Improved Prediction of Vulnerability Exploitation using CVSS Base-Score with Optimized Equation Parameters

Ben Church

**Abstract**—

**Index Terms**—Vulnerability, CVSS, base score, exploit, prediction, optimized

————————————————

1. B. Church is with the School of Computing, Queen’s University, Kingston, ON, Canada. E-mail: [ben.church@queensu.ca](mailto:ben.church@queensu.ca)

—————————— ◆ ——————————

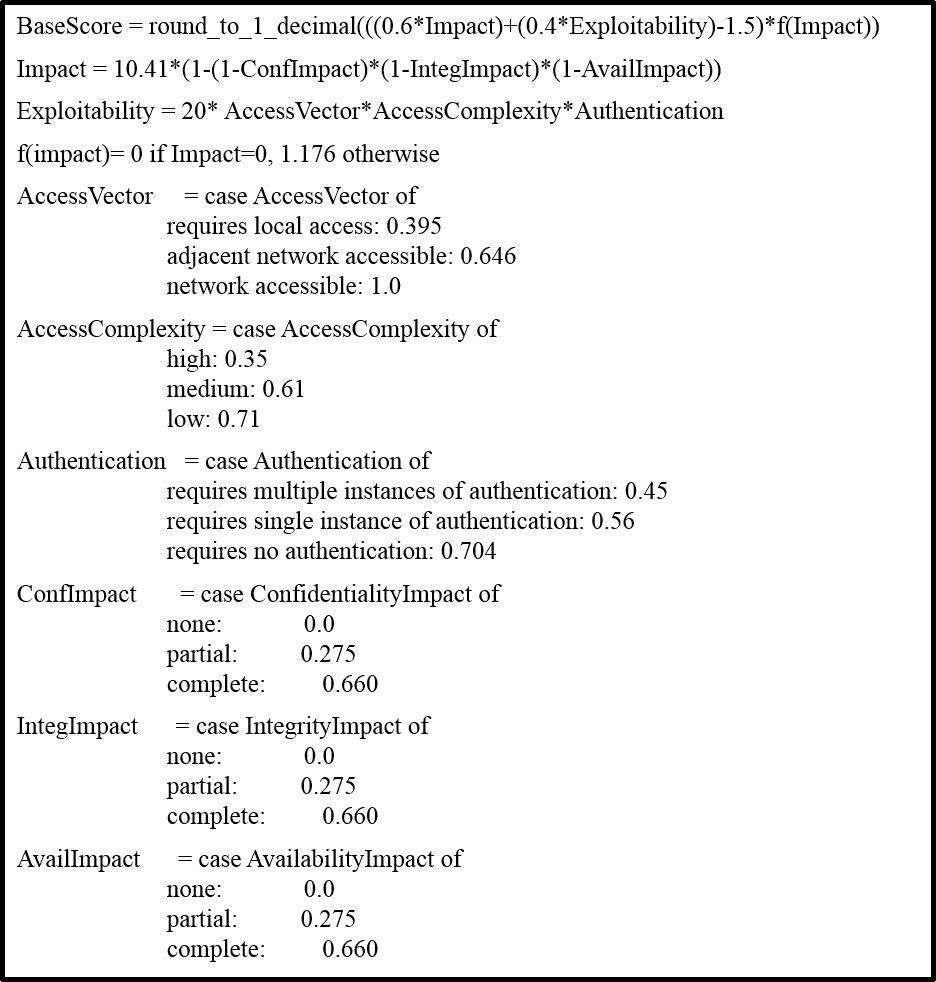
# 1 Introduction

S

oftware vulnerability addressment is an important problem for organizations since exploitation of such vulnerabilities causes preventable losses. This problem tends to grow with time because vulnerabilities are often reported more frequently than they can be fixed. [Find example from VRP?] To minimize losses in spite of the increasing number of vulnerabilities becoming known, and with limited available resources, organizations must prioritize their vulnerability addressment strategy. Organizations minimize real losses by prioritizing vulnerabilities based on the cost each one is expected to impart if left unaddressed. Probabilistically, this expected cost, commonly called risk, is equal to the real cost in the event of exploitation, times the probability of that exploitation occurring.

The Common Vulnerability Scoring System (CVSS) was created to provide a single, objective measure of the risk presented by any software vulnerability [Scarfone2007]. The CVSS offers three scores in the range of 0-10, measuring the risk presented by the vulnerability. They are the base, temporal, and environmental scores. The base score coveys the intrinsic vulnerability risk, or time and context invariant risk, by considering six mostly objective vulnerability metrics, which will be examined shortly. The temporal and environmental scores essentially provide context specific re-evaluations of the base score. The temporal score takes into consideration changes in public or expert vulnerability knowledge, or vulnerability patch deployment, one might achieve a more accurate measure of the risk currently posed by a vulnerability. The environmental score considers the distribution of the vulnerability in the computer system or organization of interest, as well as the subjective importances of various computer system services which could be compromised. Because of their (mostly) objective and unchanging nature, CVSS base scores are widely available from vulnerability databases and are often used in research.

The complete formula for the CVSS base score equation v2 is shown in Figure 1. To the end of providing an objective scoring system, most of the metrics in the equation are objective and where possible, quantitative. Variation in a particular vulnerabilities scoring will not come from subjectivity in whether zero, one, or multiple instances of user authentication are required for exploitation. Likewise, whether a system’s data confidentiality is not at all compromised, partially compromised, or totally compromised, is objective and quantitative. A notable exception to the objective metrics is the AccessComplexity metric, which can be high, medium, or low. The CVSS has in fact received criticisms of subjectivity [Khazaei2016], and been the subject of several critical evaluation studies.

Figure : CVSS base score equation, v2 [Scarfone2007]

Bozorgi *et al.* [Bozorgi2010] and Khazaei *et al.* [Khazaei2016] proposed machine learning methods for automated base score prediction using additional information from vulnerability database entries. They proposed these methods in response to questions of the ability of the CVSS base score to predict actual exploitation of vulnerabilities. While machine learning may be useful for correlating vulnerability information with base scores, or even predicting exploitation, it probably cannot be used to achieve the single scoring system intended by the CVSS. If machine learning was integrated into the CVSS, different users of the system could not take advantage of online learning from new vulnerabilities. System specific online learning will cause the system’s model to change, potentially scoring a vulnerability differently than another system. Otherwise, it cannot improve itself from new information. As such, the need remains for a single, standardized vulnerability scoring system.

Allodi and Massacci [Allodi2014] examined the risk assessment capabilities of the CVSS base score by retrospectively correlating base scores with both proof-of-concept and real black market exploits. To measure exploit prediction, they computed confusion matrices where the CVSS base score’s relation to a threshold serves as the predictor for exploitability, and appearance of the vulnerability in Symantec’s AttackSignature[[1]](#footnote-1) and ThreatExplorer[[2]](#footnote-2) datasets as true exploitation in the wild. They measured prediction system’s risk reduction as:

(1)

Where P(e|f) denotes the probability of event e given event f, EV is the event that vulnerability V is exploited (appearing in Symantec’s data, in this case), score(V) is V’s CVSS base score, and T is some threshold value. They found that using CVSS base scores alone provided an optimal risk reduction of about 0.15 at a threshold of 2, little better than guessing.

Younis *et al.* [Younis2015] performed a study on the effectiveness of the CVSS base score and Microsoft’s vulnerability rating systems for predicting vulnerability exploitation. Similarly to Allodi and Massacci, they correlated predictions based on CVSS base scores and Microsoft system scores with real exploitation as determined by the existence of Exploit Database[[3]](#footnote-3) (EDB) entries. Both systems performed similarly, in that they were both sensitive enough to detect most exploitable vulnerabilities, but they had false positive rates above 90% for vulnerabilities found in Internet Explorer.

The failure of the CVSS to reliably predict exploitation in the wild is illustrated both in literature such as [Allodi2014, Younis2015], and by the adoption of organization-specific vulnerability rewards programs (VRPs) such as Mozilla’s[[4]](#footnote-4) or Google’s[[5]](#footnote-5). Nonetheless, the system’s open availability, the amount of published research, and the official recognition from organizations such as the National Institute of Standards and Technology under the Security Content Automation Protocol[[6]](#footnote-6) (SCAP) make the CVSS an attractive starting point for the development of a better scoring system. Improvements in the CVSS’ ability to predict vulnerability exploitation might make it a viable alternative to VRPs.

The work presented in the remainder of this paper explores the possibility of improving CVSS exploit prediction by optimizing the scalar parameters in its base score equation (Figure 1). Details about the optimization process including data metrics used for objective functions, are given in the Methods section. Prediction system performance metrics are presented in the Results section. Finally, implications and limitations of the results, as well as potential improvements are covered in the Discussion & Conclusions section.

# 2 Methods

## 2.1 Data

To optimize the CVSS for exploit prediction, both vulnerability exploit existence and base score data were required. The National Vulnerability Database[[7]](#footnote-7) (NVD) was used as the source of this data. The NVD was chosen for the prospect of a large, unbiased set of vulnerability entries, by virtue of its 80,000 plus entries and governmental operation. Only vulnerability entries from between the years 2010 and 2016 (inclusive) were chosen for this study. This was to avoid including increasingly dated and therefore irrelevant software and vulnerabilities, while allowing a chance for vulnerabilities to be reported by not including everything up to the present. From this chronologic range on vulnerabilities, samples of 5000 were selected randomly.

Vulnerabilities were classified into ‘exploited’ or ‘unexploited’ based on whether the NVD entry had a reference to the EDB. The six element base score vectors, containing the vulnerability’s metric cases, rather than the final numerical score, were extracted from the database entries so base scores, and the resulting classification performance metrics could be recomputed to drive the optimization.

## 2.2 Classification assessment

Performance metrics conveying the utility of the CVSS base score in predicting vulnerability exploitation were required before optimization could begin. To compare the effectiveness of the CVSS and corporation specific VRPs, Younis *et al.* [Younis2016] computed prediction confusion matrices for the scoring systems. A general confusion matrix is shown in Table 1

Table : Prediction system confusion matrix structure

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Actually exploited? | |
| True | False |
| Exploit  predicted? | True | True  positive (TP) | False  positive (FP) |
| False | False  negative (FN) | True  negative (TN) |

Sensitivity, precision, an ICC, and various combinations of them were computed and used for optimization of the base score equation parameters. Definitions for the metrics used for optimization are given in equations

(2)

(3)

(4)

Thus, sensitivity is the ratio of exploits successfully predicted, to the number of exploits which should have been predicted. Precision is the system’s ability to predict without over-predicting. The ICC, as defined in equation 4, is ratio of the variability which class x has from the mean of population A to the total variability experienced by the class, variability within itself, and between itself and the grand mean. This simple ICC definition conveys the distinctness a prediction system imparts on its predictions by classification. However, since both the ‘exploited’ and ‘unexploited’ classes have ICCs, they were combined in weighted averages for optimization. The objective function using ICCs was simply 1 – ICCavg.

Following the example by Younis *et al.* [Younis2016], sensitivity, precision, and the ICC were used to assess CVSS prediction performance. Performance metrics were collected before and after equation optimization for improvement assessment. Measures taken to enable optimization are explored next, before performance results are reported.

## 2.3 Optimization

Sensitivity and precision, as they are formulated above, cannot be used for an optimization objective function. Small changes in base score equation parameters occurring during optimization result in no change in sensitivity or precision. This is because the classification system is discrete, binary, small changes in the parameters will not usually push a vulnerability’s base score past the classification threshold. Therefore there is no gradient for the optimization. To remedy this, mean over-prediciton error (TOE) and mean under-prediction error (TUE) were introduced as shown in equations 5 and 6.

(5)

(6)

where T is the score threshold for classification. TOE and TUE provide a continuous variable measures of a system’s over and under prediction errors, respectively. For optimization, they were averaged and normalized over the range 0-1 into the unit over-prediction error (UOE) and unit under-prediction error (UUE) shown in equations 7 and 8.

(7)

(8)

where numx is the number of members in class x. These continuous, normalized measures of classification error were combined with sensitivity and precision into two F-measures, F1 and F2, shown in equations 9 and 10.

(9)

(10)

F1 and F2 were defined this way so better system performance is indicated by values closer to 0, since the optimization algorithm used minimization. Like the two class’ ICCs, the F-measures were simply combined by weighted averaged for the final optimization objective function value.

# 3 Results

Table 2 shows the performance metrics for the CVSS scoring system on a random sample of 10,000 vulnerabilities before and after optimization with both the ICC and F-measure objective functions.

Table : Performance metrics before and after both optimization methods for a sample of 10,000 vulnerabilities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Optimization** | **Metric** | | | |
| Sens. | Prec. | ICCUnexpl. | ICCExpl. |
| Unoptimized | 0.20 | 0.05 | 0.50 | 0.54 |
| ICC |  |  |  |  |
| F-measure | 0.97 | 0.07 | 0.50 | 0.57 |

Tables 3, 4, and 5 show the CVSS base score exploit prediction system’s performance metrics. Table 2 shows them before optimization. Table 3 shows them after optimizing using 1 – ICCavg as an optimization objective function. Table 4 shows metrics from after optimizing using the F-measure as an objective function.

Table : CVSS base score performance metrics before equation optimization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample #** | **Metric** | | | |
| Sens. | Prec. | ICCUnexpl. | ICCExpl. |
| 1 | 0.20 | 0.05 | 0.50 | 0.54 |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| **Average** |  |  |  |  |
| **Standard deviation** |  |  |  |  |

Table : CVSS base score performance metrics after optimization using ICCs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample #** | **Metric** | | | |
| Sens. | Prec. | ICCUnexpl. | ICCExpl. |
| 1 | 0.95 | 0.08 | 0.50 | 0.78 |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| **Average** |  |  |  |  |
| **Standard deviation** |  |  |  |  |

Table : CVSS base score performance metrics after optimization using F-measure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample #** | **Metric** | | | |
| Sens. | Prec. | ICCUnexpl. | ICCExpl. |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| **Average** |  |  |  |  |
| **Standard deviation** |  |  |  |  |

# 4 Discussion & Conclusions

IEEE Computer Society style is to note citations in individual brackets, followed by a comma, e.g. “[1], [5]” (as opposed to the more common “[1, 5]” form.) Citation ranges should be formatted as follows: [1], [2], [3], [4] (as opposed to [1]-[4], which is not IEEE Computer Society style). When citing a section in a book, please give the relevant page numbers [2]. In sentences, refer simply to the reference number, as in [3]. Do not use “Ref. [3]” or “reference [3]” At the beginning of a sentence use the author names instead of “Reference [3],” e.g., “Smith and Smith [3] show ... .” Please note that references will be formatted by IEEE Computer Society production staff in the same order provided by the author.

# 5 Equations

If you are using Word, use the MathType add-on (<http://www.mathtype.com>) for equations in your paper (Insert | Object | Create New | Microsoft Equation or MathType Equation). “Float over text” should not be selected.

For display equations as seen below, number equations consecutively with equation numbers in parentheses flush with the right margin, as in (1). First, use the equation editor to create the equation. Then, select the “Equation” markup style. Press the tab key and write the equation number in parentheses. To make your equations more compact, you may use the solidus ( / ), the exp function, or appropriate exponents. Use parentheses to avoid ambiguities in denominators. Punctuate equations when they are part of a sentence, as in

 (1)

Be sure that the symbols in your equation have been defined before the equation appears or immediately following. Italicize symbols (*T* might refer to temperature, but T is the unit tesla). Per IEEE Computer Society, please refer to “(1),” not “Eq. (1)” or “equation (1),” except at the beginning of a sentence: “Equation (1) shows ... .” Also see *The Handbook of Writing for the Mathematical Sciences*, 1993. Published by the Society for Industrial and Applied Mathematics, this handbook provides some helpful information about math typography and other stylistic matters. For further information about typesetting mathematical equations, please visit the IEEE Computer Society style guide: <http://www.computer.org/portal/web/publications/style_math>.

Please note that math equations might need to be reformatted from the original submission for page layout reasons. This includes the possibility that some in-line equations will be made display equations to create better flow in a paragraph. If display equations do not fit in the two-column format, they will also be reformatted. Authors are strongly encouraged to ensure that equations fit in the given column width.

# 6 Helpful Hints

## 6.1 Figures and Tables

Because IEEE Computer Society staff will do the final formatting of your paper, some figures may have to be moved from where they appeared in the original submission. Figures and tables should be sized as they are to appear in print. Figures or tables not correctly sized will be returned to the author for reformatting.

Detailed information about the creation and submission of images for articles can be found at <http://www.computer.org/portal/web/peerreviewjournals/author#figures> where you can [View transactions art\_guide.pdf (PDF, 4.69MB)](http://www.computer.org/cms/Computer.org/Journal%20templates/transactions_art_guide.pdf) . We strongly encourage authors to carefully review the material posted here to avoid problems with incorrect files or poorly formatted graphics.

Place figure captions below the figures; place table titles above the tables. Figure captions appear as left justified. Table captions are restricted to one sentence and are formatted as title case. Any additional sentence in a table caption will be formatted as a footnote below the table (see Table 1 in this document). If your figure has two parts, include the labels “(a)” and “(b)” as part of the artwork. Please verify that the figures and tables you mention in the text actually exist. Figures and tables should be called out in sequential order, as this is how they will be placed in your paper. For example, avoid referring to figure “8” in the first paragraph of the article unless figure 8 will again be referred to after the reference to figure 7. **Please do not include figure captions as part of the figure. Do not put captions in “text boxes” linked to the figures. Do not put borders around the outside of your figures.** Per IEEE Computer Society, please use the abbreviation “Fig.” even at the beginning of a sentence. Do not abbreviate “Table.” Tables are numbered numerically.

For journals that use print for publication, please verify with IEEE Computer Society that the journal you are submitting to does indeed accept color before submitting final materials. **Do not use color unless it is necessary for the proper interpretation of your figures.**

Figures (graphs, charts, drawing or tables) should be named fig1.eps, fig2.ps, etc. If your figure has multiple parts, please submit as a single figure. Please do not give them descriptive names. Author photograph files should be named after the author’s LAST name. Please avoid naming files with the author’s first name or an abbreviated version of either name to avoid confusion. If a graphic is to appear in print as black and white, it should be saved and submitted as a black and white file (grayscale or bitmap.) If a graphic is to appear in color, it should be submitted as an RGB color file.



Fig. 1. Magnetization as a function of applied field. Note that “Fig.” is abbreviated. There is a period after the figure number, followed by one space. It is good practice to briefly explain the significance of the figure in the caption.

Figure axis labels are often a source of confusion. Use words rather than symbols. As an example, write the quantity “Magnetization,” or “Magnetization *M*,” not just “*M*.” Put units in parentheses. Do not label axes only with units. As in Fig. 1, for example, write “Magnetization (A/m)” or “Magnetization (Am−1),” not just “A/m.” Do not label axes with a ratio of quantities and units. For example, write “Temperature (K),” not “Temperature/K.” Table 1 shows some examples of units of measure.

Multipliers can be especially confusing. Write “Magnetization (kA/m)” or “Magnetization (103 A/m).” Do not write “Magnetization (A/m) × 1,000” because the reader would not know whether the top axis label in Fig. 1 meant 16,000 A/m or 0.016 A/m. Figure labels should be legible, approximately 8 to 12 point type. When creating your graphics, especially in complex graphs and charts, please ensure that line weights are thick enough that when reproduced at print size, they will still be legible. We suggest at least 1 point.

## 6.3 Footnotes

Number footnotes separately in superscripts (Insert | Footnote)[[8]](#footnote-8). Place the actual footnote at the bottom of the column in which it is cited; do not put footnotes in the reference list (endnotes). Use letters for table footnotes (see Table 1). Please do not include footnotes in the abstract and avoid using a footnote in the first column of the article. This will cause it to appear above the affiliation box, making the layout look confusing.

TABLE 1  
Units for Magnetic Properties



Statements that serve as captions for the entire table do not need footnote letters.

aGaussian units are the same as cgs emu for magnetostatics; Mx = maxwell, G = gauss, Oe = oersted; Wb = weber, V = volt, s = second, T = tesla, m = meter, A = ampere, J = joule, kg = kilogram, H = henry.

## 6.4 Lists

The IEEE Computer Society style is to create displayed lists if the number of items in the list is longer than three. For example, within the text lists would appear 1) using a number, 2) followed by a close parenthesis. However, longer lists will be formatted so that:

1. Items will be set outside of the paragraphs.
2. Items will be punctuated as sentences where it is appropriate.
3. Items will be numbered, followed by a period.

## 6.5 Theorems and Proofs

Theorems and related structures, such as axioms corollaries, and lemmas, are formatted using a hanging indent paragraph. They begin with a title and are followed by the text, in italics.

**Theorem 1.** *Theorems, corollaries, lemmas, and related structures follow this format. They do not need to be numbered, but are generally numbered sequentially.*

Proofs are formatted using the same hanging indent format. However, they are not italicized.

**Proof.** The same format should be used for structures such as remarks, examples, and solutions (though these would not have a Q.E.D. box at the end as a proof does). 

# 7 End Sections

## Appendices

Appendices, if present, appear online as supplemental material. In the event multiple appendices are required, they will be labeled “Appendix A,” “Appendix B, “ etc.

IEEE Computer Society Transactions accepts supplemental materials for review with regular paper submissions. These materials may be published on our Digital Library with the electronic version of the paper and are available for free to Digital Library visitors. Please see our guidelines below for file specifications and information. Any submitted materials that do not follow these specifications will not be accepted. All materials must follow US copyright guidelines and may not include material previously copyrighted by another author, organization or company. More information can be found at <http://www.computer.org/portal/web/peerreviewjournals/author#supplemental>.

## 7.2 Acknowledgments

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” Sponsor and financial support acknowledgments are included in the acknowledgment section. For example: This work was supported in part by the US Department of Commerce under Grant BS123456 (sponsor and financial support acknowledgment goes here). Researchers that contributed information or assistance to the article should also be acknowledged in this section. Also, if corresponding authorship is noted in your paper it will be placed in the acknowledgment section. Note that the acknowledgment section is placed at the end of the paper before the reference section.

## 7.3 References

Unfortunately, the Computer Society document translator cannot handle automatic endnotes in Word; therefore, type the reference list at the end of the paper using the “References” style. See the IEEE Computer Society’s style for reference formatting at: <http://www.computer.org/portal/web/publications/style_refs>. The order in which the references are submitted in the manuscript is the order they will appear in the final paper, i.e., references submitted nonalphabetized will remain that way.

Please note that the references at the end of this document are in the preferred referencing style. Within the text, use “et al.” when referencing a source with more than three authors. In the reference section, give all authors’ names; do not use “et al.” Do not place a space between an authors' initials. Papers that have not been published should be cited as “unpublished” [4]. Papers that have been submitted or accepted for publication should be cited as “submitted for publication” [5]. Please give affiliations and addresses for personal communications [6].

Capitalize all the words in a paper title. For papers published in journals not published in English, please give the English citation first, followed by the original foreign-language citation [7].

## 7.4 Additional Formatting and Style Resources

Additional information on formatting and style issues can be obtained in the IEEE Computer Society Style Guide, which is posted online at: <http://www.computer.org/portal/web/publications/styleguide>. Click on the appropriate topic under the Special Sections link.

# Conclusion

Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions. Authors are strongly encouraged not to reference multiple figures or tables in the conclusion—these should be referenced in the body of the paper.

**Acknowledgment**

The authors wish to thank A, B, C. This work was supported in part by a grant from XYZ.

**References**

[Scarfone2007] - Mell P, Scarfone K, and Romanosky S. “A complete guide to the common vulnerability scoring system version 2.0” Published by FIRST-Forum of Incident Response and Security Teams. 2007.

[Khazaei2016] - Khazeai A, Ghasemzadeh M, and Derhami V. “An automatic method for CVSS score prediction using vulnerabilities description” Journal of Intelligent & Fuzzy Systems. 2016; 30:89-96.

[Bozorgi2010] - Bozorgi M. Lawrence KS, Savage S, and Voelker GM. 2010 “Beyond Heuristics: Learning to Classify Vulnerabilities and Predict Exploits” Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. 2010; 105-114.

[Allodi2014] - Allodi L, and Massacci F. “Comparing Vulnerability Severity and Exploits Using Case-Control Studies” ACM Transactions on Information and System Security. 2014; 17(1):1- 20.

[Younis2015] - Younis AA, and Malaiya YK. “Comparing and Evaluating CVSS Base Metrics and Microsoft Rating System” IEEE International Conference on Software Quality, Reliability and Security. 2015; 252-261.

[NVD] - https://nvd.nist.gov/

**First A. Author** Allbiographies should be limited to one paragraph consisting of the following: sequentially ordered list of degrees, including years achieved; sequentially ordered places of employ concluding with current employment; association with any official journals or conferences; major professional and/or academic achievements, i.e., best paper awards, research grants, etc.; any publication information (number of papers and titles of books published); current research interests; association with any professional associations. Author membership information, e.g., is a member of the IEEE and the IEEE Computer Society, if applicable, is noted at the end of the biography.

**Second B. Author Jr.** biography appears here.

**Third C. Author** biography appears here.

1. <http://www.symantec.com/security_response/attacksignatures/> [↑](#footnote-ref-1)
2. <http://www.symantec.com/security_response/threatexplorer/> [↑](#footnote-ref-2)
3. <https://www.exploit-db.com> [↑](#footnote-ref-3)
4. <https://www.mozilla.org/en-US/security/bug-bounty/> [↑](#footnote-ref-4)
5. <https://www.google.com/about/appsecurity/reward-program/> [↑](#footnote-ref-5)
6. <https://scap.nist.gov/index.html> [↑](#footnote-ref-6)
7. <https://nvd.nist.gov/> [↑](#footnote-ref-7)
8. It is recommended that footnotes be avoided (except for the unnumbered footnote with the receipt date on the first page). Instead, try to integrate the footnote information into the text. [↑](#footnote-ref-8)