REVS: A Vulnerability Ranking Tool for Enterprise Security

Igor Forain\textsuperscript{a}, Robson de Oliveira Albuquerque\textsuperscript{b} and Rafael Timóteo de Sousa Júnior\textsuperscript{c}

Professional Program in Electrical Engineering (PPEE), Dept. of Electrical Engineering (ENE), University of Brasília (UnB), Brasília, Brazil

Keywords: Cybersecurity, Vulnerabilities, Pentest, NVD, CNVD, TOPSIS.

Abstract: Information security incidents currently affect organizations worldwide. In 2021, thousands of companies suffered cyber attacks, resulting in billions of dollars in losses. Most of these events result from known vulnerabilities in information assets. However, several heterogeneous databases and sources host information about those flaws, turning the risk assessment difficult. This paper proposes a Recommender Exploitation-Vulnerability System (REVS) with the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) to rank vulnerability-exploit. The REVS is a dual tool that can pinpoint the best exploits to pentest or the most sensitive vulnerabilities to cybersecurity staff. This paper also presents results in the GNS3 emulator leveraging data from the National Vulnerability Database (NVD), the China National Vulnerability Database (CNVD), and Vulners. They reveal that the CNVD, despite data issues, has 23,281 vulnerabilities entries unmapped in the NVD. Moreover, this work establishes criteria to link heterogeneous vulnerability databases.

1 INTRODUCTION

Today, most organizations need to provide services in computing environments. Beyond the impact on the healthcare industry worldwide, the COVID-19 epidemic has also had a disruptive effect on the way businesses operate (Ferreira et al., 2021). Once performed in person, services and processes had to undergo an almost instantaneous digital re-adaptation to remote access (Khan et al., 2020).

The digitalization of processes has been underway since the Internet became ubiquitous. However, it increased the attack surface (Pimenta Rodrigues et al., 2017; Chowdhary et al., 2020; Gualberto et al., 2020). The rush to provide services based on Cloud Computing has been accompanied by the introduction of a series of vulnerabilities in the IT environments of organizations, especially with the advent of services based on the Internet of Things (IoT) (Liu et al., 2019; Thamilarasu and Chawla, 2019).

Although there is a plethora of data regarding information security, several heterogeneous databases and sources host those data, turning the risk assessment difficult (Du et al., 2019). Besides, most organizations still lack effective methods to choose the best option and path in real scenarios (Bertoglio and Zorzo, 2017). The National Vulnerability Database (NVD) from the National Institute of Standards and Technology (NIST) provides reliable information about software and hardware flaws (Mavroeidis and Bromander, 2017; Hemberg et al., 2020).

However, there might be a delay between NVD and other open sources (Rodriguez et al., 2018). For this reason, new databases like China National Vulnerability Databases (CNVD) (CNCERT/CC, 2021), Metasploit, and Vulners (VULNERS, 2021) can help achieve better situational awareness of enterprise risks.

Also, picking the optimal attack action and exploit in the wild is still an open question (Kanakogi et al., 2021a). To this end, Vulnerability Assessment (VA) and Penetration Testing (PT) are essential steps (Shah and Mehtre, 2015; Yaqoob et al., 2017; Ghanem and Chen, 2020; Zhou et al., 2021). However, choosing the most critical and threatening outcomes provided by those steps is a decision process subjected to several constraints.

There are several approaches to solve decision process issues. They can use from Markov chain to Deep Learning algorithms (Awiszus and Rosenhahn, 2018). However, the VA is bounded by several attributes from the Common Vulnerability Scoring System (CVSS) (Cheng et al., 2012). So, it becomes
an optimization problem that can leverage operations research methods like the Multi-Criteria Decision-Making (MCDM) (Dožić, 2019). The Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) is an algorithm of the MCDM family used for the cybersecurity of 5G networks (Kholidy, 2022), power control systems (Liu et al., 2010) and Intrusion Detection System (IDS) (Alharbi et al., 2021).

Based on the described open issues, this work aims to optimize the vulnerability-exploit choosing process and integrate new vulnerability databases. To this end, this work proposes the Recommender Exploitation-Vulnerability System (REV$S$) based on a MCDM approach. It leverages data from the NVD, CNVD, and Vulners to create a vulnerability-exploit ranking using the TOPSIS algorithm. REV$S$ is a dual tool that can pinpoint the best exploits to attackers or the most sensitive vulnerabilities to cybersecurity staff. The experimental results reveal that the CNVD, despite data issues, has 23,281 vulnerabilities entries unmapped in the NVD. Moreover, this work establishes criteria to link heterogeneous vulnerability databases.

This work is structured as follows. Section 2 presents background and related work. Section 3 describes the proposed architecture. Section 4 uses GNS3 and open source tools to implement the architecture. Finally, section 5 concludes this article.

2 LITERATURE REVIEW

This section presents a literature review related to this work. It presents vulnerabilities and exploits and their available databases. Besides, it reviews approaches regarding attack path prediction and pentest automation. Finally, it shows state-of-the-art recommendation algorithms applied in the cybersecurity field.

2.1 Vulnerabilities and Exploits

Wang and Guo proposed an Ontology for Vulnerability Management (OVM) to achieve knowledge representation of the Common Vulnerabilities Exposures (CVE) of the NVD (Wang and Guo, 2009). It was the first attempt to create a knowledge base (KB) of vulnerabilities, but it did not consider the exploits possibility, different from this work. GAO et al. created a taxonomy and ontology for network attack classification using description logic (DL) (Gao et al., 2013). Kanakogi et al. used natural language processing (NLP) to match CVE to Common Attack Pattern Enumerations and Classifications (CAPEC) (Kanakogi et al., 2021a; Kanakogi et al., 2021b). These works guided the CVE data model that REV$S$ uses for vulnerability databases.

Householder et al. evaluated a systematic analysis of the relation between vulnerabilities and exploits. By the end of 2019, only around 4.1% of the vulnerabilities exposed after 2013 had an exploit publication (Hu et al., 2020). It supported this work in giving more weight to vulnerability features than exploit.

Rodriguez et al. showed that vulnerability disclosure delay between the NVD and other open sources, e.g. in the SecurityFocus, may reach 244 days (Rodriguez et al., 2018). Rytel et al. compared several databases of vulnerabilities, like the NVD and CNVD, but only regarding the IoT devices (Rytel et al., 2020). They presented the necessity of using more vulnerability databases besides the NVD.

The works presented in this subsection proposed vulnerability ontologies or evaluated vulnerability database assessment. Different from them, this work leverages and integrates those databases for pentest and security assessment.

2.2 Attack Path and Pentest

Valea and Oprisa proposed pentest automation using Nmap and Metasploit. Unlike this work, it covered only vulnerabilities regarding a root shell with Meterpreter and used decision trees to avoid overfitting (Valea and Oprisa, 2020). REV$S$ leverages the TOPSIS algorithm. Polatidis et al. (Polatidis et al., 2017; Polatidis et al., 2020) proposed an attack path prediction algorithm to achieve an information risk assessment. Their work used only CVE data from the MITRE Corporation (MITRE, 2021) on IT maritime infrastructure to generate the attack graphs. This work uses more databases with a different optimization approach.

Huo et al. used the multi-host Multi-Stage Vulnerability Analysis (MulVAL) algorithm to generate the attack tree. It leveraged a Deep Q-Learning Network (DQN) to choose the best attack path based on the CVSS of the CVE as a reward function. Their experiment evaluated a small topology without specification about the information assets and exploits only CVE-2012-0053 (Hu et al., 2020). It showed the high computational cost of deep learning approaches for extensive networks and sometimes convergence issues. This work uses an operation research method and improves the integration between Nmap and external data sources.
2.3 Recommender Systems

Pawlicki et al. wrote a comprehensive survey about recommendation systems for Cybersecurity (Pawlicka et al., 2021). It showed to this work that type of systems as a promising approach to the vulnerability-exploit recommendation. Polatidis et al. used a recommendation system with a multi-level collaborative filtering method (Polatidis et al., 2018; Polatidis and Georgiadis, 2017) to Microsoft Windows. REVS leverages TOPSIS targeting any platform.

Some works treated the recommendation problem with an MCDM approach. It relies on minimizing or maximizing the geometric distance from an ideal solution like the classical recommendations systems. Most of them leveraged the TOPSIS for security assessment and Intrusion Detection System (IDS), but not for pentesting (Kholidy, 2022; Alharbi et al., 2021; Liu et al., 2010).

Table 1: Comparison with Related Works.

<table>
<thead>
<tr>
<th>Work</th>
<th>Data Sources</th>
<th>Algorithm</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Pawlicka et al., 2021)</td>
<td>Doesn’t apply</td>
<td>Doesn’t apply</td>
<td>Survey</td>
</tr>
<tr>
<td>(Polatidis et al., 2018)</td>
<td>Mitre CVE</td>
<td>CF</td>
<td>MS Windows</td>
</tr>
<tr>
<td>(Polatidis and Georgiadis, 2017)</td>
<td>Mitre CVE</td>
<td>CF</td>
<td>MS Windows</td>
</tr>
<tr>
<td>(Kholidy, 2022)</td>
<td>NVD, Mitre exploit</td>
<td>TOPSIS</td>
<td>5G networks</td>
</tr>
<tr>
<td>(Alharbi et al., 2021)</td>
<td>Doesn’t apply</td>
<td>TOPSIS</td>
<td>IDS attributes</td>
</tr>
<tr>
<td>(Liu et al., 2010)</td>
<td>Private</td>
<td>TOPSIS</td>
<td>Power Systems</td>
</tr>
<tr>
<td>This work</td>
<td>NVD, CNVD, Metasploit, Vulners</td>
<td>TOPSIS</td>
<td>IPv4, IPv6</td>
</tr>
</tbody>
</table>

Table 1 presents an outline comparison between the works cited in this subsection and this work. It shows that it leverages more vulnerability database sources to a broader network scope.

3 PROPOSED SYSTEM

This section describes the proposed architecture of the Recommender Exploitation-Vulnerability System (REVS). Figure 1 presents the system diagram, which has four main modules.

Four modules comprise the REVS: database, scanner, matcher, and recommender. The following subsections describe these four modules.

3.1 Database

The database module comprises two main sets: vulnerabilities and exploits. Their function is to provide numeric features to feed the TOPSIS decision matrix. REVS gathers the first set from two sources: the NVD and CNVD. The second set comes from Metasploit’s database. This layer is a batch process to set up the databases before the Matcher step. REVS uses the NVD and CNVD in an ensemble, matching registers with the same CVE number and joining the scores of the two databases.

3.1.1 NVD

The NIST from the USA supports the NVD. It is a well-known and de facto authority regarding system vulnerabilities for the research community (NIST, 2021). That database comprises several features: Common Weakness Enumeration (CWE), Common Platform Enumeration (CPE), vendors, and ratings (MITRE, 2021). The NVD uses the CVSS to rate the characteristics and severity of system vulnerabilities. The CVSS has two versions, v2.0 and v3.0, with a different range for similar attributes. Table 2 shows the features from CVSS used by REVS.

Table 2 presents the six CVSS features that comprise the criteria used by the TOPSIS decision matrix.

3.1.2 CNVD

The National Computer Network Emergency Response Technical Team/Coordination Center of China (CNCERT/CC) sponsors the CNVD. China has been a global actor in the information security field, making the CNVD a relevant source of vulnerabilities (CNCERT/CC, 2021). However, it did not provide an interface for data feeding, nor did it provide
Table 2: CVSS Features.

<table>
<thead>
<tr>
<th>CVSSv2</th>
<th>CVSSv3</th>
</tr>
</thead>
<tbody>
<tr>
<td>accessComplexity</td>
<td>attackComplexity</td>
</tr>
<tr>
<td>accessVector</td>
<td>attackVector</td>
</tr>
<tr>
<td>availabilityImpact</td>
<td>availabilityImpact</td>
</tr>
<tr>
<td>confidentialityImpact</td>
<td>confidentialityImpact</td>
</tr>
<tr>
<td>integrityImpact</td>
<td>integrityImpact</td>
</tr>
<tr>
<td>authentication</td>
<td>privilegesRequired</td>
</tr>
</tbody>
</table>

The TOPSIS requires calculating the Euclidean distance between the m vectors and the ideal solutions.

3.4 Recommender

This work approach chooses the best vulnerability-exploit pair using a decision-making method. In this case, REVS uses the TOPSIS algorithm based on the optimization technique described in (Hwang and Yoon, 1981; Chen and Hwang, 1992; Opricovic and Tzeng, 2004). The attack is a one-layer problem of picking up the best vulnerability-exploit pair, i.e., the lowest cost and highest impact. So, the decision matrix has m vulnerabilities (rows) compared to n features (columns).

\[
A = (a_{ij}) \quad i = 1, 2, \ldots, m; j = 1, 2, \ldots, n
\]  

Equation 1 shows that the \(a_{ij}\) is the \(j\)th feature value of the \(i\)th vulnerability. After that, the \(A\) is normalized and weighted by columns to its \(X\) form:

\[
X = (x_{ij}) \quad x_{ij} = \lambda_j * a_{ij} / \sqrt{\sum_{j=1}^{n} a_{ij}^2}
\]

The technique requires choosing the best and worst options from the attacker role.

\[
Z^+ = (z_1^+, z_2^+, \ldots, z_n^+ )
\]

\[
Z^- = (z_1^-, z_2^-, \ldots, z_n^- )
\]

Equations 5 and 6 indicate the calculus for benefit and cost features, respectively.

\[
z_j^+ = \max(x_{ij}) \quad z_j^- = \min(x_{ij})
\]

\[
z_j^+ = \min(x_{ij}) \quad z_j^- = \max(x_{ij})
\]

Now, there are \(m\) vectors with dimension \(n\), which are the rows of the matrix \(X\). There are also two new vectors with size \(n\), \(Z^+\) and \(Z^-\), which are the ideal solutions. The TOPSIS requires calculating the Euclidean distance between the \(m\) vectors and the ideal solutions.
Finally, calculate the performance ratio to rank each one of the vulnerabilities, from highest to lowest.

\[ p_i = \frac{d_i^+}{d_i^+ + d_i^-}, \quad i = 1, 2, 3, \ldots, m \]  

### 4 PRELIMINARY RESULTS

REVS is an ongoing project with preliminary results regarding the following modules: database, scanner, and matcher. This section presents the test environment based on the GNS3 and discusses those results.

#### 4.1 Test Environment

This work emulates a medium enterprise Security Operation Center (SOC) using Graphical Network Simulator-3 (GNS3). It uses Quick Emulator (QEMU) on kernel-based virtual machines (KVM) in Linux Ubuntu 20.04. Furthermore, KVM in Linux performs better than type 2 hypervisors like VirtualBox and VMware because of hardware acceleration and kernel embedded commands. Figure 2 presents the test environment.

![Emulated SOC with QEMU/KVM](image)

The GNS3 makes it possible to build an environment with different Operating Systems (OSs): Kali Linux, Ubuntu 18.04, CentOS 8, Windows Server, and Metasploitable III (Rapid7, 2021) machine. This last is a vulnerable public VM based on Ubuntu 14.04. Table 3 lists the environment hardware and VMs.

<table>
<thead>
<tr>
<th>Machine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Ryzen 7 4800h / 16gb RAM</td>
</tr>
<tr>
<td>Emulator</td>
<td>GNS3 2.2.28</td>
</tr>
<tr>
<td>Metasploitable III</td>
<td>Ubuntu 14.04</td>
</tr>
<tr>
<td>Firewall</td>
<td>pfSense 2.5.2</td>
</tr>
<tr>
<td>Routers</td>
<td>VyOS 1.1.8</td>
</tr>
<tr>
<td>Switches</td>
<td>Open vSwitch 2.4.0</td>
</tr>
</tbody>
</table>

### 4.2 Database Results

This work implements two data scrapers to collect vulnerabilities data from the NVD and CNVD. It downloads the two databases simultaneously on January 24, 2022, for a fair comparison. The first scraper uses the REST API supported by NIST, which provides vulnerability registers with unique CVE id and CVSS v2 and v3 scores. Besides, some of these registers also contain CPEs and CWEs related to the vulnerability. REVS makes use of the OPENCVE tool (Crocfer, 2020) to retrieve data from the REST API and store it locally in a PostgreSQL relational database.

The second scraper requires implementation from scratch. The CNVD does not provide any API to return vulnerability data. Otherwise, it provides a set of XML files lacking schema definition with part of the data scored by CVSS v2. Moreover, CNCERT/CC generates a new XML file every Monday 18h:00 (CST) with vulnerability registers from the past week. Different from the work of (Rytel et al., 2020), REVS uses the python requests library with custom "User-Agent" tag and cookie parameters (\_jsluid and \_jsl clearance\_s) to bypass the CNVD blocking system. Table 4 presents a summary of the NVD and CNVD.

<table>
<thead>
<tr>
<th>Feature</th>
<th>NVD</th>
<th>CNVD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vulnerability</td>
<td>178,906</td>
<td>99,261</td>
</tr>
<tr>
<td>Missing Weeks</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>Duplicated ID</td>
<td>0</td>
<td>88</td>
</tr>
<tr>
<td>Without CVE</td>
<td>0</td>
<td>23,281</td>
</tr>
<tr>
<td>Duplicated CVE</td>
<td>0</td>
<td>108</td>
</tr>
<tr>
<td>Nonexistent CVE</td>
<td>0</td>
<td>193</td>
</tr>
</tbody>
</table>

Table 4 shows that the CNVD dataset has the fol-
lowing issues: six missing week files, 23,281 vulner-
abilities without corresponding CVE, 88 vulnerabili-
ties with duplicated id, 193 entries linked to nonexistent
CVEs. They look like malformed CVEs during string copy from other databases. Moreover, preced-
ing files are not up-to-date; they started in January
2015, and most of the XML files have scape charac-
ters. REVS corrected these last issues and stored the
CNVD data in a relational database on PostgreSQL.
Figure 3 compares the NVD and the CNVD regarding
the CVSS, CPE, and external references.

Figure 3: Databases Comparison.

Figure 3 shows that despite being the main responsi-
ble for CVE Mitre framework implementation, the
NVD has 6.11% of the entries without any CVSS met-
ric, while the CNVD has 0.33%. Moreover, the NVD
and the CNVD have 6.23% and 0.11% of the entries
without CPE, respectively. This last issue prevents a
comprehensive identification of the vulnerable hard-
ware or software. The NVD carries more information
regarding external URL references, with 13.43% of
the entries without this information, while the CNVD
shows 19.15% missing this feature.

This work uses two approaches for exploits gath-
ering: parsing the entire Metasploit source code and
scraping the Exploit-DB website. The former executes a string pattern search looking for CVE men-
tions in the Metasploit source code structure. The lat-
ter scrapes the Exploit-DB website to get the relations
between exploits and CVEs. After that, REVS also
stores the results in PostgreSQL. Moreover, it lever-
ages the Vulners API on the fly to search for vulnera-
bilities and exploits.

4.3 Scanner and Matcher Results

Figure 2 shows that the attack scenario uses REVS
outside the topology against the Metasploitable III
VM on the DMZ. In this scenario, REVS runs in
the host machine against the guest VM emulated by
GNS3 using the NVD and the CNVD as vulnerability
database and Vulners as exploit source. It finds the
results listed in table 5.

<table>
<thead>
<tr>
<th>CPE</th>
<th>NVD</th>
<th>CNVD</th>
<th>Exploits</th>
</tr>
</thead>
<tbody>
<tr>
<td>proftpd:proftpd:1.3.5</td>
<td>9</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>linux:linux_kernel</td>
<td>6</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>apache:http server:2.4.7</td>
<td>44</td>
<td>38</td>
<td>53</td>
</tr>
<tr>
<td>samba:samba</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>apple:cups:1.7</td>
<td>6</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>mysql:mysql</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 5 shows that most of the vulnerabilities be-
gold to the Apache webserver. The table also presents
the same situation about the exploits available in the
Metasploit framework. Table 5 also shows that REVS
found 48 vulnerabilities using the CVE id as the
search key in the CNVD. The 68 vulnerabilities de-
tected by REVS using the NVD include these 48.

The 20 unmatched vulnerabilities in the CNVD
are before 2015. It explains why they are not in
the CNVD. However, a future approach is a new
search for these unmatched results using CPE as
the search key in the CNVD. Furthermore, a sec-
ond official database, the China National Vulnerabil-
ity Database of Information Security (CNNVD), also
requires comparison against NVD and CNVD.

Figure 4 presents the data returned from
REVS-Vulners interface regarding the CPE
proftpd:proftpd:1.3.5.

Figure 4: REVS and Vulners Interface Results.

Figure 4 shows that REVS-Vulners interface re-
turns 9 CVEs and two exploits from Metasploit. Be-
sides, it returns data regarding obsolete zero-days and
forum messages.
5 CONCLUSIONS

The CNVD has several protection mechanisms to prevent downloads despite being a public database. Also, there are missing data files and 23,281 vulnerabilities without CVE mapping. It indicates process issues or the existence of vulnerabilities known only to the Chinese community because most of the text data are in mandarin.

REVS is an ongoing project that is working on the results of the Recommender module. It has already downloaded and normalized the NVD and CNVD databases. Furthermore, REVS integrated those two national vulnerability databases, Nimap and Vulners, using CPE and CVE as the search keys. The vulnerability assessment against the VM behind a SOC emulated in GNS3 showed the NVD as a more comprehensive database than the CNVD.

As Future work, the authors suggest using the TOPSIS fuzzy version with attack paths calculated by the MULVAL algorithm in the recommender module. The batch translation to English of the already downloaded database and an automatic one for new data are necessary improvements for REVS. This research will go further into integrating with the CNNVD and NLP approaches to vulnerability search.

ACKNOWLEDGEMENTS

The authors acknowledge the support of ABIN grant 08/2019. R.d.O.A gratefully acknowledges the General Attorney of the Union - AGU grant 697.935/2019 and the General Attorney’s Office for the National Treasure - PGFN grant 23106.148934/2019-67. R.T.S.J. gratefully acknowledges the support from EC Horizon 2020 HEROES project grant 101021801, CNPq grants 465741/2014-2 and 312180/2019-5, the Administrative Council for Economic Defense grant 08700.000047/2019-14, and the National Auditing Department of the Brazilian Health System SUS (Grant 23106.118410/2020-85).

REFERENCES


