# Towards System Security: What a Comparison of National Vulnerability Databases Reveals

Igor Forain*†, Robson de Oliveira Albuquerque‡†, Rafael Timóteo de Sousa Júnior§†

†Professional Post-Graduation Program in Electrical Engineering - PPEE - Electrical Engineering Department,
Faculty of Technology, University of Brasília (UnB), Brasília, Brazil, Zip Code 70910-900
Email: igor.freire@aluno.unb.br*, robson@redes.unb.br‡, desousa@ene.unb.br§

*Abstract* — **System vulnerabilities are ubiquitous nowadays. In 2021, millions of cyberattacks exploited system flaws resulting in billions of losses. Despite massive vulnerability databases supported by the USA and China governments, there are still several unknown issues between them. This paper proposes a methodology to compare the National Vulnerability Database (NVD), the China National Vulnerability Database (CNVD), and the China National Vulnerability Database of Information Security (CNNVD). The results reveal that the CNNVD has 1,661 vulnerabilities entries more than the NVD and at least 40 more entries regarding Chinese vendors. Moreover, there is a temporal correlation of 0.917560 between the NVD and CNNVD. To the best of the authors' knowledge, this work is the first to normalize and compare the NVD, CNVD, and CNNVD using their data feeds.**

***Keywords - System Security; NVD; CNVD; CNNVD.***

## I. Introduction

Nowadays, computer systems are ubiquitous in daily life. The COVID-19 epidemic forced a rush in providing online services [1]. However, it also exposed system vulnerabilities and increased the attack surface [2], [3]. Although there are several databases regarding system vulnerabilities, the most comprehensive are those sponsored by the USA and China [4].

The National Vulnerability Database (NVD) from the National Institute of Standards and Technology (NIST) of the USA is the authoritative source of systems vulnerability information [5], [6]. It provides extensive data regarding system vulnerabilities using Common Vulnerability Exposure (CVE), Common Weakness Enumeration (CWE), Common Platform Enumeration (CPE), and Common Vulnerability Scoring System (CVSS) [7], [8]. The MITRE Corporation established these frameworks for data classification and enumeration regarding software vulnerabilities [9]. Afterward, it transferred these standards to NIST, which nowadays is responsible for the NVD [10].

The software and hardware supplied by vendors from China like Xiaomi and Huawei might carry new vulnerabilities which the NVD does not tackle [11]. For this reason, systems security improvement requires tracking social media [12], bug reports, or new vulnerability sources [13]. In this scenario, the China National Vulnerability Database (CNVD) and the China National Vulnerability Database of Information Security (CNNVD) are new options for better vulnerability assessment and mitigation [14], [15].

The CNVD [16] and the CNNVD [17] are also state-sponsored systems like the NVD. The National Computer Network Emergency Response Technical Team/Coordination Center of China (CNCERT/CC) supports the first. The China Information Technology Security Evaluation Center (CNITSEC) hosts the second as a project. They present several obstacles to their foreign users despite claiming public access. Both provide a web interface in mandarin for vulnerability search but do not offer an API for automatic download [4]. Moreover, despite an average disclosure time lower than the NVD, there is evidence of vulnerability hiding [18]. On the other hand, the NIST provides easy access to the NVD download through JSON files or Rest API but presents inconsistencies [19], [20].

Most works about mining vulnerability databases covered only the NVD [5], [21]–[26]. Besides, some approaches leveraged the NVD, CNVD, and CNNVD together but restricted to the Internet of Things (IoT) platform [4], [15] or used only the CNNVD to seek vulnerabilities [27]. This work addresses these open issues by comprehensively comparing the NVD, CNVD, and CNNVD. The process to achieve the comparison includes Python multiprocessing with PostgreSQL, Pandas, and Natural Language Processing (NLP) techniques [28].

The results reveal that the CNNVD has 1,661 vulnerabilities entries more than the NVD, more entries regarding Chinese vendors, and a temporal correlation of 0.917560 between the NVD and CNNVD. To the best of the authors' knowledge, this work is the first to normalize and compare the NVD, CNVD, and CNNVD using their data feeds. These results will enable the joint use of those three databases to improve vulnerability assessment (VA) and threat management.

This work is structured as follows. Section II presents background and related work. Section III describes the proposed methodology. Section IV explains the results. Finally, section V concludes this article.

## II. Literature Review

This section presents the background and related works. It describes the vulnerability framework from MITRE that comprises every vulnerability database. Besides, it reviews articles that use the NVD, CNVD, or CNNVD for mining vulnerabilities.

*2022 17th Iberian Conference on Information Systems and Technologies (CISTI)*
*22 – 25 June 2022, Madrid, Spain*
ISBN: 978-989-33-3436-2

### A. Background

The majority of the available public vulnerability databases rely on the CVE identification provided by CVE Numbering Authorities (CNA) [24]. The MITRE Corporation and the NIST are in the top tier of this workflow sponsored by the United States Department of Homeland Security (DHS) [24]. After MITRE registering and identifying a new vulnerability, it receives a CVE unique identifier, a summary description, and external references. NIST gets this data from MITRE and adds more features like a detailed description, vendor identification, CWE, CPE, and CVSS [20]. The latter is a scoring system to assign CVEs to a severity group based on a score from 0 to 10.

The CWE is a framework to group them according to a vulnerability type, e.g., web, buffer overflow, etc. Lastly, the CPE is a method to logically describe the affected hardware and software, considering versions and conditions for the vulnerability [29]. For example, a web server software may be only vulnerable when running on a specific Operating System (OS) [30].

There were attempts to propose an Ontology for the CVE system of the NVD [7]. Also, some approaches created a taxonomy and ontology together to mix vulnerability and threat [8], [21], [22]. They aimed to become the CVE representation more understandable and less inconsistent [19].

### B. Related Works

Despite several existing works regarding NVD mining, most of them leveraged some text mining (TM) algorithm to visualize trends and patterns [5], [24], [26], even chaotic patterns (CP) [23]. Also, there were other approaches using TM and machine learning (ML) to correlate the Common Attack Pattern Enumeration and Classification (CAPEC) and exploit to CVE [21], [22]. Lastly, there were works leveraging text mining to detect disclosure delays [25], NVD feature estimations [30] and inconsistencies [19], [20]. This work learned from these articles that the CVSS and the CPE features carry many inconsistencies, but none of them evaluated beyond the NVD. They did not leverage a data normalization (DN) approach.

TABLE I. COMPARISON WITH RELATED WORKS

| *Work* | *Data Sources* | *Approach* | *Objective* |
|---|---|---|---|
| [5], [24], [26] | NVD | TM | Data Visualization<br>Pattern Detection |
| [23] | NVD | CP | Pattern Detection |
| [21], [22] | NVD | TM | Link CAPEC to CPE |
| [25] | NVD,<br>Web Data | TM | Disclosure Delay |
| [30] | NVD | ML | CPE discovering |
| [19], [20] | NVD | TM,<br>ML | Data inconsistencies |
| [4], [15] | NVD,<br>CNVD,<br>CNNVD,<br>JNVD, etc. | TM | IoT devices |
| [11] | CNNVD | DS | Data Analysis |
| [27] | CNNVD | TM,<br>ML | Vulnerability,<br>Classification |
| [31] | CNNVD | DL | Severity Prediction |
| This work | NVD,<br>CNVD,<br>CNNVD | TM,<br>DN | Database Comparison |

Some works also leveraged more than one national vulnerability database. Two of them used the NVD, CNVD, and CNNVD to extract data, but only regarding the IoT devices [4], [15]. They leveraged a mix of web crawlers, XML and JSON data feeds to create an unstructured IoT vulnerability database.

Lastly, other approaches used only the CNNVD to evaluate data analysis through descriptive statistics (DS) [11], vulnerability classification with TM and ML [27] and severity prediction with deep learning (DL) [31] without comparing with the NVD. Table I presents a summary comparison between the related works and this work. It seeks a more extensive comparison between the NVD, CNVD, and CNNVD.

## III. METHODOLOGY

This section describes the methodology leveraged to get the vulnerability databases and evaluate the data analysis. Figure 1 presents the summary of the methodology flow, which has three steps.

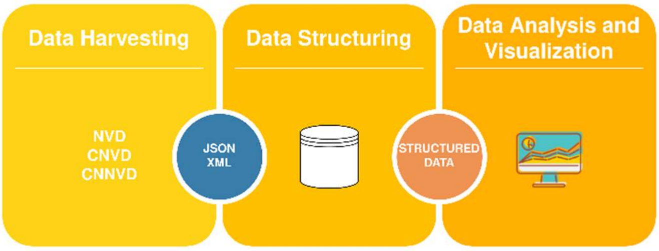


Figure 1. Summary Flow.

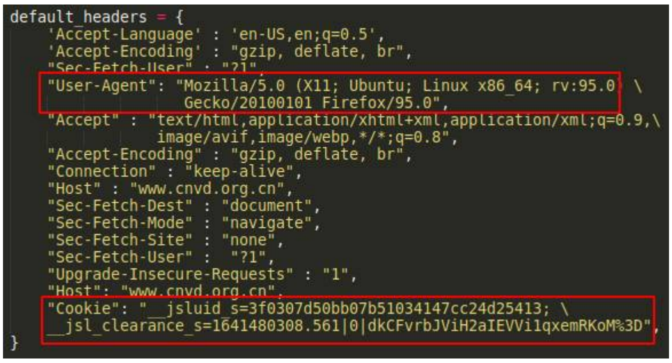
### A. Data Harvesting

The Data Harvesting step comprises the data collection of the following national vulnerability databases: the NVD, the CNVD, and the CNNVD. The first offers compressed data files for each year in the JSON format with the entire dataset. Besides, NIST also offers an API to get data receiving one CVE ID as a parameter. This work approach uses the OpenCVE [32] program to download the files and host them in a local PostgreSQL database. It provides a JSON data type that this work leverages to a fast store of the entire database.

The CNVD does not offer an interface or documentation regarding data download. Moreover, the website is in mandarin and blocks access when the user tries a lot of connections. However, the URLs to download the XML data files are hidden in the page source code. They follow the pattern https://www.cnvd.org.cn/shareData/download/ concatenated with an integer number from 1 to 900.

Furthermore, the web scraper must also set up the HTTP header with customized "User-Agent" tag and cookie (*__jsluid* and *__jsl_clearance_s*). The Web Browser sets up these parameters values during the HTTP GET Request to the CNVD Site. Figure 2 shows an example of a custom HTTP Tag to bypass the CNVD blocking system.

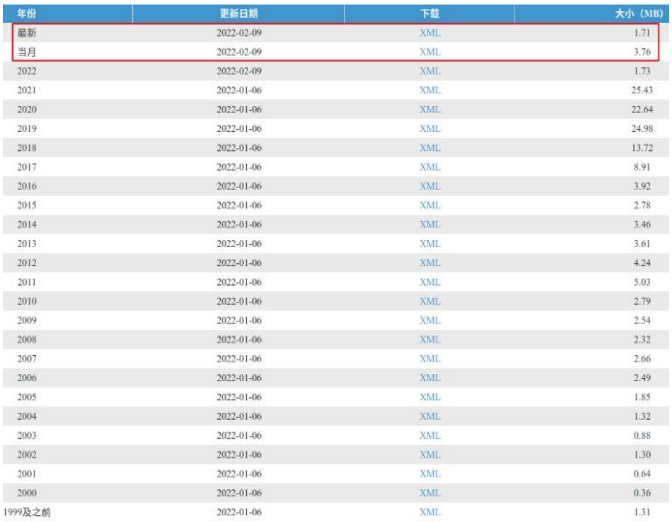
The CNNVD site is also in mandarin like the CNVD and claims to be an open database. It provides the entire data in XML format hosted in the URL http://www.cnnvd.org.cn/web/xxk/xmlDown.tag as shown in figure 3.



```
default_headers = {
    'Accept-Language' : 'en-US,en;q=0.5',
    'Accept-Encoding' : "gzip, deflate, br",
    "Sec-Fetch-User" : "?1",
    "User-Agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:95.0) \
                   Gecko/20100101 Firefox/95.0",
    "Accept" : "text/html,application/xhtml+xml,application/xml;q=0.9,\
                image/avif,image/webp,*/*;q=0.8",
    "Accept-Encoding" : "gzip, deflate, br",
    "Connection" : "keep-alive",
    "Host" : "www.cnvd.org.cn",
    "Sec-Fetch-Dest" : "document",
    "Sec-Fetch-Mode" : "navigate",
    "Sec-Fetch-Site" : "none",
    "Sec-Fetch-User" :  "?1",
    "Upgrade-Insecure-Requests" : "1",
    "Host": "www.cnvd.org.cn",
    "Cookie": "__jsluid_s=3f0307d50bb07b51034147cc24d25413; \
    __jsl_clearance_s=1641480308.561|0|dkCFvrbJViH2aIEVVi1qxemRKoM%3D",
}
```

Figure 2. HTTP GET Configuration

However, the download page in figure 3 does not work because of unavailable sign-in options. Anyway, the page source code hides the download URL to each file: the string pattern http://www.cnnvd.org.cn/ concatenated with the file name available in the screen presented in figure 3.



| 年份 | 更新日期 | 下载 | 大小（MB） |
|---|---|---|---|
| 最新 | 2022-02-09 | XML | 1.71 |
| 当月 | 2022-02-09 | XML | 3.76 |
| 2022 | 2022-02-09 | XML | 1.73 |
| 2021 | 2022-01-06 | XML | 25.43 |
| 2020 | 2022-01-06 | XML | 22.64 |
| 2019 | 2022-01-06 | XML | 24.98 |
| 2018 | 2022-01-06 | XML | 13.72 |
| 2017 | 2022-01-06 | XML | 8.91 |
| 2016 | 2022-01-06 | XML | 3.92 |
| 2015 | 2022-01-06 | XML | 2.78 |
| 2014 | 2022-01-06 | XML | 3.46 |
| 2013 | 2022-01-06 | XML | 3.61 |
| 2012 | 2022-01-06 | XML | 4.24 |
| 2011 | 2022-01-06 | XML | 5.03 |
| 2010 | 2022-01-06 | XML | 2.79 |
| 2009 | 2022-01-06 | XML | 2.54 |
| 2008 | 2022-01-06 | XML | 2.32 |
| 2007 | 2022-01-06 | XML | 2.66 |
| 2006 | 2022-01-06 | XML | 2.49 |
| 2005 | 2022-01-06 | XML | 1.85 |
| 2004 | 2022-01-06 | XML | 1.32 |
| 2003 | 2022-01-06 | XML | 0.88 |
| 2002 | 2022-01-06 | XML | 1.30 |
| 2001 | 2022-01-06 | XML | 0.64 |
| 2000 | 2022-01-06 | XML | 0.36 |
| 1999及之前 | 2022-01-06 | XML | 1.31 |

Figure 3. CNNVD Download Site.

Figure 3 shows that the CNNVD download page looks like the NVD page: one file for each year. It also provides two more files: daily updates in the first row and monthly updates in the second row. Moreover, this work starts downloading the NVD, CNVD, and CNNVD simultaneously to have a fair base for comparing the databases.

## B. Data Structuring

The JSON data type is not the best option for data and text processing with Python in PostgreSQL. So, this work evaluates a database normalization of the NVD. It also uses the same approach for the CNVD and CNNVD, with the difference that the NVD has deeply nested JSON. Moreover, the CPE in the NVD includes logical conditions that do not exist in the CNVD and CNNVD. Figure 4 describes the database tables.

The CNVD and CNNVD use a database model very similar to the NVD. The main difference is that they do not carry information about complex CPEs. This work stores the complex and basic CPEs in different tables to enhance the data analysis. The CNVD and CNNVD only list the vulnerable platforms without considering logical conditions. Furthermore, there are no data about CWE in these databases. This work leverages the Pandas library and list compression technique with the Python Multiprocessing library to accelerate the file reading, processing, and data normalization. Moreover, the data processing includes a data cleaning to remove the \r, \n, \t, and \\ characters.

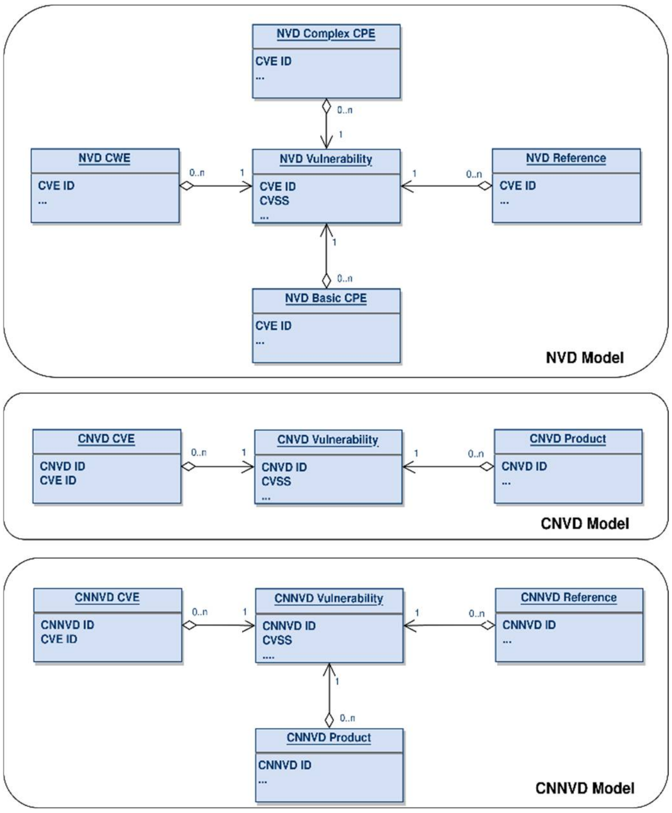


Figure 4. Data Models.

Figure 4 shows that the central entity is the vulnerability table. Besides, there are other entities for the CVE metrics: CVSS, description, CWE, CPE, and references. These last are data like URLs for external sites or exploits.

## C. Data Analysis and Visualization

This step depends on the previous one. The data analysis seeks inconsistencies between the NVD, CNVD, and CNNVD. So, it requires a descriptive outline of each database regarding the features presented in table II. The features of table II allow a database summary and tracking the evolution of each database since the beginning of the historical series. For example, this work will look for a possible correlation between the three national vulnerability databases.

Lastly, this works leverages text mining techniques in PostgreSQL to seek uncommon vulnerability descriptions, vendors, and platforms. This part pays more attention to comparing Chinese software and hardware vendors, e. g. Huawei and Xiaomi, in each one of the databases.

TABLE II. VULNERABILITY DATABASE FEATURES

| *Feature* | *Target Database* |
|---|---|
| Vulnerability Count | NVD, CNVD, CNNVD |
| Missing CPE | NVD, CNVD, CNNVD |
| Missing CVSS | NVD, CNVD, CNNVD |
| Missing Reference | NVD, CNVD, CNNVD |
| Missing CVE. | CNVD, CNNVD |
| Missing Weeks | NVD, CNVD, CNNVD |
| Duplicated CVE | CNVD, CNNVD |
| CVSS version. | NVD, CNVD, CNNVD |
| Missing CWE | NVD, CNVD, CNNVD |

## IV. EXPERIMENTAL RESULTS

This section describes the software and hardware in the test environment. Moreover, it discusses the performance results during the Structuring step. Lastly, it explains the feature results and differences between the national vulnerability databases.

### A. *Test Environment*

Table III describes the software and hardware that comprise the test environment.

TABLE III. HARDWARE AND SOFTWARE

| *Component* | *Description* |
|---|---|
| Host | Ryzen 7 4800h / 16gb RAM |
| OS | Ubuntu 20.04 |
| Database | PostgreSQL 12.9 |
| Language | Python 3.8.10 |
| Data Parsing | lxml 4.7.1, pandas 1.3.5 |
| Parallelization. | Python multiprocessing |

### B. *Performance Results*

The Data Structuring step uses a pool of five workers. They can process in parallel five blocks of data leveraging multicore processors. Each data file is assigned to a worker for data normalization in this approach.

TABLE IV. BATCH PROCESSING

| *Database* | *Format* | *Files* | *Total Size* | *Process Time* |
|---|---|---|---|---|
| NVD | JSON | 23 | 1.5 GB | 266.74s |
| CNVD | XML | 364 | 146.9 MB | 11.33s |
| CNNVD | XML | 25 | 2.4 GB | 48.88s |

The performance results presented in table IV consider the execution time to transform the unstructured XML and JSON to structured data into PostgreSQL. Although there is more data in the CNNVD than in the NVD, this takes more processing time than that because of deeply nested JSON data.

### C. *Vulnerability Databases Comparison*

Table V shows an outline of the three national vulnerability databases.

TABLE V. DATA SUMMARY

| *Feature* | *NVD* | *CNVD* | *CNNVD* |
|---|---|---|---|
| Vulnerability | 178,906 | 99,261 | 180,567 |
| Missing Weeks | 0 | 6 | 0 |
| Missing CVE | 0 | 23,281 | 9,963 |
| Repeated ID | 0 | 88 | 0 |
| Repeated CVE | 0 | 108 | 15 |
| Wrong CVE | 0 | 193 | 0 |
| Missing CVSS | 10,933 | 326 | 8,466 |
| Missing CPE | 11,143 | 113 | 25,694 |
| Missing CWE | 10,928 | 99,261 | 180,567 |
| Missing Reference | 24,035 | 19,012 | 5,786 |

From Table V one can see that the CNNVD has more entries than the NVD, providing a more comprehensive dataset regarding system vulnerabilities. The CNNVD also enables mapping to the NVD using the CVE id to link the two databases. There are only 9,963 entries without CVE id mapping in the CNNVD. Despite fewer entries than the CNNVD, the CNVD has many more vulnerabilities without CVE mapping. Other than that, the previous files from the CNVD are not updated.

At first sight, the CVSS metric seems to be more available in the Chinese databases than in the NVD. Nevertheless, after profiling the 10,933 entries without CVSS score in the NVD, 10,531 has a rejected description. CNAs required those 10,531 CVE ids without assigning any vulnerability. Only 721 vulnerabilities of these 10,531 are stored in the CNNVD, showing that this Chinese database evaluates filtering mechanisms. Besides, the CVSS in the CNVD and CNNVD is a one-column data to the severity metric. There is no score calculation. Moreover, the CNVD uses CVSS version 2 (low, medium, and high) and the CNNVD uses versions 2 and 3 (low, medium, high and critical), following the NVD standard.

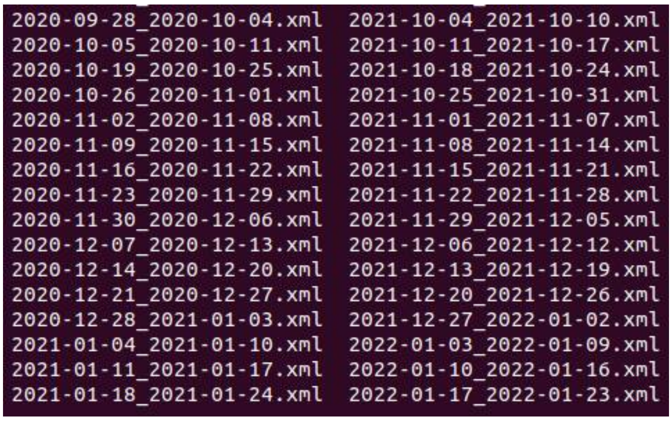
The CWE, critical information to group the hardware and software vulnerabilities, is not embedded in the Chinese databases. Nevertheless, they offer a text defining the type of vulnerability that looks like the CWEs description from the MITRE framework.

The CPE information of the NVD is more comprehensive than that of the CNNVD and CNVD. The earlier covers three configurations: Basic, Running On/With, and Advanced. The Chinese databases do not include the CPE information with this granularity. The CNNVD lists every software version affected by the vulnerability, while the NVD describes logical conditions covering the possibilities of vulnerability occurrence. Furthermore, the CNNVD provides data about the possible solutions with external URLs. The NVD does not offer this solution information in a separate column.

### D. *CNVD Missing Data*

Figure 5 displays a screenshot of the Linux terminal. It shows that the CNVD provides one file for each week. The CNVD data start in 2015.

After downloading the XML files from the CNVD dataset, this work detects six missing files (weeks): 2019-02-11_2019-02-17.xml, 2019-05-20_2019-05-26.xml, 2019-09-23_2019-09-29.xml, 2020-03-02_2020-03-08.xml, 2020-10-12_2020-10-18.xml and 2015-02-16_2015-02-22.xml.



```
2020-09-28_2020-10-04.xml  2021-10-04_2021-10-10.xml
2020-10-05_2020-10-11.xml  2021-10-11_2021-10-17.xml
2020-10-19_2020-10-25.xml  2021-10-18_2021-10-24.xml
2020-10-26_2020-11-01.xml  2021-10-25_2021-10-31.xml
2020-11-02_2020-11-08.xml  2021-11-01_2021-11-07.xml
2020-11-09_2020-11-15.xml  2021-11-08_2021-11-14.xml
2020-11-16_2020-11-22.xml  2021-11-15_2021-11-21.xml
2020-11-23_2020-11-29.xml  2021-11-22_2021-11-28.xml
2020-11-30_2020-12-06.xml  2021-11-29_2021-12-05.xml
2020-12-07_2020-12-13.xml  2021-12-06_2021-12-12.xml
2020-12-14_2020-12-20.xml  2021-12-13_2021-12-19.xml
2020-12-21_2020-12-27.xml  2021-12-20_2021-12-26.xml
2020-12-28_2021-01-03.xml  2021-12-27_2022-01-02.xml
2021-01-04_2021-01-10.xml  2022-01-03_2022-01-09.xml
2021-01-11_2021-01-17.xml  2022-01-10_2022-01-16.xml
2021-01-18_2021-01-24.xml  2022-01-17_2022-01-23.xml
```

Figure 5. CNVD File Samples.

### *E. CNNVD Analysis*

Lastly, this work evaluated three tests with the CNNVD: seek an explanation for unmatched CVEs, compare Chinese vendors in that database with the NVD, and temporal analysis. For the first issue, a complete translation for the English language is a necessary step that was not evaluated by this work. Despite this, this work found 25 vulnerabilities entries of the type “information leakage”, which are not available to the public. They have been published since 2019 and received an id, but they are classified information.

Figure 6 shows vulnerability entries for three famous Chinese vendors, Huawei, ZTE, and Xiaomi, in the CNNVD and NVD.

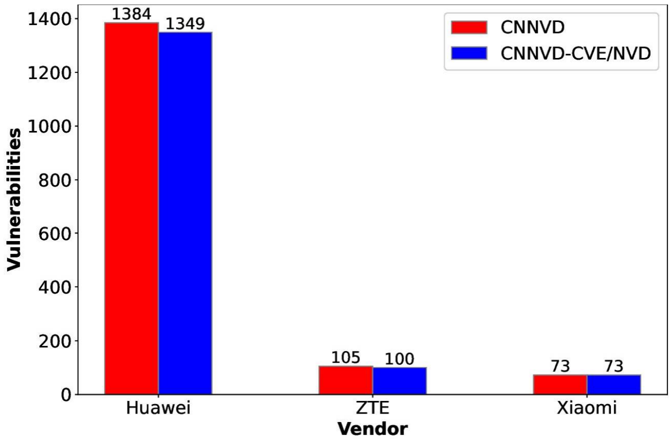


Figure 6. Vendors Comparison.

Figure 6 indicates that the CNNVD has more vulnerability entries regarding Huawei than the NVD, while there is no difference for Xiaomi. By the way, the data of figure 6 come from counting the vulnerabilities with the string “Huawei”, “ZTE” and “Xiaomi” in the description. There may be more entries from these vendors in mandarin. Lastly, figure 7 shows the monthly time series for the NVD and CNNVD.

Figure 7 reveals that the two times series are very similar, with a Pearson correlation of 0.917560. It indicates that they may have been using the same information sources, or maybe they are tracking each other.

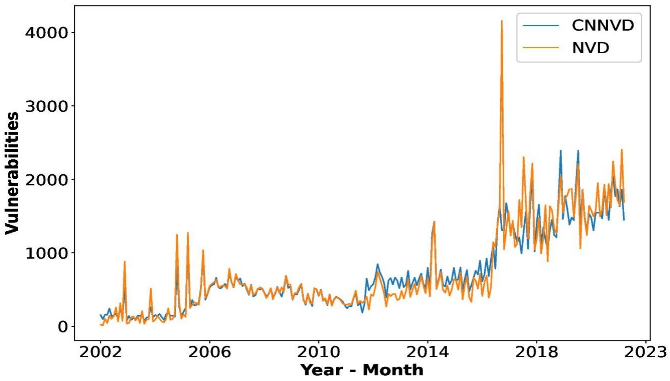


Figure 7. Monthly Time Series.

## V. Conclusions

This work downloaded the three databases simultaneously during the Harvesting Step to enable a fair comparison between them. Moreover, it leverages a Python parallel multiprocessing approach. The CNNVD has 1,661 more vulnerability entries than the NVD, including 25 classified registers. This classification process is compliant with the Chinese legislation. Moreover, the CNNVD contains at least 35 and 5 more Huawei and ZTE vendors’ entries. That database resembles the NVD but has fewer CVSS metrics. Besides, the CNNVD provides an extensive list of vulnerable CPEs but does not offer vulnerable combinations of software like the NVD. The CNNVD also provides richer information regarding external references and possible solutions. Further, a temporal correlation of 0.917560 between the NVD and CNNVD indicates sources in common or that they are tracking each other.

Despite claiming themselves as public XML databases, the CNVD and CNNVD hinder access. Several sign-in issues demand data scrape techniques to download the files. Besides, there are six weeks without data in the CNVD, previous files are not up to date, and its files series began in 2015. It suggests an intention to hide information or process issues. Most of the text data are in mandarin except for URLs, software, and vendor names. Thereby, those two vulnerability databases are turned to the local Chinese community.

Further, as future work, this research intends to translate the entire Chinese databases to English and compare the external references presented by the NVD and CNNVD. Lastly, this work will enable an orchestrated use of those three national vulnerability databases for vulnerability assessment and threat management.

## Acknowledgment

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; the General Attorney’s Office for the National Treasure - PGFN grant 23106.148934/2019-67; the support from EC Horizon 2020 HEROES project grant 101021801. 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). R.d.O.A. thanks Nubank for its support.

## References

[1] M. H. Bejarano, R. J. Rodr´ıguez, and J. Merseguer, “A vision for improving business continuity through cyber-resilience mechanisms and frameworks,” in 2021 16th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1–5, IEEE, 2021.

[2] H. S. Lallie, L. A. Shepherd, J. R. Nurse, A. Erola, G. Epiphaniou, C. Maple, and X. Bellekens, “Cyber security in the age of covid-19: A timeline and analysis of cyber-crime and cyber-attacks during the pandemic,” Computers & Security, vol. 105, p. 102248, 2021.

[3] A. H. Amarullah, A. J. S. Runturambi, and B. Widiawan, “Analyzing cy- ber crimes during covid-19 time in indonesia,” in 2021 3rd International Conference on Computer Communication and the Internet (ICCCI), pp. 78–83, IEEE, 2021.

[4] M. Rytel, A. Felkner, and M. Janiszewski, “Towards a safer internet of things—a survey of iot vulnerability data sources,” Sensors, vol. 20, no. 21, p. 5969, 2020.

[5] M. A. Williams, S. Dey, R. C. Barranco, S. M. Naim, M. S. Hossain, and M. Akbar, “Analyzing evolving trends of vulnerabilities in national vulnerability database,” in 2018 IEEE International Conference on Big Data (Big Data), pp. 3011–3020, IEEE, 2018.

[6] Y. Jia, Y. Qi, H. Shang, R. Jiang, and A. Li, “A practical approach to constructing a knowledge graph for cybersecurity,” Engineering, vol. 4, no. 1, pp. 53–60, 2018.

[7] J. A. Wang and M. Guo, “Ovm: an ontology for vulnerability manage- ment,” in Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies, pp. 1–4, 2009.

[8] J.-b. Gao, B.-w. Zhang, X.-h. Chen, and Z. Luo, “Ontology-based model of network and computer attacks for security assessment,” Journal of Shanghai Jiaotong University (Science), vol. 18, no. 5, pp. 554–562, 2013.

[9] MITRE, “The MITRE Corporation.” https://cve.mitre.org/, 2021. [On- line; accessed 06-December-2021].

[10] NIST, “National Vulerability Database (NVD).” https://nvd.nist.gov/vuln/data-feeds, 2021. [Online; accessed 10-January-2022].

[11] R. Jin and J. Nan, “Combining sources from cve and cnnvd: Data analysis in information security vulnerabilities,” in Journal of Physics: Conference Series, vol. 1800, p. 012004, IOP Publishing, 2021.

[12] G. A. de Oliveira Júnior, R. de Oliveira Albuquerque, C. A. Borges de Andrade, R. T. de Sousa, A. L. Sandoval Orozco, and L. J. García Vil- lalba, “Anonymous real-time analytics monitoring solution for decision making supported by sentiment analysis,” Sensors, vol. 20, no. 16, p. 4557, 2020.

[13] X. Feng, X. Liao, X. Wang, H. Wang, Q. Li, K. Yang, H. Zhu, and L. Sun, “Understanding and securing device vulnerabilities through automated bug report analysis,” in SEC’19: Proceedings of the 28th USENIX Conference on Security Symposium, 2019.

[14] D. Gonzalez, H. Hastings, and M. Mirakhorli, “Automated characterization of software vulnerabilities,” in 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), pp. 135–139, IEEE, 2019.

[15] M. Janiszewski, A. Felkner, P. Lewandowski, M. Rytel, and H. Ro- manowski, “Automatic actionable information processing and trust man- agement towards safer internet of things,” Sensors, vol. 21, no. 13, p. 4359, 2021.

[16] CNCERT/CC, “China National Vulnerability Database (CNVD).” https://www.cnvd.org.cn/shareData/list, 2021. [Online; accessed 10- January-2022].

[17] CNITSEC, “China National Vulnerability Database of Information Se- curity (CNVD).” http://www.cnnvd.org.cn/web/xxk/xmlDown.tag, 2021. [Online; accessed 10-January-2022].

[18] P. Moriuchi and B. Ladd, “China’s Ministry of State Security Likely Influences National Network Vulnerability Publications.” https://www.recordedfuture.com/chinese-mss-vulnerability-influence/, 2021. [Online; accessed 06-December-2021].

[19] Y. Jiang, M. Jeusfeld, and J. Ding, “Evaluating the data inconsistency of open-source vulnerability repositories,” in The 16th International Conference on Availability, Reliability and Security, pp. 1–10, 2021.

[20] Y. Dong, W. Guo, Y. Chen, X. Xing, Y. Zhang, and G. Wang, “To- wards the detection of inconsistencies in public security vulnerability reports,” in 28th USENIX Security Symposium (USENIX Security 19), pp. 869–885, 2019.

[21] K. Kanakogi, H. Washizaki, Y. Fukazawa, S. Ogata, T. Okubo, T. Kato, H. Kanuka, A. Hazeyama, and N. Yoshioka, “Tracing capec attack patterns from cve vulnerability information using natural language processing technique,” in Proceedings of the 54th Hawaii International Conference on System Sciences, p. 6996, 2021.

[22] K. Kanakogi, H. Washizaki, Y. Fukazawa, S. Ogata, T. Okubo, T. Kato, H. Kanuka, A. Hazeyama, and N. Yoshioka, “Tracing cve vulnerability information to capec attack patterns using natural language processing techniques,” Information, vol. 12, no. 8, p. 298, 2021.

[23] I. Tsantilis, T. K. Dasaklis, C. Douligeris, and C. Patsakis, “Searching deterministic chaotic properties in system-wide vulnerability datasets,” in Informatics, vol. 8, p. 86, Multidisciplinary Digital Publishing Insti- tute, 2021.

[24] V. Pham and T. Dang, “Cvexplorer: Multidimensional visualization for common vulnerabilities and exposures,” in 2018 IEEE International Conference on Big Data (Big Data), pp. 1296–1301, IEEE, 2018.

[25] L. G. A. Rodriguez, J. S. Trazzi, V. Fossaluza, R. Campiolo, and D. M. Batista, “Analysis of vulnerability disclosure delays from the national vulnerability database,” in Anais do I Workshop de Seguranc¸a Ciberne´tica em Dispositivos Conectados, SBC, 2018.

[26] A. D. Householder, J. Chrabaszcz, T. Novelly, D. Warren, and J. M. Spring, “Historical analysis of exploit availability timelines,” in 13th USENIX Workshop on Cyber Security Experimentation and Test (CSET 20), 2020.

[27] K. Shi, Y. Dai, and J. Xu, “Construction of a security vulnerability identification system based on machine learning,” Journal of Sensors, vol. 2020, 2020.

[28] E. S. Gualberto, R. T. De Sousa, P. D. B. Thiago, J. P. C. Da Costa, and C. G. Duque, “From feature engineering and topics models to enhanced prediction rates in phishing detection,” Ieee Access, vol. 8, pp. 76368– 76385, 2020.

[29] D. Tovarňák, L. Sadlek, and P. Čeleda, “Graph-based cpe matching for identification of vulnerable asset configurations,” in *2021 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 986–991, 2021.

[30] E. Wåreus and M. Hell, “Automated cpe labeling of cve summaries with machine learning,” in International Conference on Detection of Intru- sions and Malware, and Vulnerability Assessment, pp. 3–22, Springer, 2020.

[31] K. Liu, Y. Zhou, Q. Wang, and X. Zhu, “Vulnerability severity prediction with deep neural network,” in 2019 5th International Conference on Big Data and Information Analytics (BigDIA), pp. 114–119, IEEE, 2019.

[32] N. Crocfer, “CVE Alerting Platform.” https://github.com/opencve/opencve, 2020. [Online; accessed 06-December-2021].