Risk assessment method for IoT software supply chain vulnerabilities

Zhicheng Zhu*, Kun Lan, Zhihong Rao and Yuguang Zhang
China Electronics Technology Cyber Security Co., Ltd., Chengdu, China

*Corresponding author e-mail: zhuzcwork@126.com

Abstract. With the development of IoT technology, the number of attacks against IoT software chain vulnerabilities is greater than ever, and a reasonable vulnerability assessment system needs to be established for research. The Common Vulnerability Scoring System (CVSS) is a free, public risk assessment system used by information security vendors to assess the severity of vulnerability threats. However, CVSS is insufficient because of the strong subjectivity in the selection of measurement standards and the allocation of evaluation index weights. Based on this reason, this paper designs and proposes a more objective risk assessment method for IoT software chain vulnerabilities, and verifies the feasibility and effectiveness of the method through experiments.

1. Introduction

From the first mention of the Internet of Things concept in 1995 by Bill Gates to today, the Internet of Things has become a typical representative of the development of a new generation of information and communication technologies, which is profoundly changing the traditional industrial form and human production and lifestyle. As an emerging product in the information technology era, IoT is quietly entering millions of households at a rapid development rate. According to Gartner statistics, the number of IoT devices is expected to increase to approximately 20.8 billion by 2020, and IoT will be transmitted on the Internet. The data will exceed 20 ZB.

With the global development of the Internet of Things technology and the globalization of its software supply chain, the design, research and development, and operation of IoT devices are increasingly dependent on a globally interconnected supply chain ecosystem. The supply chain ecosystem including manufacturers, suppliers (network vendors and software and hardware vendors), system integrators, purchasers, end users and external service providers, and other entities, product and service design, research and development, production, distribution, deployment, and use, and software environments such as technology, law, and policy.

Simply put, the basic structure of the IoT software supply chain includes the entire process of internal development, information, information systems, services, components, and product (or service) manufacturing maintenance and exit from the information system. Because it runs through multiple supply and demand links, involving multiple types of entities such as manufacturers, suppliers, system integrators, service providers, and software environments such as technology, laws, and policies, it is already the basis of other supply chains.

Due to the global distribution of the IoT software supply chain, the structure of the network chain, purchasers, and users, the risk control capabilities of the IoT supply chain are gradually reduced as the transparency of the supplier decreases. Malicious functions, data leakage, and key product...
interruptions exist in any link or service provision will destroy the continuity of related business and bring uncontrollable security risks.

Nowadays, IoT companies are increasingly using open source software and the security risks caused by the homogeneous and heterogeneous nature of the IoT itself. The number of attacks against IoT software chain vulnerabilities is more than ever. Therefore, it is necessary to conduct in-depth research on the classification and evaluation of IoT software chain vulnerabilities

2. CVSS assessment method

In the vulnerability classification system, in addition to the vulnerability classification research, how to quantitatively evaluate the harm caused by various types of vulnerabilities to the entire information system is also one of the research hotspots. Vulnerability hazard assessment can classify vulnerabilities from another dimension, such as high-risk vulnerabilities, medium-risk vulnerabilities, and low-risk vulnerabilities. It is an indispensable and important attribute of the vulnerability classification system. Currently, there are four types of vulnerability hazard assessment methods commonly used in the world: CVSS, DREAD, owasp, and SRC ratings. Among them, CVSS is the most widely used.

To address the confusion of vulnerability ratings, the National Infrastructure Advisory Committee (NIAC) has proposed the CVSS public standard, which is maintained by the FIRST organization. CVSS aims to rank information system vulnerabilities and provide end users with a comprehensive score that represents the severity and risk of the vulnerability.

CVSS uses a modular scoring system, which includes three parts: the Base Metric Group, the Temporal Metric Group, and the Environmental Metric Group.

![Three indicators of CVSS](image)

Figure 1. Three indicators of CVSS.

The scores calculated from these three parts constitute a total score of 0 to 10, with 10 being the worst.

The basic process of vulnerability level assessment is as follows:

1) Analyze the various indicators of the vulnerability, and get the value corresponding to each indicator;
2) Bring the quantified score corresponding to each indicator value into the vulnerability risk assessment algorithm to obtain the vulnerability score;
3) According to the vulnerability score, compare the vulnerability classification table to get the vulnerability level.
Figure 2. CVSS evaluation process.

The vulnerability submitter analyzes the vulnerability attributes according to the indicators. The calculated vulnerability score can be divided into the following levels: low risk (0.1 ~ 3.9), medium risk (4.0 ~ 6.9), high risk (7.0 ~ 10.0)

However, in the classification and evaluation of IoT software chain vulnerabilities, the shortcomings of CVSS are also obvious:

1) Excessive weight allocation of basic evaluation indicators depends on subjective decision-making by experts and scholars, and the reproducibility of measurement standards is poor.
2) The same weight is assigned to the impact of confidentiality, availability, and integrity in the basic evaluation indicators, and differences in internal attributes of vulnerabilities with similar ratings cannot be reflected.

3. Improved CVSS assessment method

Although CVSS has designed corresponding mathematical formulas to calculate the three-level scores of vulnerability threats, but the key is how to assign vulnerability assessment metrics and calculate the basis.

In practical applications, evaluation of multiple decision-making indicators is needed, so how to determine the weight of each indicator is important. The method proposed in this paper is aiming to solve this problem.

The implementation steps of the evaluation index weight are as follows:

1) Formally represent the randomly generated data as a matrix \( Z = (Z_1, Z_2, ..., Z_n)^T \), and:
\[
Z = \begin{pmatrix}
z_1(1) & z_1(2) & \cdots & z_1(m) \\
z_2(1) & z_2(2) & \cdots & z_2(m) \\
\vdots & \vdots & \ddots & \vdots \\
z_n(1) & z_n(2) & \cdots & z_n(m)
\end{pmatrix}
\]

2) Give reference data columns. The composition weight of the reference data column \( Z_0 \) is the value with the largest weight in \( Z \), and
\[
Z_0 = (z_0(1), z_0(2), ..., z_0(m))
\]

3) Calculate the distance between \( n \) evaluation index sequences \( Z_1, Z_2, ..., Z_n \) and \( Z_0 \)
\[
D_{oi} = \sum_{k=1}^{m}(z_0(k) - z_i(k))^2
\]

4) Calculate the weights of \( n \) evaluation indexes.
\[
W_i = \frac{1}{1+D_{oi}}
\]

4. Experimental process and analysis

The overall design and implementation steps of the experimental in this paper are as follows:

1) Select a sample of the experimental data set. In the China National Vulnerability Database of Information Security (CNNVD), 200 samples of IoT software supply chain vulnerabilities collected in the last five years were randomly selected.
(2) Each sample randomly generates 10 index weighting schemes according to the rules. According to the CVSS weight distribution rules, when the confidentiality, integrity, and availability of the system are not damaged, the weight value is set to 0; the weight that causes the completion impact is set to 2 times the weight of the partial impact. From the above conditions, 5 sets of weight distribution schemes that meet the rules are randomly generated. The results are shown in the following table.

| No | Confidentiality | Integrity | Usability |
|----|----------------|-----------|-----------|
| 1  | 1.5            | 1         | 0.7       |
| 2  | 1.9            | 1.1       | 0.9       |
| 3  | 1.7            | 1.2       | 0.9       |
| 4  | 1.4            | 1         | 1.1       |
| 5  | 1.6            | 1.3       | 0.6       |

(3) The sample data is used to statistically analyze the 5 index weighting schemes obtained, and the optimal weighting scheme is calculated. Apply step (2) to calculate the reference sequence $Z_0$. From the above table, we know that the maximum value of the weight is 1.7, so $Z_0 = \{1.7,1.7,1.7,1.7,1.7\}$

The formula (1) can calculate the evaluation index weight matrix, that is,

$$Z = Z_0 \cdot Z_1 \cdot Z_2 \cdot Z_3$$

Bring the sequence $Z_0, Z_1, Z_2, Z_3$ into the formula (2) (4) to calculate the optimized weight set $W = \{w_1, w_2, w_3\} = \{1.4, 1.3, 0.6\}$

(4) Through the above steps, an optimized basic index weight distribution scheme is obtained, and then frequency analysis is performed on 200 randomly sampled CNNVD vulnerabilities. The statistical results are shown in the following table.

| CVSS and new method statistical results. |
|-----------------------------------------|
| Mean | Minimum | Median | Maximum | Standard Deviation | Number of Different Points |
|------|---------|--------|---------|--------------------|---------------------------|
| CVSS | 6.94    | 0      | 6.4     | 10                 | 2.37                      | 61                        |
| New Method | 5.79 | 0      | 5.5     | 9.5                | 2.41                      | 88                        |

The experimental results show that the score distribution of the proposed method is smoother and continuous than CVSS, which effectively avoids the occurrence of excessive extreme values.

5. Conclusion

This paper proposes an improved vulnerability risk assessment method. The proposed method optimizes the allocation according to the importance of the scoring elements, making the results more efficient. And it is easier to distinguish between different IoT software chain vulnerabilities with similar ratings.
6. Acknowledgments
This work was financially supported by Vulnerability Analysis and Security Enhancement of Internet of Things Software Chain for Complex Battlefield Environment (No.61403120502) fund.

References
[1] Igure V M, Williams R D. Taxonomies of attacks and vulnerabilities in computer systems[M]. IEEE Press, 2008.
[2] Stergiopoulos G, Katsaros P, Gritzalis D. Execution Path Classification for Vulnerability Analysis and Detection[C]// International Conference on E-Business and Telecommunications. Springer International Publishing, 2016.
[3] Davari M, Zulkernine M, Jaafar F. An Automatic Software Vulnerability Classification Framework[C]// 2017 International Conference on Software Security and Assurance (ICSSA). IEEE Computer Society, 2017.
[4] Luszcz, Jeff. How maverick developers can create risk in the software and IoT supply chain[J]. Network Security, 2017, 2017(8):5-7.
[5] Yang K, Forte D, Tehranipoor M M. Protecting Endpoint Devices in IoT Supply Chain[J]. 2015.
[6] Hadar N, Siboni S, Elovici Y. A Lightweight Vulnerability Mitigation Framework for IoT Devices[C]// Workshop. 2017.
[7] Ko E, Kim T, Kim H. A Study on the Management Structure of Vulnerability Information in IoT Environment[M]// Advanced Multimedia and Ubiquitous Engineering. 2017. Assessing the Performance of a Classification-Based Vulnerability Analysis Model[J]. Risk Analysis, 2014, 35(9).