Prioritize Software Vulnerabilities by Classifying based on the CVSS score and Textual Description

Parimi Mastan Rao
Research Associate
Amrita Vishwa Vidyapeetham
Bengaluru, India
pr_mastan@blr.amrita.edu

Prof. Shekar Babu
Professor & Founding Head
Amrita Vishwa Vidyapeetham
Bengaluru, India
sb@amrita.edu

Abstract

Software vulnerabilities are one of the main security risks involved in Information systems. The vulnerabilities might be a path for cyber attackers or hackers to exploit the information systems. Thus, vulnerabilities might lead to loss of data or the ability of the information system to serve. The number of vulnerabilities discovered is increasing. The Information Security manager's key challenges are to fix these vulnerabilities on time before they are exploited. In this paper, to minimize the operational challenges in fixing the vulnerabilities, the authors proposed a new method based on the textual description of the software vulnerabilities with help of data collected from a multinational company with 77,360 vulnerabilities over ten years.

Keywords
Software Vulnerability, CVSS, Vulnerability Management

1. Introduction

The usage of Information systems is increased in business operations. Thus, systems are vulnerable to attacks or exploitation, which might lead to the loss of data present in the systems or not able to serve in the business operations (NVD, 2021). Software vulnerabilities are increasing continuously. Thus increasing the number of software vulnerabilities makes the difficult for the security manager of IT firms to fix vulnerabilities before exploiting them by the attackers.

To understand these losses the information technology firms spent most of the resources on identifying and rectifying the vulnerabilities in devices or the systems of the company network. IT firms are following many methods of identifying vulnerabilities and fixing them. Also with the growth of IT infrastructure, the number of vulnerabilities is increasing. This makes IT firms difficult to prioritize and fix the vulnerabilities based on the impact that would create them.

In July 2005, the National Infrastructure Advisory Council and Forum of Incident Response and Security Team (FIRST) came with Common Vulnerability Scoring System (CVSS) to rate the risk of the vulnerability on a scale of 0 to 10 (FIRST, 2021). The vulnerabilities with ten are very risk and with zero are very low risk. The CVSS is generic and does not take care of system environment parameters. This makes it the IT firms difficult in managing the human resources to fix the vulnerabilities

2. Literature Review

The common vulnerability scoring system (CVSS) version 1 was not subjected to peer review across IT firms, deployed in production. With the feedback of various companies, Mell and K S (2007) improved the CVSS version1 to version 2 with fine-tuned mathematical equations. This CVSS version2 is widely used across IT firms. CVSS developed with quantitative methods of scoring. Dobrovoljc et al. (2017) stated that considering the characteristics of vulnerabilities would improve the ability to the prediction of exploitations of vulnerabilities.

The literature says that very little attention was paid to the textual description in vulnerabilities databases. Bozorgi et al. (2010) were the first research to focus on the textual description and their results showed that texting mining tools would give the information for cybersecurity managers for decision making to fix the vulnerabilities. Hence, the researchers currently have started paying to pay more attention to the textual descriptions of the vulnerabilities. Dobrovoljc et al. (2017) stated that considering the characteristics of vulnerabilities would improve the ability to the prediction of exploitations of vulnerabilities.

Liu & Zhang (2011) based on CVSS score the vulnerabilities classified into three severity levels. High Severity with CVSS score from 7.0 to 10.0, Medium Severity from 4.0 to 6.9, and Low Severity with CVSS score from 0 to 3.9. In the previous work of the author (Parimi & Babu, 2020), the highest number of vulnerabilities were high severity due to the wide range of CVSS scores. Therefore, the authors in this paper by narrowing the CVSS range and grouped the vulnerabilities into ten Priority groups instead of three.

3. Data

In this paper, the authors had collected the software vulnerabilities data from the labs' network with 1336 systems of a multinational company that had 77360 vulnerabilities over time of ten years. Out of 77360, 49145 vulnerabilities are in one year. The data had 35 variables. In this study, the authors had considered QID, Title, CVSS, Impact, and Exploitability variables.

Table 1. Variables in the software vulnerability data that had taken consider in this research.

Sl. No	Variable
1	QID
2	Title
3	CVSS
4	Impact
5	Exploitability

QID is the identification number given to the software vulnerability report. Unique Software vulnerability has a unique QID number. **Title** is the textual description of the software vulnerability. **CVSS** is the Common Vulnerability Scoring System score for a given vulnerability. **Impact** is the textual description of the impact that would be caused by the given software vulnerability if it is exploited. **Exploitability** is the textual explanation of how could be given software vulnerability exploited.

4. Methodology

The CVSS score ranges from 0 to 10. The software vulnerabilities with 0 has less risk and 10 has high risk.Table2 describes the severity ranges based on the CVSS score that had three groups. The majority of vulnerabilities follow in Medium and High severity had 1569 vulnerabilities. Therefore, the author grouped software vulnerabilities into ten priority groups based on the CVSS scores instead of three group and identified very high-risk vulnerabilities with CVSS score range of 8.5 to 10. Table 3 describes the CVSS scores for respective priority groups.

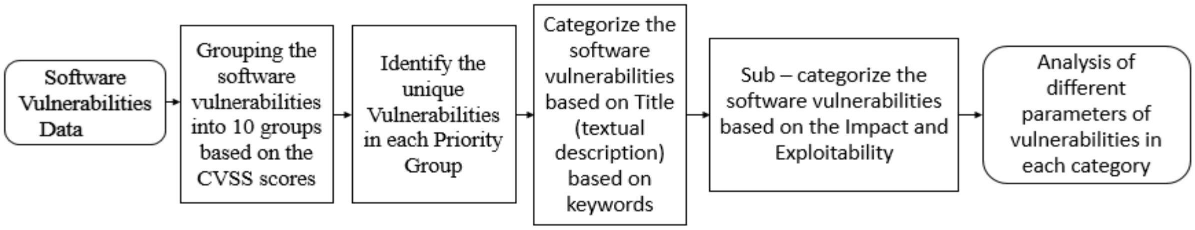
Software Vulnerabilities Data
Grouping the software vulnerabilities into 10 groups based on the CVSS scores
Identify the unique Vulnerabilities in each Priority Group
Categorize the software vulnerabilities based on Title (textual description) based on keywords
Sub – categorize the software vulnerabilities based on the Impact and Exploitability
Analysis of different parameters of vulnerabilities in each category

Figure 1. Methodology Framework

Table 2. Vulnerability severity based on CVSS score.

CVSS Score	Vulnerability severity based on CVSS	Count of Vulnerabilities	Unique Vulnerabilities
7.0 to 10.0	High	1569	96
3.0 to 6.9	Medium	40668	755
0.0 to 2.9	Low	6908	73

Table 3. Grouping the vulnerabilities based on the CVSS scores.

CVSS score	Priority	Count of Vulnerabilities	Unique Vulnerabilities
8.6 to 10.0	1	660	20
7.6 to 8.5	2	276	35
6.6 to 7.5	3	832	66
5.6 to 6.5	4	2349	94
4.6 to 5.5	5	17650	283
3.6 to 4.5	6	11038	220
2.6 to 3.5	7	9834	152
1.6 to 2.5	8	5937	39
0.6 to 1.5	9	15	4
0.0 to 0.5	10	554	11

Manually, the authors had intercept the textual description of 20 unique vulnerabilities in priority one and formulated the keywords. Based on the keywords as mentioned in table 4 classified the software vulnerabilities of priority 1 into seven categories.

Table 4. Keywords used for categorizing Vulnerabilities in the group Priority 1.

Category	Keywords
Malware	Types of Malware such as Shadow Brokers
Sniffing	Sniffing
EOL software	EOL, end of life, Obsolete
Java	Java
Remote Execution / Unauthenticated Users	Remote Execution, Unauthenticated, without Authentication
Login Errors / Password	Login, Password, Credentials, Null Authentication
Information Accessible / writeable	Readable information, writeable information

5. Analysis of Categories in Priority 1 vulnerabilities.

From figure 2, Categories across the priority 1 group vulnerabilities. The Login Errors / Password vulnerabilities present in 482 systems and Information Accessible/writeable vulnerabilities present in 143 systems. 95 % of vulnerabilities follow in Login Errors / Password and Information Accessible/writeable.

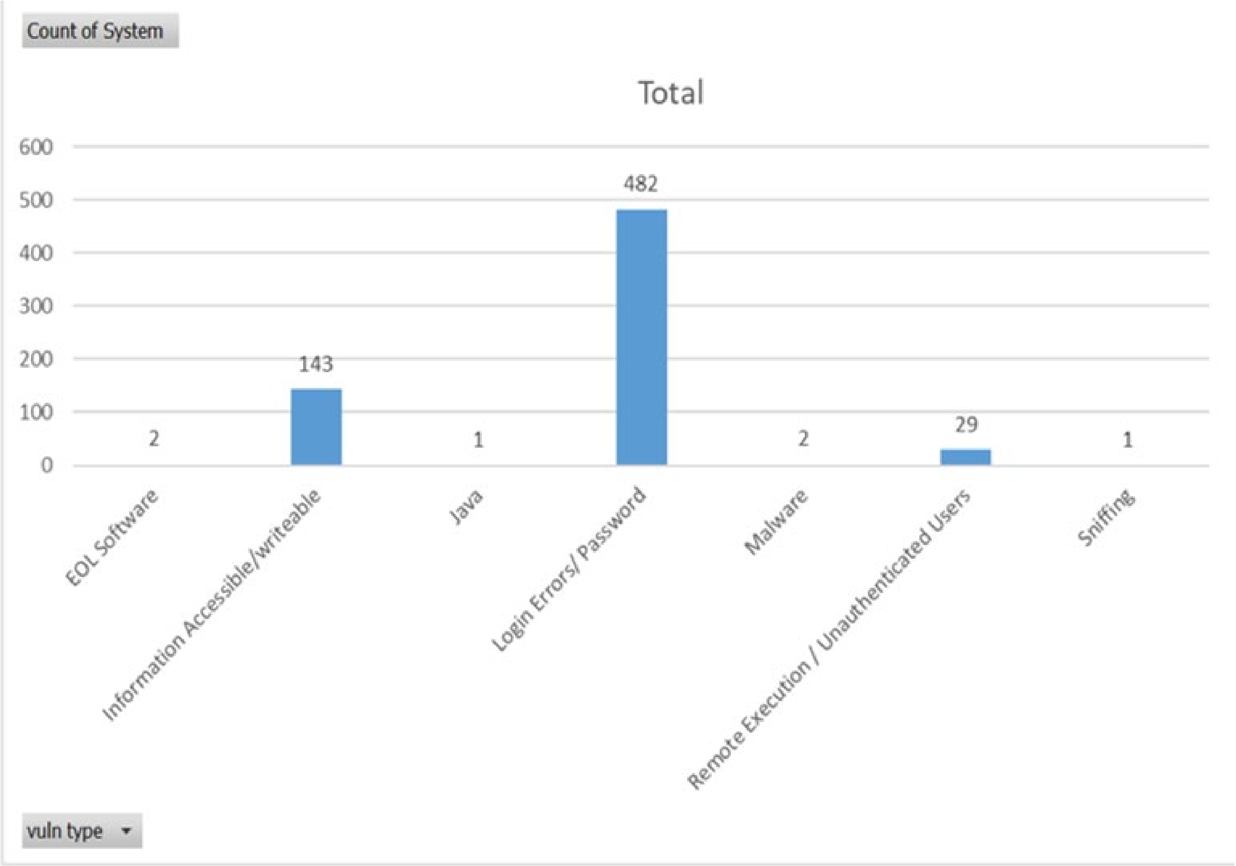

Figure 2. Categories of vulnerabilities in Priority 1 group.

From figure 3, 20 unique vulnerabilities are present in the Priority 1 vulnerability group. Out of 20, 7 vulnerabilities follow in the ‘Remote Execution / Unauthenticated Users’ category, 5 vulnerabilities follow in ‘Login Errors / Password’ category and 4 vulnerabilities follow in the ‘Information Accessible / Writeable’ category.

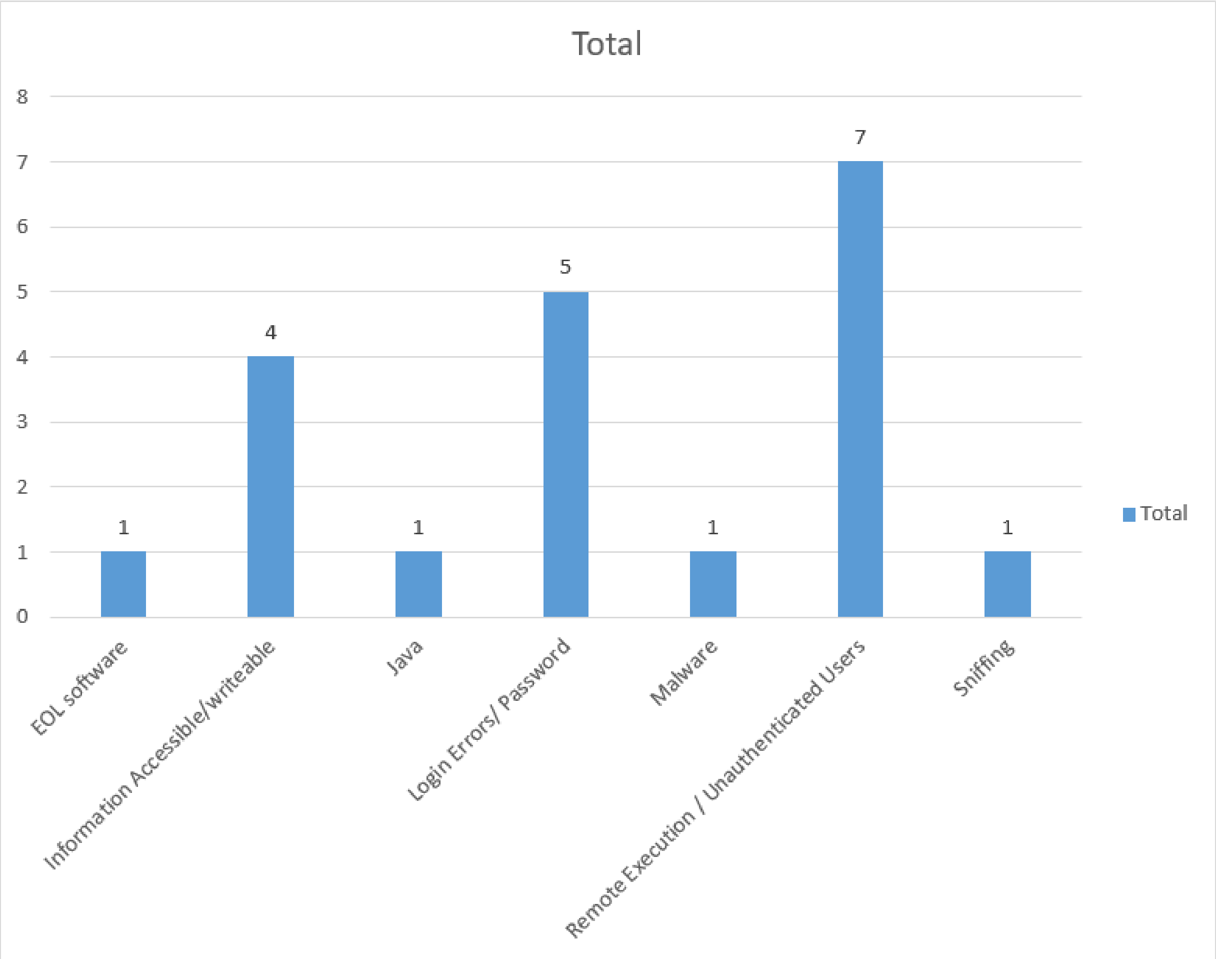

Figure 3. The number of unique vulnerabilities in each category of Priority 1 Vulnerabilities.

From figure 4, 10 out of 20 vulnerabilities lead to gain access to the system and six vulnerabilities lead to run arbitrary code on systems. Form figure 5, the vulnerabilities that lead to gain access to the system present in ‘Information Accessible / Writeable’, ‘Login Errors / Password’ and ‘Remote Execution / Unauthenticated Users’ categories. The vulnerabilities that lead to run Arbitrary Code on the system are present in ‘Information Accessible / Writeable’, ‘Java’, ‘Login Errors / Password’ and ‘Remote Execution / Unauthenticated Users’ categories

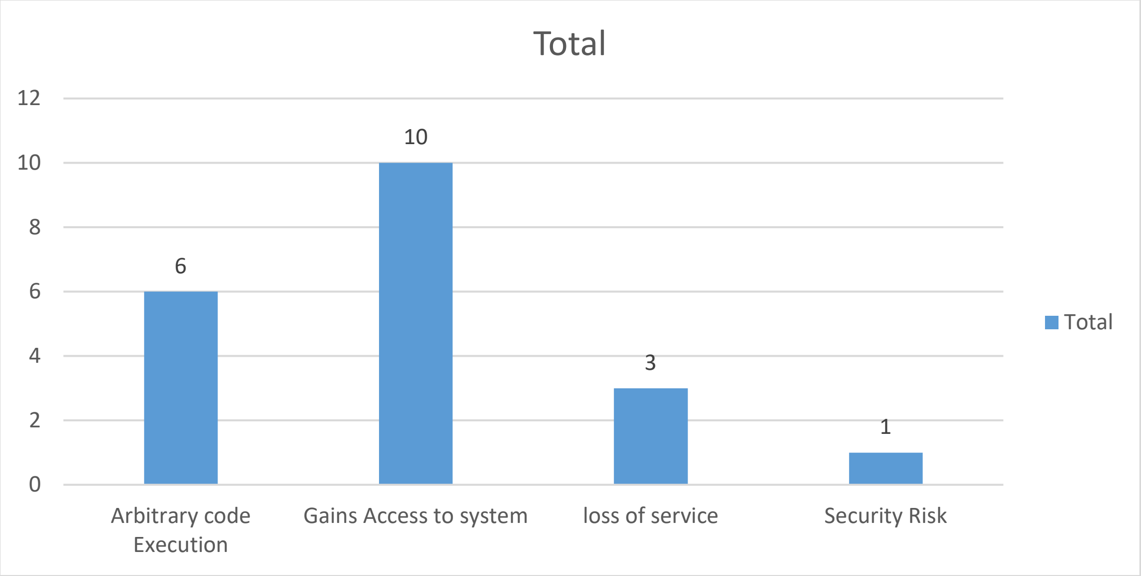

Figure 4. Impact caused by Vulnerabilities.

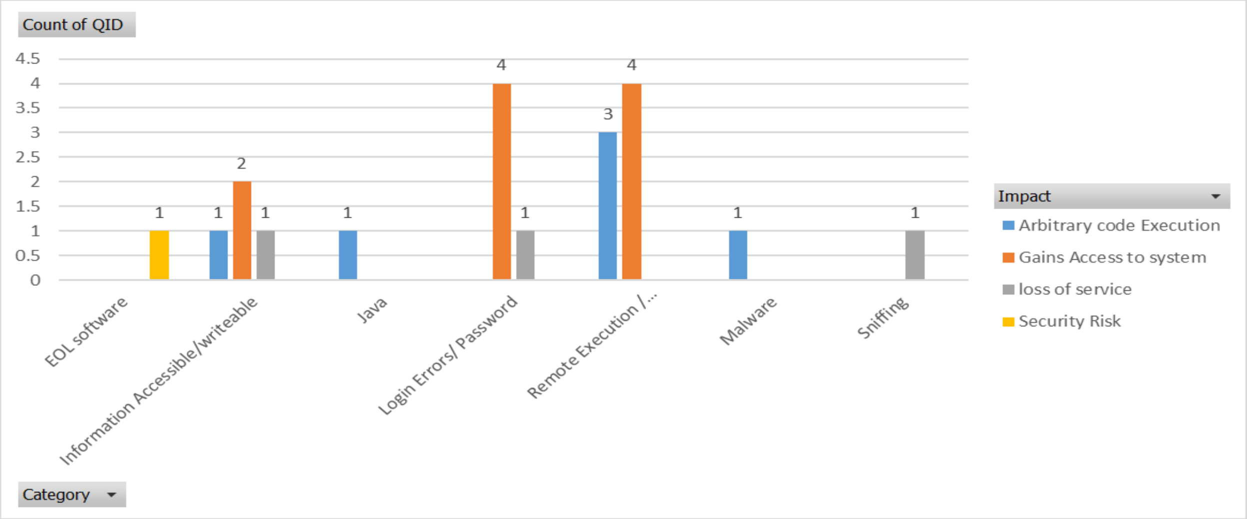

Figure 5. Priority 1 Group vulnerabilities: Categories across Impact.

From figure 6. Four vulnerabilities of twenty could be exploitable by attackers.

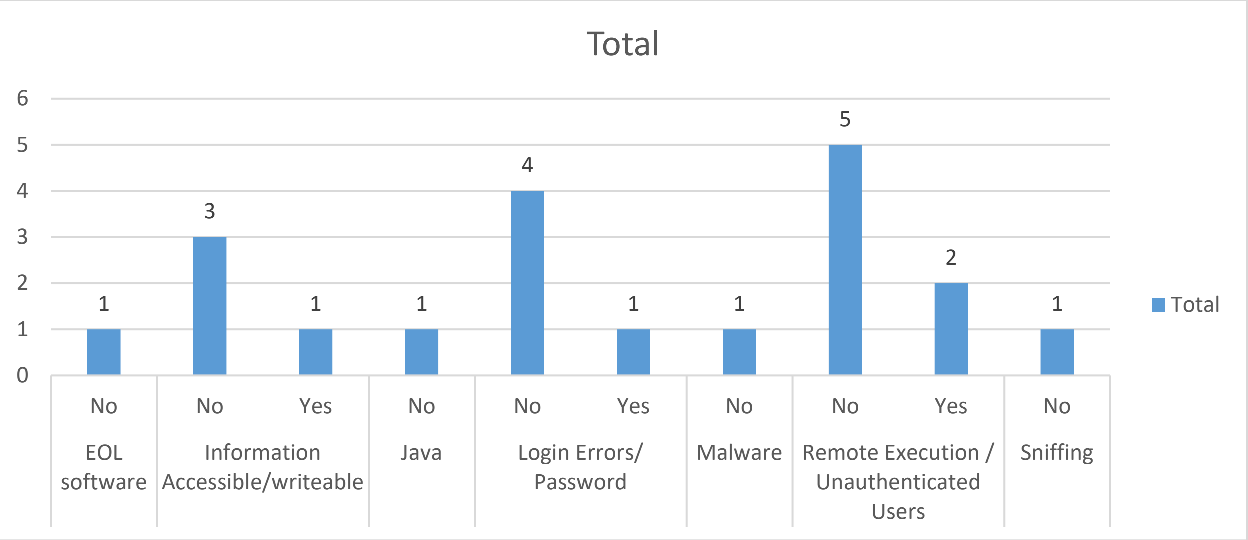

Figure 6. Exploitability of vulnerabilities across Categories in Priority 1 Group.

6. Conclusion

The authors analyzed the vulnerabilities within labs of a multi-national company based on the textual description. They grouped them into priority groups and then manually classified them into various categories based on the textual description of the vulnerability. This analysis narrows down the vulnerability to the top 20 and would help the security managers in efficiently managing the software vulnerabilities.

7. Future Work

This research work could be taken further by automatic the classification process of software vulnerabilities based on the text description and would validate the method with efficiency improvement of managing the software vulnerabilities in the lab.

References

Bozorgi, M., Saul, L., Savage, S., & Voelker, G. (2010). Beyond Heuristics: Learning to Classify Vulnerabilities and Predict Exploits. *KDD,10*.

C.H. Lin, C. C. (2008). A study and implementation of vulnerability assessment and misconfiguration detection. *IEEE Asia-Pacific Service Computing Conference (APSCC 2008), vol. 1–3*, 1252-1257.

Dobrovoljc, A., Trcek, D., & Likar, B. (2017). Predicting Exploitations of Information Systems Vulnerabilities Through Attackers Characteristics. *Digital Object identifier*, 26063-26075.

FIRST. (2021, May). *Common Vulnerability Scoring System*. Retrieved from https://www.first.org/cvss/v1/guide.html

Jerome, D. D., Yapo M, A. O., & Patrice, E. M. (2018, November). Rules-Based Method For Software For Software Vulnerabilities Levels Evaluation Using Machine Learning. *International Journal of Scientific & Engineering Research, 9*(11), 1747 - 1756.

John Chambers, J. T. (2020, Jan). *Common vulnerability scoring system,*. Retrieved from https://www.first.org/cvss/v1/guide.html

Johnson, P., Lagerstrom, R., Ekstedt, M., & Franke, U. (2018, November / December). Can the Common Vulnerability Scoring System be Trusted? A Bayesian Analysis. *IEEE Transactions on Dependable and Secure Computing, 15*(6), 1002 - 1015.

K. Kim, J. K. (2008). hardening windows security configurations,. *International Conference on Convergence and Hybrid Information Technology*, 1252 -1257.

Keramati, M. (2016). New Vulnerability Scoring System for Dynamic Security Evaluation. *International Symposium on Telecommunications*, 746 -751.

Liu, Q., & Zhang, Y. (2011). VRSS: A new system for rating and scoring vulnerabilities. *Computer Communications*, 264 - 273.

Mark J. Cox, Classification of security issues. (n.d.). Retrieved from http://www.redhat.com/f/pdf/rhel4/SecurityClassification

MSRC, Microsoft security response center security bulletin severity rating system. (n.d.). Retrieved from https://www.microsoft.com/technet/security/bulletin/rating.mspx

NVD. (2021). Retrieved from http://nvd.nist.gov/statistics.cfm

Parimi, M., & Babu, S. (2020). Critical Analysis of Software Vulnerabilities through Data Analytics. *International Conference on Industrial Engineering and Operations Management*, 923-934.

Peter Mell, K. S. (2006). Common Vulnerability Scoring System Security & Privacy. *IEEE*, 85 - 89.

Peter Mell, K. S. (2007). Improving the Common Vulnerability Scoring System. *Information Security IET*, 119 - 127.

Peter Mell, K. S. (2020, January). *A complete guide to the common vulnerability scoring system version 2.0,*. Retrieved from http://www.first.org/cvss/cvss-guide.pdf

Biographies

Parimi Mastan Rao is a research associate at "Amrita Center for Responsible Innovations and Sustainable Enterprises", "ARISE" Labs within Amrita Vishwa Vidyapeetham, Bangalore campus. Mr. Rao holds a Bachelor of Technology in Electronics and Communication Engineering from Vignan's Foundation for Science, Technology & Research (VFSTR) University, Master of Business Administration (MBA) in Marketing Management from Amrita Vishwa Vidyapeetham and Master of Science (MS) in Business Analytics and Systems from The State University of New York (SUNY) at Buffalo. He also holds Postgraduate Diploma (PGD) in Data Science from Manipal Academy of Higher Education (Manipal University). His research interests include Business Analytics, Data-driven business Solutions, and Artificial Intelligence in Management.

Shekar Babu Ph.D. is the Professor and Director of "AMRITA Center for Responsible Innovations and Sustainable Enterprises", "ARISE" Labs. He is also the Founding Head, Department of Management (DoM), Bangalore Campus, AMRITA Vishwa Vidyapeetham University, Bangalore, India. Dr. Shekar holds a Bachelor of Engineering (BE) degree in Electronics and Communications from Bangalore University and a Master of Science (MS) degree in Electrical and Computer Science from California State University, Los Angeles and a Doctoral Degree in Strategic Management from Amrita University. He is a Management Consultant with over 25 years of experience in working at Price Waterhouse, Hewlett-Packard Co and AMRITA University. His research areas are Corporate Social Responsibility (CSR), Corporate Governance (CG), Strategy and Social Development and Sustainable Goals (SDG). He has taught courses in Marketing, Leadership, Management Consulting and Business Ethics and Values.