVG(CVE\_Remediations\_man\_days) \\ *HasPatchAvailable * DeploymentEnvironment \\ RME(Device) = SUM(RME(CVE)) \\ RME(TS) = SUM(RME(Device)) \\ RME(BC) = SUM(RME(TS)) \end{aligned} \tag{7}$$

4. Evaluation

In Figure 2 we created two showcases frequently encountered in real environments for two devices used in our laboratory and named Hermes and Hefaistos. The first one has a reduced number of vulnerabilities, including a greater percentage of *Critical* severity, while the second device has a higher number of *Medium* and *Low* vulnerabilities. Despite this inconsistency in the number of vulnerabilities, the Hermes device achieved a higher CRS even if it has almost a third of the number of vulnerabilities discovered on the Hefaistos device.

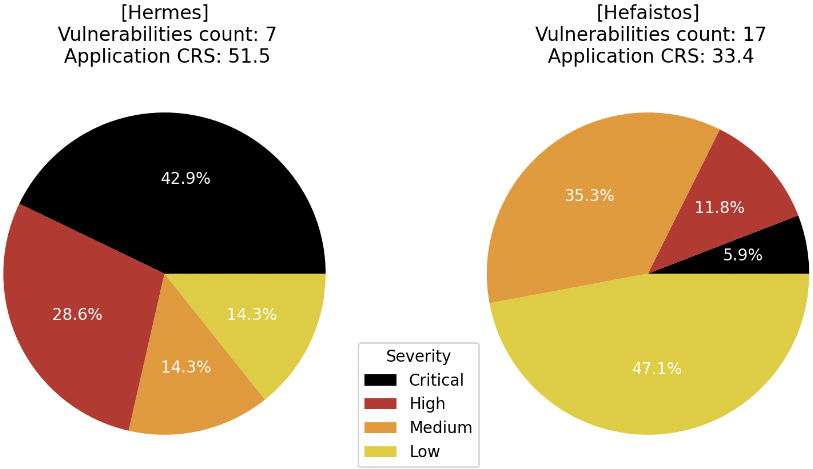

FIG. 2. Application CRS in showcases

To enhance readability of the Table 1 and 2, we have created o list of abbreviations as follow: BC - Business Criticality, WP - Weaponized, CP - Campaign, OP - Has Open Port, OST - OpenSCAP Tests, PP - Purpose, UDB - User Data Breach, AZ - Asset Zone, AB - Accessible By, CN - AVG Connections Number, EP - Exploitable.

Table 1 present multiple showcases targeted at understanding diverse influences on CRS across various context. We started the evaluation from an initial config for the Hermes device that has a couple of vulnerabilities that results in an initial CRS_SUM of *168*. Afterwards, we gradually changed certain indicators to observe the CRS variation.

In Table 2 we present the evolution in time of a Common Vulnerability Exposure detected on a certain server during the scans. We choose *CVE-2022-41080* vulnerability having a 9.8 CVSS 3.1 Base Score that affects Microsoft Exchange Server that was exploited in the wild from the beginning of 2023. Starting with the discovered point on a certain device, and going through several stages, it ended up being in the top list for Contextual Risk Report, reaching the top CRS score of 100.

Tbl. 1. Device CRS variation by indicators changes

FQDN	Initial	Stage 1	Stage 2	Stage 3	Stage 4
Hosting	Datacenter	Datacenter	Datacenter	**Cloud**	Datacenter
OST	0.4	0.4	0.4	**0.2**	0.4
SP	UpToDate	UpToDate	UpToDate	UpToDate	**OutOfDate**
PP	Workstation	Workstation	Workstation	Workstation	**Server**
UDB	No	**Yes**	No	No	No
BC	60	60	**90**	60	60
AZ	WAN	WAN	**Internet**	WAN	WAN
AB	Servers &Users	Servers &Users	Servers &Users	Servers &Users	Servers &Users
CRS_SUM	168	168	**177**	**168**	**168**
Device CRS	219	**229**	**230**	**248**	**252**

Tbl. 2. CRS Evolution in time for an instance of CVE-2022-41080 vulnerability

Date	**Event**	**BC**	**WP**	**CP**	**PT**	**CN**	**EP**	**LLH**	**CRS**
4 dec 22	First Discovered	50	No	None	No	0	No	60	**76.5**
6 dec 22	Business Criticality increased	**80**	No	None	No	0	No	60	**84**
11 dec 22	Port opened	80	No	None	**Yes**	0	No	**65**	**85.25**
12 dev 22	Number of Connections increased	80	Yes	None	Yes	**50**	No	**70**	**86.5**
14 dec 22	CVE becomes Weaponized	80	**Yes**	None	Yes	0	No	**90**	**91.5**
20 dec 22	Exploited in the wild (in trend)	80	Yes	**High**	Yes	50	No	**100**	**94**
15 jan 23	Became Exploitable	80	Yes	High	Yes	50	**Yes**	100	**100**

Therefore, a vulnerability that was not important at the beginning of its lifecycle reached the top of the CRS Report at a certain moment, providing a concrete example that highlights the cyberspace dynamics.

Tbl. 3. Remediation Effort Showcases

Remediation	**CRS Business System**	**Remediation Effort**	**Mitigation**	**Patch**
Update to version 20.24	2040	14 days	No	Yes
Consult vendor for workarounds	1580	3 days	Yes	No

In Table 3 we presented a showcase with two different remediations to apply for multiple applications within the same Business System. Although the first remediation has a higher Contextual Risk Score, this remediation effort is almost 5 times higher than the next remediation with a lower score. Thus, in this case, a better approach is to start with the second remediation, apply the temporary mitigation, and afterwards, start fixing the first remediation. Analyzing from both CRS Report and Remediation Effort perspectives is essential before sending to execution in the Collaborative Threat Remediation Workflows.

5. Discussions

Because networks are in a state of constant change, the scores obtained from the suggested solution are susceptible to abrupt modifications. The inherent volatility of the situation may present a potential disadvantage, as remediation teams might commence actions to patch specific vulnerabilities, only to discover the emergence of new, more critical concerns shortly thereafter. This situation highlights the difficulty associated with giving priority to vulnerabilities within an ever-changing security environment.

Although the new release of CVSS v4 proposal has lots of improvements, it is still offering real-time threat data and additional impact details only from a theoretical perspective. This work presented implementations for the most important aspects of CVSS v4 specification document, which are presented in Table 4. The proposed solution uses multiple details from the Threat Intelligence service such as campaign, zero-day vulnerabilities, impact on vulnerabilities in progress, and information about patches.

All of the scores and weights proposed in Section 3 were established based on our deep analysis carried out over the past few years. We focused on keeping the least relevant parameters from having a significant influence on the critical findings. Regarding the scale change in Contextual Risk Scoring, we decided to use the 0-100 scale for CVE CRS and an unbounded upper scale for the

Tbl. 4. CVSS v4 Specifications - Proposed Components

CVSS v4 principle	**Proposed Components**
Threat Metric Group	Dynamic Vulnerability Indicators
Environmental Group	Vulnerability Context Indicators
Vulnerability Response Effort	Remediation Effort
CVSS-BTE	Contextual Risk Score

Application, Device, Technical System, and Business System CRS due to the high level of complexity and dynamism.

6. **Conclusions**

The primary goal of this paper is to find a solution for prioritizing vulnerabilities' remediation designed for implementation of certain missing fundamentals of CVSS V4. The goal was achieved by implementing a solution grounded in combining multiple types of knowledge in modeling a risk calculation system.

To achieve this, we drilled down into CVSS solutions and leveraged data collected from an intelligent agent system and data from the Yggdrasil Threat Intelligence Service. Based on these data, we succeeded in creating 3 indicators used in the vulnerability's score computation: static, dynamic, and contextual.

There are also some limitations regarding validation in this research subject, as Jung et al. [13] concluded, it is hard to evaluate a new proposal because we need a structure and certain standards to compare different approaches and extract future improvements.

What sets this project apart from other established solutions is the incorporation of numerous data types that were purposefully selected for risk assessment, in addition to evaluating them using a probabilistic model within the given context. An additional benefit of our solution pertains to the standardization of data, which enables the integration of said data through the utilization of consistent risk evaluation parameters. In this paper, we have presented a fundamental risk calculation model. Nevertheless, the ranking of security aspects may be modified in conformity with the specific activities of the organization, according to a distinct importance hierarchy.

This project highlights the fact that evaluating a network is a complex undertaking that cannot be fully captured by a single method, as it is dependent on a multitude of results obtained from empirical data. However, by utilizing a methodology that combines concrete elements and mathematical principles with statistical approaches and industry security norms, it is possible to obtain approximative outcomes that assist in mitigating the risks associated with data corruption, monetary depletion, or system shutdowns.

REFERENCES

[1] V. L. Sujay, "Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2023, with forecasts from 2022 to 2030," 2023. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/

[2] "Security issues of IoT: Securing your IoT Device in 2023," Device Authority Ltd. [Online]. Available: https://www.deviceauthority.com/blog/security-issues-of-iot-securing-your-iot-device-in-2023/

[3] "Data security: How a proactive C-suite can reduce cyber-risk for the enterprise," The Economist Intelligence Unit, 2016. [Online]. Available: https://impact.economist.com/perspectives/technology-innovation/data-security-how-proactive-c-suite-can-reduce-cyber-risk-enterprise/article/data-security-how-proactive-c-suite-can-reduce-cyber-risk-enterprise

[4] "Proactive cybersecurity: What it is and why it matters," PDQ, 2023. [Online]. Available: https://www.pdq.com/blog/proactive-cybersecurity/

[5] "Common Vulnerability Scoring System." [Online]. Available: https://www.first.org/cvss/

[6] "Common Vulnerability Scoring System version 4.0: Specification Document," FIRST. [Online]. Available: https://www.first.org/cvss/v4.0/specification-document

[7] "Why organizations struggle with vulnerability management?" [Online]. Available: https://heimdalsecurity.com/blog/vulnerability-management-challenges/

[8] "Forum of incident response and security teams." [Online]. Available: https://www.first.org/

[9] "Common Vulnerability Scoring System version 4.0: Specification Document." [Online]. Available: https://www.first.org/cvss/v4.0/specification-document

[10] "Cisa known exploited vulnerabilities catalog." [Online]. Available: https://www.cisa.gov/known-exploited-vulnerabilities-catalog

[11] "Exploit Prediction Scoring System (EPSS)." [Online]. Available: https://www.first.org/epss/

[12] "Stakeholder-specific vulnerability catego- rization (svcc)." [Online]. Available: https://www.cisa.gov/stakeholder-specific-vulnerability-categorization-ssvc

[13] B. Jung, Y. Li, and T. Bechor, "Cavp: A context-aware vulnerability prioritization model," *Computers & Security*, vol. 116, p. 102639, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404822000384

[14] V. Ahmadi Mehri, P. Arlos, and E. Casalicchio, "Automated context-aware vulnerability risk management for patch prioritization," *Electronics*, vol. 11, no. 21, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/21/3580

[15] "Rudder cve plugin," accessed: 2024. [Online]. Available: https://docs.rudder.io/reference/6.2/plugins/cve.html

[16] O. Grigorescu, C. Săndescu, and R. Rughiniș, "CODA Footprint Continuous Security Management Platform," 2016, pp. 1–5.

[17] "Automatic system for early detection of cyber vulnerabilities," FIRST. [Online]. Available: https://yggdrasil.codaintelligence.com/

[18] "Cis benchmarks list," accessed: 2024. [Online]. Available: https://www.cisecurity.org/cis-benchmarks

[19] "Nist, security content automation protocol," 2022. [Online]. Available: https://csrc.nist.gov/projects/security-content-automation-protocol

[20] "Openscap portal." [Online]. Available: https://www.open-scap.org/

[21] D. Iorga, D.-G. Corlatescu, O. Grigorescu, C. Sandescu, M. Dascalu, and R. Rughinis, "Yggdrasil—early detection of cybernetic vulnerabilities from twitter," in *2021 23rd*

International Conference on Control Systems and Computer Science (CSCS). IEEE, 2021, pp. 463–468.

[22] I. Babalau, D. Corlatescu, O. Grigorescu, C. Sandescu, and M. Dascalu, "Severity prediction of software vulnerabilities based on their text description," in *2021 23rd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC).* IEEE, 2021, pp. 171–177.

[23] D. Iorga, D. Corlătescu, O. Grigorescu, C. Săndescu, M. Dascălu, and R. Rughiniş, "Early detection of vulnerabilities from news websites using machine learning models," in *2020 19th RoEduNet Conference: Networking in Education and Research (RoEduNet).* IEEE, 2020, pp. 1–6.

[24] C. Vladescu, M.-A. Dinisor, O. Grigorescu, D. Corlatescu, C. Sandescu, and M. Dascalu, "What are the Latest Cybersecurity Trends? A Case Study Grounded in Language Models," in *2021 23rd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Dec. 2021, pp. 140–146.