CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Security Analysis of Popular MQTT Broker Platforms

Enhancing MQTT broker security for resilient IoT communication

Master's Thesis in Computer science and engineering

Reshad Qurishi
Zhiyuan Zhang

Departments of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2025

Master's Thesis 2025

Security Analysis of Popular MQTT Broker Platforms

Enhancing MQTT broker security for resilient IoT communication

Reshad Qurishi
Zhiyuan Zhang

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2025

Security Analysis of MQTT Broker platforms

Reshad Qurishi Zhiyuan Zhang

Supervisor: Srijita Basu, Department of Computer Science and Engineering
Examiner: Chih-Hong Cheng, Department of Computer Science and Engineering

Master's Thesis 2025
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2025

Reshad Qurishi, Zhiyuan Zhang
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

The Message Queuing Telemetry Transport (MQTT) protocol has emerged as a fundamental communication mechanism in Internet of Things (IoT) environments due to its lightweight and efficient publish-subscribe architecture. However, its widespread adoption has introduced significant security challenges, particularly within MQTT broker platforms. This thesis aims to analyze popular MQTT broker platforms for known vulnerabilities and prioritize them based on severity and impact to enhance the overall security posture of these systems. The study investigates the evolution of vulnerabilities over time, examining their type, frequency, and severity, through data collected from Snyk and publicly available databases such as the National Vulnerability Database (NVD). To inform secure design decisions, the thesis compares the vulnerability distribution across the major MQTT broker platforms. Although over 70 broker platforms exist, the analysis focuses on accessible open-source platforms including EMQX, VerneMQ, HiveMQ, Mosca, and Eclipse Mosquitto. Special attention is given to architectural design choices and third-party dependencies that contribute to security risks. A key contribution of this study is the development of an improved vulnerability scoring model that addresses the limitations of the Common Vulnerability Scoring System (CVSS). Unlike CVSS, the proposed model incorporates additional context-aware metrics such as frequency, i.e., how often a vulnerability appears in the NVD and popularity, i.e., how many broker platforms are affected by the same vulnerability within a given timeframe. The findings aim to support organizations and developers in strengthening IoT infrastructures by enabling more resilient, secure, and context-aware vulnerability management strategies.

Keywords: Broker platform, Common Vulnerability Scoring System (CVSS), MQTT protocol, Vulnerability.

Acknowledgements

We thank our examiner, Chih-Hong Cheng, for his valuable feedback and challenging us to refine our ideas. We also thank our supervisor, Srijita Basu, for her insightful feedback on both content and structure. We express our gratitude to all those who supported and guided us throughout the course of this thesis. Finally, we thank Chalmers for making this thesis possible.

Reshad Qurishi, Zhiyuan Zhang, Gothenburg, June 2025

Contents

List of Figures **xiii**

List of Tables **xv**

Terminology **1**

1 Introduction **4**
1.1 Goals and Purpose 4
1.2 Problem Description 5
1.3 Research Question 5
1.3.1 How have vulnerabilities in MQTT broker platforms evolved over time, and what trends can be observed in their types, frequency, and severity? 5
1.3.2 What are the critical differences in vulnerability distribution across major MQTT broker implementations, and how can these differences inform more secure design choices? 5
1.3.3 How are the vulnerabilities in MQTT broker platforms related to the underlying protocol architecture and the third-party libraries they integrate? 6
1.3.4 What metrics significantly impact vulnerability prioritization and how can we design a better model that reflects the security of MQTT? 6
1.4 Thesis Outline 6

2 Background **7**
2.1 IoT communications 7
2.1.1 Comparative Analysis of IoT Communication Protocols 9
2.2 MQTT Broker platforms 11
2.3 Challenges 13
2.4 Limitations and Delimitations 13
2.4.1 Delimitation 13
2.4.2 Limitations 13
2.4.3 Limitations of The Improved CVSS Scoring System 14

3 Related Work **15**
3.1 Review of Existing Research 15
3.2 Significance of the Study 16

4 Theory **17**
4.1 Literature Review . . . 17
4.2 Vulnerability Scanning Tools . . . 18
4.3 Metrics of CVSS . . . 20
4.3.1 Calculation of CVSS . . . 22
4.3.1.1 Base Score . . . 22
4.3.1.2 Environmental Score . . . 23
4.4 CVSS Evaluation . . . 25

5 Methods **26**
5.1 Methodology . . . 26
5.1.1 Evaluation research . . . 26
5.1.2 Empirical study . . . 27
5.2 Data Collection . . . 28
5.2.1 Data from NVD . . . 28
5.2.2 Data from Snyk generated reports . . . 29

6 Results **31**
6.1 RQ1: How have vulnerabilities in MQTT broker platforms evolved over time and what trends can be observed in their types, frequency, and severity? . . . 31
6.2 RQ2: What are the critical differences in the vulnerability distribution across the major MQTT broker implementations and how can these differences inform more secure design choices? . . . 34
6.2.1 Distributions of Vulnerabilities from NVD . . . 34
6.2.2 Vulnerability Distribution in Snyk Generated Reports . . . 35
6.3 RQ3: How are the vulnerabilities in MQTT broker platforms related to the underlying protocol architecture and the third-party libraries they integrate? . . . 37
6.4 RQ4: What metrics significantly impact vulnerability prioritization and how can we design a better model that reflects the security of MQTT? . . . 39
6.4.1 Designing an Improved Model to Reflect the Security of MQTT? 39
6.4.1.1 Severity . . . 42
6.4.1.2 Exploitability . . . 43
6.4.1.3 Fixability . . . 45
6.4.1.4 Repeatability . . . 46
6.4.2 Key Differences Between the New Model and CVSS . . . 48
6.4.3 Calculating Vulnerability Score with proposed Vulnerability Scoring Model . . . 51
6.4.3.1 Path Traversal . . . 51
6.4.3.2 BufferOverflow . . . 54
6.4.3.3 Double Free . . . 56
6.4.3.4 Hard Coded Credentials . . . 58
6.4.3.5 Evaluation of improved scoring model . . . 60

7 Discussion **62**

7.1 Key Findings 62
7.2 Improved Vulnerability Scoring Model 63
7.3 Future Work 63

8 Conclusion **64**

Bibliography **65**

A Appendix 1 - AI Usage **I**

B Appendix 2 Report of Snyk **II**

C Appendix 3 Source Code of MQTT Brokers and Version Information **IV**

D Appendix 4 Appearance of Path Traversal in NVD **V**

E Appendix 5 Appearance of Buffer Overflow in NVD **XVI**

F Appendix 6 Appearance of Double Free in NVD **XXXIV**

G Appendix 7 Appearance of HardCoded Credentials in NVD **XXXV**

H Appendix 8 Calculation Program and source code **XXXVI**

List of Figures

2.1 Different messaging protocols and their operability on UDP and TCP [31] . . . 8
2.2 Message size and overhead comparison of different messaging protocols [31] . . . 9
2.3 Reliability and interoperability of different messaging protocols [31] . 10
2.4 Illustration of publish-subscribe communication of MQTT broker platform. The figure shows a publisher publishing the temperature "temp" and subscribers who subscribed the temperature and receiving them. 11

4.1 Different sets of metrics that contributes to final score of CVSS. . . . 20
4.2 Security Requipment and Modified Base Metrics in Environmental Metrics of CVSS . . . 21

5.1 Result of searching on NVD with key word "Mosquitto". The figure shows vulnerabilities related to Mosquitto broker platform. . . . 28
5.2 Step by step setup and configuration of Snyk. . . . 29
5.3 Result of scanning Mosquitto broker platform by Snyk . . . 29
5.4 A specific issue snyk found in a broker platform and recommended step to mitigate the risk . . . 30

6.1 Distribution of vulnerabilities found in NVD, across five broker platforms. Rectangles represent broker names, circles represent vulnerabilities and the CWE inside circles represent types of vulnerabilities. . 34
6.2 Distribution of vulnerabilities from Snyk generated report. Rectangles represent broker names, circles represent vulnerabilities. . . . 35
6.3 Metrics of improved scoring system . . . 41
6.4 Replaced Scope in CVSS 4.0 [15] . . . 42
6.5 Exploit Maturity and its setting value [12] . . . 45
6.6 Visualization of Six significant changes listed in the table 6.16 . . . 49
6.7 Path Traversal in mosquitto, and where in the source code Path Traversal could be exploited . . . 51
6.8 Souce code of Mosquitto broker platform where it leads to buffer overflow. . . . 54
6.9 NVD page for Buffer Overflow official patches . . . 55
6.10 Mosquitto source Code that leads to Double Free vulnerability . . . 56
6.11 Patch of CVE-2022-49541[41] . . . 57
6.12 Mosquitto source code that contains hard coded credentials[41]. . . . 58

6.13 Hard coded credentials and fix suggested by Snyk. 59

B.1 Result of running Static analysis on broker platform II
B.2 Result of running static analysis on Mosca broker platform. II
B.3 Result of running static analysis on EMQX broker platform. III

D.1 First 9 of Path Traversal V
D.2 Second 10 of Path Traversal VI
D.3 Third 10 of Path Traversal VII
D.4 Fourth 10 of Path Traversal VIII
D.5 Fifth 10 of Path Traversal IX
D.6 Sixth 10 of Path Traversal X
D.7 Seventh 10 of Path Traversal XI
D.8 Eighth 9 of Path Traversal XII
D.9 Nineth 11 of Path Traversal XIII
D.10 Tenth 10 of Path Traversal XIV
D.11 11th 9 of Path Traversal XV

E.1 First 9 of Buffer Overflow XVI
E.2 Second 6 of Buffer Overflow XVII
E.3 Third 10 of Buffer Overflow XVII
E.4 Fourth 9 of Buffer Overflow XVIII
E.5 Fifth 1 of Buffer Overflow XVIII
E.6 Sixth 6 of Buffer Overflow XIX
E.7 Seventh 10 of Buffer Overflow XX
E.8 Eighth 4 of Buffer Overflow XX
E.9 Ninth 9 of Buffer Overflow XXI
E.10 10th 11 of Buffer Overflow XXII
E.11 11th 10 of Buffer Overflow XXIII
E.12 12th 10 of Buffer Overflow XXIV
E.13 13th 11 of Buffer Overflow XXV
E.14 14th 9 of Buffer Overflow XXVI
E.15 15th 10 of Buffer Overflow XXVII
E.16 16th 10 of Buffer Overflow XXVIII
E.17 17th 11 of Buffer Overflow XXIX
E.18 18th 9 of Buffer Overflow XXX
E.19 19th 11 of Buffer Overflow XXXI
E.20 20th 9 of Buffer Overflow XXXII
E.21 21th 8 of Buffer Overflow XXXIII

F.1 First 5 of Double Free XXXIV

G.1 First 3 of HardCoded Credentials XXXV

List of Tables

2.1 Comparison of MQTT quality of Service (QoS) Levels and example in where it is typically used. 12

4.1 Comparison of different Vulnerability Scanning Tools for MQTT Brokers 19
4.2 Vulnerability levels of CVSS and their numerical value. [13] 21

6.1 Observed vulnerability trends across phases in MQTT Brokers. 33
6.2 Vulnerability distribution between platforms, their types and root cause. 36
6.3 Summary of some vulnerabilities in MQTT brokers and the third-party library they originate from. 38
6.4 Vulnerability levels of Improved Scoring System and numerical value. 41
6.5 Scoring of Confidentiality (C) 43
6.6 Scoring of Integrity (I) 43
6.7 Scoring of Availability (A) 43
6.8 Scoring of Scope (S) 43
6.9 Attack vector(AV) metric and its four options according to CVSS [13] 44
6.10 Values of PR 44
6.11 Values of AV 44
6.12 Values of UI 44
6.13 Scoring of Remediation Levels 46
6.14 Scoring of Frequency 47
6.15 Scoring of Popularty 47
6.16 Limitations of CVSS and improvements made in the proposed model 48
6.17 Setting values of CVE-2019-10743, in[34], CVSS Version 3x 52
6.18 Setting values of CVE-2022-45918, in[33], CVSS Version 3x 52
6.19 Exploitability and Severity Metric Setting 53
6.20 Fixability and Repearability Metric Setting 53
6.21 CVE-2018-1000300 Metric Setting [32],CVSS version 3x 55
6.22 Exploitability and Severity Setting of Buffer Overflow 56
6.23 Fixability and Repearability Metric Setting 56
6.24 Metrics setting of CVE-2022-49541 in CVSS [35],CVSS Version 3x . . 58
6.25 Metric setting in Improved Scoring System 58
6.26 Metrics setting of CVE-2024-41794 in CVSS [36],CVSS Version 3x . . 59
6.27 Metric setting in Improved Scoring System 59
6.28 Scores in CVSS and Improved Scoring System 60

C.1 Source Code of MQTT Brokers and Version Information IV

Terminology

Term	Description
CVSS (Common Vulnerability scoring system)	A quantitative vulnerability scoring model that ranges from zero to ten. Higher scores represents higher risk of vulnerabilities and the most dangerous level is critical.
CVE (Common Vulnerabilities and Exposures)	A public catalog of known cybersecurity vulnerabilities. Each entry contains an ID, a description, and references.
CWE (Common Weakness Enumeration)	A classification framework for identifying potential software and hardware weakness that might lead to vulnerabilities.
NVD (National Vulnerability Database)	An extensive repository recording known vulnerabilities.
Internet of Things (IoT)	In IoT, physical devices are linked together to transmit data within a shared network.
MQTT Broker	Messages from publishers are received by the MQTT broker, which validates and delivers them to the correct subscribers.
infrastructure-as-code (IaC)	Automating IT infrastructure management by writing code rather than manual configuration.

CWE-ID and corresponding vulnerability type [7]

CWE-ID	**Full Name**
CWE-319	Cleartext Transmission of Sensitive Information
CWE-327	Use of a Broken or Risky Cryptographic Algorithm
CWE-476	NULL Pointer Dereference
CWE-23	Relative Path Traversal
CWE-122	Heap-based Buffer Overflow
CWE-190	Integer Overflow or Wraparound
CWE-415	Double Free
CWE-798	Use of Hard-coded Credentials
CWE-416	Use After Free
CWE-170	Improper Null Termination
CWE-290	Authentication Bypass by Spoofing
CWE-611	Improper Restriction of XML External Entity Reference
CWE-916	Use of Password Hash With Insufficient Computational Effort
CWE-22	Improper Limitation of a Pathname to a Restricted Directory
CWE-770	Allocation of Resources Without Limits or Throttling
CWE-284	Improper Access Control
CWE-20	Improper Input Validation
CWE-502	Deserialization of Untrusted Data
CWE-79	Cross-site Scripting
CWE-617	Reachable Assertion
CWE-276	ncorrect Default Permissions
CWE-400	ncontrolled Resource Consumption

List of Metric Abbreviations [13, 16]

Metrics	**Abbreviations**
Attack Vector	AV
Attack Complexity	AC
Privileges Required	PR
User Interaction	UI
Scope	S
Confifentiality	C
Integrity	I
Availability	A
Exploit Code Maturity	ECM
Report Confidence	RC
Security Requirements	SR
Modified Base Metrics	MC, MI, MA, MS, MAC, MUI, MAV , shown in Figure 4.2
Confidentiality Requirement	SR
Availability Requirement	AR
Integrity Requirement	IR
Remediation	RL
Popularity	P
Frequency	F

1

Introduction

The growth of IoT has turned the concept of connected smart devices into a real phenomenon, seen in everything from home appliances to electric cars. At the center of this evolution lies the Message Queuing Telemetry Transport protocol (MQTT), well known for its reliability and lightweight structure. However, this messaging protocol comes with significant security issues. Since MQTT has emerged as a critical hub for IoT communication, it has become simultaneously an attractive target for cyberattacks, as it offers limited built-in security. Recent incidents involving IoT devices relying on MQTT have exposed these limitations, raising questions about security architecture and risk assessment tools. One such widely adopted risk assessment tool is the Common Vulnerability Scoring System (CVSS), which works well in many contexts but does not account for contextual factors unique to MQTT broker platforms, such as spatial distribution (i.e., the spread of vulnerabilities across different platforms) and frequency (i.e., how often vulnerabilities are exploited).
This thesis analyzes the security posture of five widely used open-source MQTT broker platforms, identifying vulnerabilities and their types, origin, severity, and trends. Based on this analysis, we propose an improved risk assessment model that incorporates additional dimensions to address the CVSS's limitations in the MQTT-specific context, facilitating accurate and MQTT-specific vulnerability prioritization.
Our objective through this thesis is to outline a path for improvement and provide insight to help developers and organizations strengthen the backbone of the IoT ecosystem through a secure MQTT-based infrastructure.

1.1 Goals and Purpose

The goal of this study is to analyze MQTT broker platforms for vulnerabilities and prioritize them based on severity and impact. In addition, the goal is to evaluate the effectiveness of CVSS and propose improvements.

The purpose of this research is to enhance the security of MQTT broker platforms by conducting an in-depth vulnerability study and analysis of popular and open-source MQTT broker platforms. This study will provide valuable insight for organizations and developers in fostering more resilient and secure IoT infrastructures.

1.2 Problem Description

MQTT brokers remain vulnerable to various security threats that could pose significant consequences for consumers and industries that rely on them. Recent incidents have highlighted some potential security flaws in MQTT brokers that can be exploited to disrupt operations, gain unauthorized access, or compromise the confidentiality and integrity of data [1].
There is a need for a comprehensive analysis of the vulnerabilities present in MQTT broker platforms to understand the severity of these issues and identify gaps in current security implementations [1]. There is also a gap in risk assessment in the MQTT-specific context, as widely adopted assessment tools, such as CVSS, do not capture contextual factors [51]. CVSS overlooks critical characteristics unique to the MQTT broker platform, such as frequency, that is, how often a vulnerability appears in the NVD, and popularity, reflecting how many broker platforms are affected by the same vulnerability.

A deeper understanding of vulnerabilities, their distribution across brokers, and their evolution over time could inform secure software practices. Furthermore, context-aware risk management enables effective vulnerability prioritization to mitigate risks. Addressing these gaps could strengthen the resilience of the IoT infrastructure.

1.3 Research Question

The following research questions are framed to guide the security analysis of the MQTT broker platform.

1.3.1 How have vulnerabilities in MQTT broker platforms evolved over time, and what trends can be observed in their types, frequency, and severity?

This research analyzes vulnerability databases such as the NVD (the National Vulnerability Database) and real-world attack cases to identify trends in the frequency, severity, and type of vulnerabilities over time. Furthermore, this research question will also analyze the most common attack vectors such as denial of service, authentication bypass, and data tampering and observe their change over time. Furthermore.

1.3.2 What are the critical differences in vulnerability distribution across major MQTT broker implementations, and how can these differences inform more secure design choices?

Each MQTT broker platform has a unique architecture and coding practices, leading to different security outcomes. This research question focuses on comparison of implementations of popular open-source MQTT broker platforms and explores

how the specific design of component (such as message queue managements and authentication mechanisms) impacts the emergence of vulnerabilities.

1.3.3 How are the vulnerabilities in MQTT broker platforms related to the underlying protocol architecture and the third-party libraries they integrate?

The vulnerabilities found in MQTT broker platforms are shaped by both the core MQTT protocol architecture and the third-party libraries on which these platforms rely. This will be examined by analyzing the inherent design choices of the architecture along with the external dependencies used during implementation.

1.3.4 What metrics significantly impact vulnerability prioritization and how can we design a better model that reflects the security of MQTT?

It is crucial to prioritize the vulnerabilities of MQTT broker platforms to focus on the critical factors first. Various metrics were explored considering how they reflect the severity of the vulnerability.

1.4 Thesis Outline

Chapter 1 introduces the thesis by describing the goal, purpose, description of the problem, and research question. **Chapter 2** provides a technical background, key concepts, and limitations of the thesis to understand the scope of the study. **Chapter 3** reviews related work on the security of MQTT protocols and highlights the significance of the study. **Chapter 4** establishes the theoretical foundation for this thesis, including the choice of vulnerability scanning tools. In addition, an in-depth overview of the CVSS framework is provided along with a critical evaluation of its effectiveness. **Chapter 5** outlines the methodology needed to achieve the research objectives, including data collection and tool configuration. The proposed model, along with the justification of the metrics, is also discussed in this chapter. **Chapter 6** addresses research questions by presenting the results and underlines how the findings contribute to a general understanding of the vulnerabilities related to MQTT. **Chapter 7** summarizes the findings and discusses the strengths and limitations of the proposed model and outlines the path for future work. **Chapter 8** concludes the thesis by summarizing the key findings of the study.

2

Background

This chapter introduces the fundamental concepts of the MQTT protocol and its relevance to the Internet of Things (IoT). It also outlines the limitations and delimitations of the thesis to define the boundaries.

2.1 IoT communications

From televisions connected to smart speakers, to home appliances communicating with our mobile devices, and even in vehicles where components interact in real time, these interconnected systems represent what is known as the Internet of Things (IoT). As the name suggests, IoT refers to a network of physical objects "things" that are utilized with sensors, software, and other technologies over the Internet. IoT is transforming the way data is generated and analyzed by enabling machine-to-machine communications.

In the realm of Internet of Things (IoT) communications, several messaging protocols have been developed to address the needs of specific requirements. Notable protocols include the widely used Hypertext Transfer Protocol (HTTP), the Advanced Message Queuing Protocol (AMQP), the Constrained Application Protocol (CoAP) and Message Queue Telemetry Transport (MQTT) which, due to its widespread adoption, will be the focus of this thesis project. Each of these protocols has unique characteristics, advantages, and drawbacks that influence their suitability for particular IoT scenarios.

Hypertext Transfer Protocol (HTTP) serves as the foundational protocol of the World Wide Web, enabling request-response communication between clients and servers. Although widespread and well-understood, HTTP's higher overhead and stateless nature limit its effectiveness for IoT applications that require real-time and efficient data exchange. Its suitability for IoT is often limited to devices with sufficient resources and applications where real-time communication is not critical [31].

Advanced Message Queuing Protocol (AMQP) is an open standard application layer protocol that supports a wide range of messaging patterns and provides a comprehensive set of features, including security, reliability, routing, queuing, and message-oriented communication. Unlike CoAP and MQTT, which focus on sim-

plicity and efficiency for resource-constrained devices, AMQP caters to complex enterprise-level messaging needs. AMQP operates over TCP, ensuring reliable communication in distributed systems, between servers, and in situations that require complex message brokering and transactions. However, its complexity and higher resource requirements make it unsuitable for simple IoT scenarios [31].

Constrained Application Protocol (CoAP) is a web transfer protocol designed for devices and networks with limited resources. It works on the User Datagram Protocol (UDP) level and uses a request-response model with reduced overhead compared to HTTP. Its lightweight design makes it particularly well suited for machine-to-machine (M2M) communications in resource-constrained environments. However, due to the UDP environment, CoAP cannot meet the requirements of reliability and thus it requires additional mechanisms to ensure message delivery [31].

Message Queue Telemetry Transport (MQTT) is a lightweight network protocol based on the publish-subscribe model and it usually use message queuing systems [30]. It is particularly suited for remote communication including resource-limited devices or low-bandwidth networks, making it a popular choice in Internet of Things (IoT) applications [30]. Figure 2.1 presents a comparison of several messaging protocols and their network compatibility, showing that MQTT functions over the Transmission Control Protocol (TCP), a dependable transport layer protocol that guarantees data transmission [31].

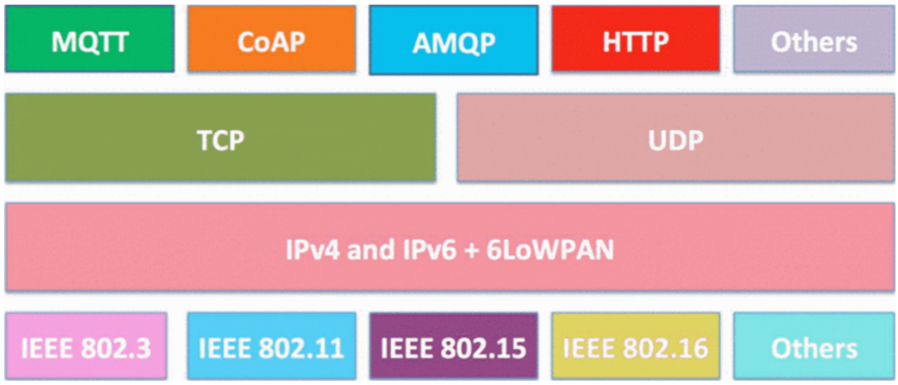


Figure 2.1: Different messaging protocols and their operability on UDP and TCP [31]

The increasing deployment of devices in the IoT has accelerated the adoption of lightweight communication protocols, such as MQTT. MQTT broker platforms Figure 2.4 play a crucial role in facilitating these communications by acting as intermediaries that manage the distribution of messages between clients. However, they also introduce significant security risks [30]. In 2023, many IoT devices that utilize MQTT were co-opted into botnets for distributed denial of service (DDoS) attacks, resulting in an increase of 300% compared to the previous year.

Attackers generated malicious traffic targeting critical infrastructures and businesses by exploiting unsecured devices. These incidents have resulted in global losses that

have exceeded 2.5 billion dollars [50]. A separate study by Kaspersky examined the implementation of MQTT in IoT devices used in the medical field, revealing that sensitive patient data was exposed to attackers [40]. Considering the importance of the MQTT protocol and its widespread use, this thesis focuses on analyzing the MQTT protocol and the broker platforms that implement it.

2.1.1 Comparative Analysis of IoT Communication Protocols

The choice of the messaging protocol in IoT systems should be aligned with the unique requirements of the applications, considering factors such as resource constraints, communication patterns, and the desired reliability level. While MQTT is often preferred for its lightweight publish-subscribe model, alternatives such as CoAP, AMQP, and HTTP offer distinct advantages and trade-offs that may better suit certain use cases [31].

Overhead and Efficiency

CoAP, which operates over UDP, has minimal overhead, making it highly efficient for constrained environments. MQTT also offers low overhead, but requires a TCP connection, which can increase overall overhead. AMQP's extensive feature set results in moderate overhead, while HTTP incurs the highest overhead due to its verbose structure [31]. Figure 2.2 illustrates the message size and message overhead for different messaging protocols.

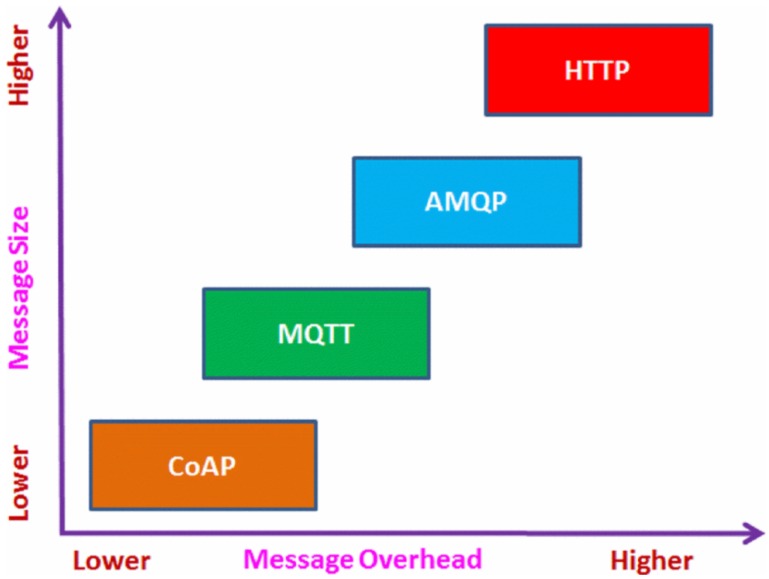


Figure 2.2: Message size and overhead comparison of different messaging protocols [31]

Communication Model

Both AMQP and MQTT support publish-subscribe patterns, which enhances efficient data distribution. In contrast, CoAP and HTTP follows a request-response models, with CoAP also providing an observer option for asynchronous communication [31].

Reliability

AMQP ensures high reliability through acknowledgments and transactions. MQTT offers configurable Quality of Service (QoS) levels, while CoAP's reliability is managed via confirmable messages. HTTP relies on TCP for inherent reliability, but lacks built-in features for messaging [31]. When it comes to reliability, MQTT scores the highest due to its quality of service levels, as illustrated in Figure 2.3.

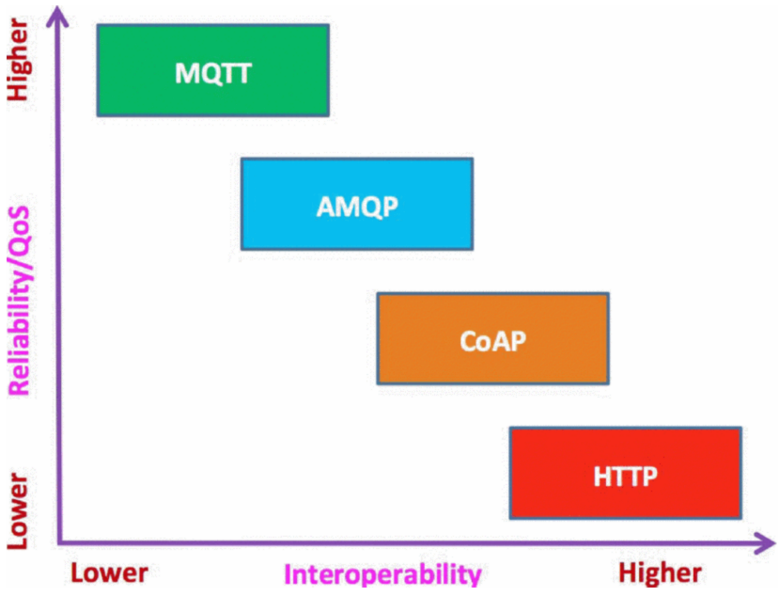


Figure 2.3: Reliability and interoperability of different messaging protocols [31]

In general, while each protocol offers unique advantages depending on the application, MQTT strikes a highly effective equilibrium between scalability, reliability, and efficiency for IoT systems. Its adaptability to constrained environments has made it the protocol of choice for many IoT Applications. It is expected to remain the leading communication standard for IoT applications for the foreseeable future, until a new protocol with better security is developed that better addresses the evolving demands of the IoT ecosystem.

2.2 MQTT Broker platforms

The MQTT broker platform is the central component and the backbone of the MQTT communication model, facilitating the routing between a publisher (device sending messages) and a subscriber (device receiving the messages) [20][21].

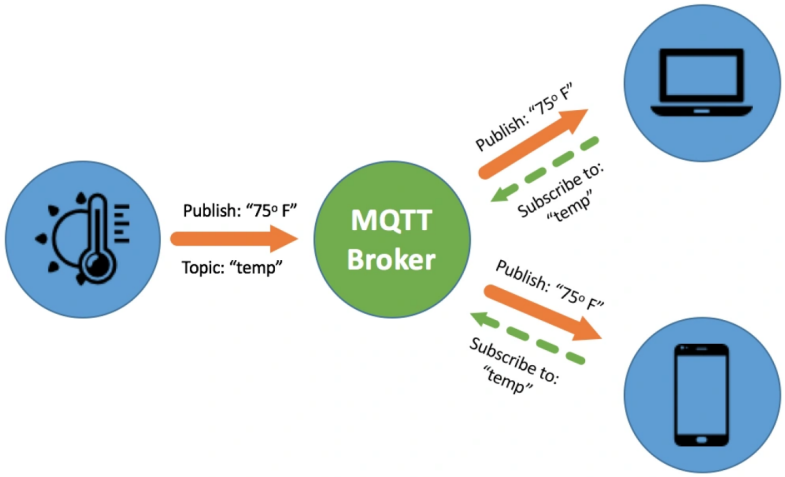


Figure 2.4: Illustration of publish-subscribe communication of MQTT broker platform. The figure shows a publisher publishing the temperature "temp" and subscribers who subscribed the temperature and receiving them.

There are over 70 broker platforms on the market [52]. However, due to limited resources, this study focuses on five open source and widely used broker platforms, including **Eclipse Mosquitto**[1] which is one of the widely used broker platform known for its reliability and simplicity in embedded systems.**VerneMQ**[2] a scalable and reliable broker built on Erlang, **EMQX**[3] a high-performance broker build to handle millions of connections simultaneously, **HiveMQ**[4] a scalable broker platform built for enterprise use, and finally **Mosca**[5] a lightweight broker platform Built on Node.js for lightweight applications.

All of these broker platforms differ in terms of scalability, performance, and security features, although they implement the MQTT protocol. Broker platforms play a vital role in mediating a high level of sensitive IoT data traffic and are the primary target of attackers to exploit them. In this thesis, the above broker platforms and their third-party dependencies are analyzed for vulnerabilities and the security risks they may introduce.

Brokers implement essential functionalities such as **topic-based filtering** meaning that brokers create a hierarchical topic structure and ensure that messages are

[1] `https://mosquitto.org`
[2] `https://vernemq.com`
[3] `https://www.emqx.com/en`
[4] `https://www.hivemq.com`
[5] `https://www.mosca.io`

delivered to clients who have subscribed to the relevant topic [22]. **Session management** is another feature of the brokers, which means that the broker maintains session states such as subscriptions and undelivered messages to ensure reliable and consistent communication [22]. **Authentication management** is also a functionality that modern broker platforms provide, which is authentication through username / password, tokens, or certificates and control access to protect the system from unauthorized users [11].

Quality of Service (QoS) is another functionality that MQTT broker platforms provide, which is a mechanism that allows developers to balance delivery and performance according to the needs of the application [21]. QoS is the guarantee of the delivery of message between the client and the broker. MQTT facilitates three levels of QoS as summarized in the table 2.1 to balance network traffic with a higher degree of reliability and performance [20][21].

Table 2.1: Comparison of MQTT quality of Service (QoS) Levels and example in where it is typically used.

QoS Level	**Guarantee**	**Acknowledgment**	**Duplicate Messages**	**Typical Use Case**
0	At most once	No	No	Non-critical sensor data, where occasional loss is acceptable.
1	At least once	Yes	Possible	Logging, notifications, or state updates where message loss is not acceptable but duplicates can be managed.
2	Exactly once	Yes (Four-step handshake)	No	Financial transactions, control messages, or critical commands that must be processed only once.

QoS level 0, known as fire and forget, where messages are delivered once at most. The delivery is not acknowledged by the receiver, nor is a retry performed in case the message is lost during transmission [20][21]. In **QoS level 1,** messages are guaranteed to arrive at least once in this level. However, messages may be delivered more than once due to the loss of acknowledgments and retransmission of messages [20][21]. In **QoS level 2,** messages are guaranteed to be delivered exactly once in this level. This is the strongest QoS with no loss and duplication of messages and is achieved through a four-step handshake between the publisher and subscriber [20][21].

2.3 Challenges

This research encountered challenges that needed to be addressed to ensure the effectiveness of the analysis.

Diversity in Implementations of MQTT
MQTT broker platforms vary in their implementation, security measures, and features. Differences between platforms can present challenges in creating a standardized evaluation methodology that is applicable across all brokers.

Nature of Threats
Cyber attacks constantly evolve and new vulnerabilities may emerge even if existing ones are addressed. Keeping the analysis relevant requires continuous monitoring of emerging threats, which can be difficult given the dynamic cybersecurity landscape.

2.4 Limitations and Delimitations

2.4.1 Delimitation

This study focuses on open-source MQTT broker platforms to ensure the availability, accessibility, and reproducibility of the results. Non-security aspects like the performance and usability of MQTT and broker platforms are beyond the scope and will be excluded from this study.

The focus of this study was on analyzing the most recent incidents to ensure that this project remains relevant to the latest cyberattacks. In addition, the effectiveness of CVSS was evaluated to propose improvements.

2.4.2 Limitations

Data Sources
Our research focuses only on selected open-source brokers listed in the Appendix C.1, commercial platforms such as AWS or Microsoft Azure IoT Hub are excluded. Therefore, the results of the analysis may not represent the prevalence of all vulnerabilities and limit the generalizability of the findings.

Environmental Dependency
The identified vulnerability may vary in different deployment environments, such as cloud versus local or different operating systems. Our research cannot account for all possible deployment scenarios, which may lead to some vulnerabilities not being correctly identified.

2.4.3 Limitations of The Improved CVSS Scoring System

Although the improved scoring system aims to address certain gaps in CVSS by modifying various metrics, it is essential to acknowledge its limitations.

Subjectivity in New Metrics
Including new metrics such as popularity and frequency in the category "Repeatability" introduces subjectivity. Unlike traditional metrics, which are based on technical exploitability or system impact, these metrics are context dependent and may vary significantly across different environments, making consistent scoring difficult.

Simplification of The Scoring System
By omitting metrics such as "exploit code maturity" and "attack complexity", which are part of the original CVSS, simplifies the new scoring system and reduces complexity. However, it also limits the granularity of the analysis. This potentially will oversimplify certain attack scenarios that depend heavily on these factors.

Tooling and Ecosystem Compatibility
Existing vulnerability scanners and security tools are built around the traditional CVSS framework. Integrating this improved model would require updating the tooling, which may not be feasible or supported without a broader community support.

Lack of Empirical Validation
The improved scoring model has yet to be validated through large-scale vulnerability datasets or widespread industry adoption. Its effectiveness in real-world scenarios remains theoretical and should be further evaluated through case studies or implementation in various domains such as IoT to evaluate its performance against the standard CVSS.

3

Related Work

3.1 Review of Existing Research

In the realm of IoT communications, the MQTT protocol has emerged as a pivotal standard due to its lightweight design and efficiency. However, its widespread adoption has also highlighted various security challenges associated with MQTT broker platforms. This section reviews related work and delves into prior research that has analyzed the security aspects of well-known MQTT broker platforms.

Di Paolo et al. conducted an empirical security assessment of several widely used MQTT components, including three client libraries and five broker libraries. Unlike this thesis, which relies on static analysis of the code, the researchers performed a dynamic analysis using fuzzing methodology and even tested a physical smart home device. In this research, no critical vulnerabilities were found, although it was revealed that some libraries deviate from standard protocols. These discrepancies could be exploited, potentially leading to inconsistencies or vulnerabilities in the system [38].
The researchers in [38] observed possible behavioral deviations from the standard that could lead applications to possibly inconsistent or critical states. While almost all the considered libraries correctly handled most of the interactions, some anomalies have been detected that could be exploited to target applications, mainly exposing them to denial of service (DoS) attacks.

The development of MQTTactic is another significant contribution to the field. MQTTactic is a semi-automatic tool designed to verify MQTT broker implementations formally. This tool aims to detect logic flaws within broker platforms by generating specific security properties, thereby improving the overall robustness and reliability of MQTT systems [52]. Furthermore, a study by Vaccari et al. revealed the susceptibility of the MQTT protocol to slow denial-of-service (DoS) attacks. This research illustrated how such attacks could compromise the availability of MQTT services, emphasizing the need for strong security measures within broker implementations [47]. These studies emphasize the need for thorough security evaluations and the adoption of strong security mechanisms in MQTT broker platforms to ensure the integrity and reliability of IoT communications.

3.2 Significance of the Study

Despite existing research efforts, there are gaps in understanding the MQTT protocol, its limitations, and how its interaction with a specific broker can introduce vulnerabilities. This underscores the urgent need for a comprehensive security analysis of software associated with MQTT brokers, considering its widespread use in IoT where robust security is vital.
Researchers in [51] highlight the inconsistencies in the CVSS scoring system and claim, "Our study reveals that most evaluators are aware of the problematic aspects of CVSS, but still see CVSS as a useful tool for vulnerability assessment". These findings indicate that current risk assessment systems such as CVSS require improvements to better align with real-world need.
The significance of this study is its potential to improve the security of MQTT broker platforms and to contribute to the research on MQTT and IoT security by addressing existing gaps in MQTT-specific risk assessment systems.

4

Theory

This chapter presents the key concepts and the theoretical foundation on which the research is based. In addition, the original CVSS framework, its metrics, and the formula for calculating vulnerability scores are presented in this section, which later in the result section will be used as a baseline to design a new MQTT-specific vulnerability scoring model.

4.1 Literature Review

The increase in the use of MQTT as a messaging protocol for IoT, combined with the security risk and vulnerabilities it introduces, has attracted not only hackers, but researchers as well. This literature review highlights several key findings related to the security risks of the MQTT protocol and broker platforms.

Vulnerability in protocol and implementation

A study highlighted that the lack of encryption in MQTT allows data to be easily intercepted, enabling attackers to tamper with the data and potentially control the system [3]. A vulnerability in MQTT is that it remains exposed to various attack methods due to the basic security mechanism it provides during data transmission. For example, on a public network, tools like "Shodan" can be used to scan unsecured MQTT brokers to publish malicious messages [3].
A study by Bakar et al. shows that in the local network, attackers can perform traffic analysis to extract key information. The DOS (Denial of Service) attack can also be launched due to the weak authentication mechanisms of MQTT [5].

Vulnerability in Broker platforms

More than 70 open source MQTT brokers are used in various production environments [52] and each differs significantly in their architecture. Any vulnerability in these brokers could have significant consequences, affecting both individual systems and larger IoT infrastructures. A study by Mahmoud et al. emphasizes the importance of optional security plugins for each broker platform in mitigating security vulnerabilities [27].

Real world incidents

A study led by Kaspersky, an international cybersecurity provider, reported a 300 % increase in DDoS attacks globally, where MQTT vulnerabilities were exploited to recruit IoT devices into botnets [40].

4.2 Vulnerability Scanning Tools

A security scanning tool is required to identify vulnerabilities in broker platforms. There are several open-source vulnerability scanning tools available. This section reviews four commonly used tools, highlighting their strengths and limitations to provide a rationale for selecting the appropriate one for the analysis in this thesis.

Grype

Grype[1] is a lightweight, open source vulnerability scanner that can scan docker images and file systems for vulnerabilities. It supports widely used vulnerability databases, such as NVD and GitHub. Grype is one of the easiest options for fast and free scans of broker platforms. It is ideal for quick checks without extra setup. However, Grype is not as feature rich as Snyk in terms of integrations or remediation guidance, and lacks advanced prioritizing and reporting features [6].

Trivy

Trivy[2] is another open source vulnerability scanning tool that is extremely fast and versatile, making it ideal for a quick overview of vulnerabilities without a complex configuration. It can scan container images, file systems, repositories, and Kubernetes manifests. It also scans for misconfigurations, for example in docker files. Trivy is beginner-friendly and supports various output formats, such as JSON, table, and templates. It is an excellent choice for scanning environments with minimal setup, but does not provide as much actionable remediation advice as Snyk [6].
A disadvantage of Trivy is that it may overwhelm users by showing a high number of low-severity vulnerabilities.

Docker Scout

Docker scout[3] is designed specifically for Docker users to analyze container images. It provides vulnerability data for layers within Docker images. Docker scout integrates seamlessly with Docker Desktop and docker CLI. A disadvantage of Docker scout is that it is focused only on docker images and is limited in its ability to scan dependencies outside the image. Docker scout is a good choice if the broker platform is already containerized and you want an easy, Docker native solution to scan it.

[1] `https://github.com/anchore/grype`
[2] `https://trivy.dev/latest/docs`
[3] `https://docs.docker.com/scout`

Docker scout is simple to use but lacks the depth and breadth of tools such as Trivy and Snyk [6].

Snyk

Snyk[4] is an excellent tool for developers who want a complete overview of vulnerabilities, including container images, open source dependencies, and infrastructure-as-code (IaC). An advantage of Snyk over other analyzing tools is that it provides actionable remediation, advice, and prioritizes vulnerabilities. It also offers integrations with Gitlabs, GitHub, and Docker Hub to seamlessly integrate with CI/CD pipelines and container registries [6].
A disadvantage of Snyk is that it requires an account to use it and might feel heavy for quick scans [6]. Furthermore, the free tier has some limitations, for example, a limited number of scans per month. Snyk is a strong choice if the broker platform is in a docker container or has dependencies that need to be monitored in the code. It scans images and provides details of vulnerabilities, including CVSS scores and fixes.

Table 4.1: Comparison of different Vulnerability Scanning Tools for MQTT Brokers

Tool	**Advantages**	**Disadvantages**
Grype	- Lightweight, open-source, easy to install - Scans Docker images and file systems - Supports NVD and GitHub databases - Great for quick checks with minimal setup	- Lacks rich features - No advanced remediation support
Docker Scout	- Docker-native tool - Analyzes vulnerabilities in image layers - Integrates with Docker CLI and Desktop	- Limited to Docker images - Cannot scan external dependencies - Not as comprehensive as Snyk
Trivy	- Fast and versatile - Scans containers, file systems, repos, and Kubernetes - Detects misconfigurations - Beginner-friendly output in JSON, table	- Can overwhelm with low-severity issues - Less actionable guidance than Snyk
Snyk	- Complete scanning: images, open source - Actionable fixes with CVSS scores - Integration with GitHub, Docker Hub - Ideal for CI/CD security	- Requires account to use - Free tier has scan limits - Heavy for quick scans

Table 4.1 summarizes the advantages and disadvantages of some popular vulnerability scanning tools. Given the advantages, Snyk is the scanning tool that was selected to analyze the broker platforms in this project.

[4] `https://snyk.io`

4.3 Metrics of CVSS

Common Vulnerability Scoring System (CVSS) is a widely used scoring model to assess the extend of risk of vulnerabilities [49]. It provides a standard and quantitative measurement method for assessing a large number of cybersecurity vulnerabilities. The latest version, CVSS V4.0, was released on November 1, 2023 [14]. However, this project uses version 3.1, because many vulnerabilities in the NVD are only recorded using CVSS v3.1.

To calculate the severity of a vulnerability, CVSS accounts for three sets of metrics: Basic, Temporal, and Environmental. Each category includes specific factors that contribute to the final score. This thesis uses abbreviations to represent the specific metrics in CVSS, which are defined in Terminology "List of metric Abbreviations".

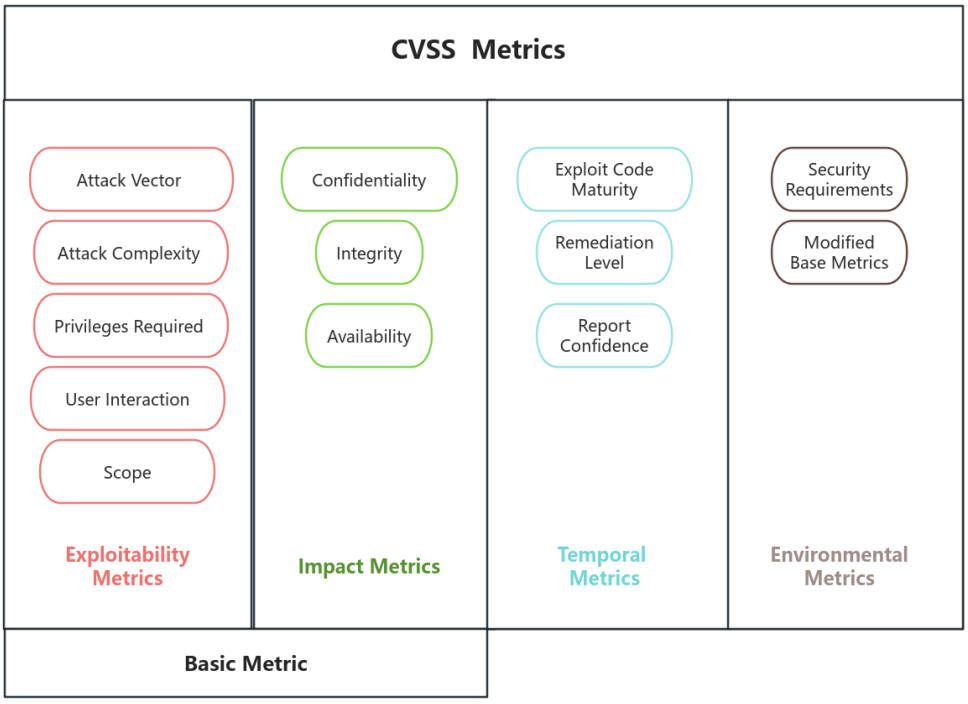


Figure 4.1: Different sets of metrics that contributes to final score of CVSS.

Figure 4.1 illustrates the specific components of each group. Basic metrics ([13]) describe core properties of a vulnerability that keep an unchanged trend over time and in different user contexts. However, many programmers or companies only use the base score as the final score. This practice is even recommended by some global organizations, such as the global payment card industry [45]. Temporal metrics reflect the status of fixing, and environmental metrics account for the settings and environment of system. To assess a vulnerability, relevant metrics are set and then a vulnerability score is calculated.

Figure 4.2 shows the specific metrics of the Security Requirements and the Modified Base metrics. The security requirement refers to the importance it has for the system. This metric does not directly alter the impact of a vulnerability, but adjusts its weight in the final score. Modified confidentiality reflects the actual impact of vulnerability in a specific environment and usually depends on the real-world requirements of the system.

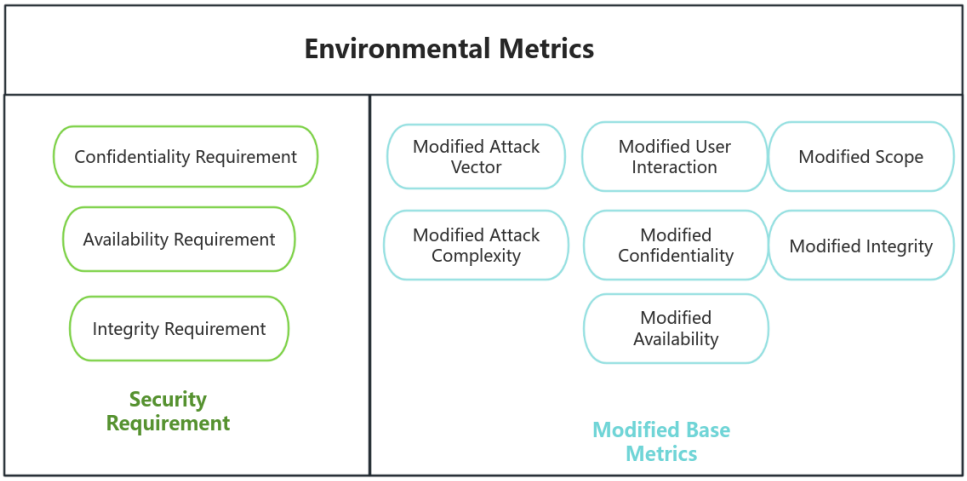


Figure 4.2: Security Requipment and Modified Base Metrics in Environmental Metrics of CVSS

Base score represents the nature of the vulnerability, the temporal score focuses on the impact of remediation technology, and the environmental score is very subjective, depending on the environment and how the programmer evaluates it. Also, CVSS uses a scoring scale of zero to ten to indicate the severity of a vulnerability, as illustrated in Table 4.2.

Table 4.2: Vulnerability levels of CVSS and their numerical value. [13]

Score	**Vulnerability Levels**
0	Safe
0 - 3.9	Low
4 - 6.9	Medium
7 - 8.9	High
9 - 10	Critical

4.3.1 Calculation of CVSS

In CVSS, metrics are quantified and incorporated into a formula to calculate the vulnerability score. Although the calculation methods differ between the different metric groups, they provide valuable insights for the design of the improved scoring system.

4.3.1.1 Base Score

Base score represents the intrinsic properties of vulnerabilities and remains constant, regardless of the environment and time. Formulas (4.1)-(4.4) show how CVSS calculates the base score [49]. In these formulas, ISC stands for Information Security Composite, which is an intermediate result. In (4.1), confidentiality (C), integrity (I), and availability (A) can take values of 0, 0.22, and 0.56. In (4.2), scope (S) is a specific metric in CVSS that refers to whether the impact of a vulnerability extends beyond the privileges or authority level of its initial context. For example, in a Linux environment, a user-level vulnerability that escalates to affect the superuser level, would indicates a change in scope. In formula (4.3), attack complexity (AC), attack vector (AV), privileges required (PR), and user interaction (UI) are four metrics ranging from 0 to 1. Finally, in equation (4.4), CVSS performs a rounding operation (i.e, "round up" returns the smallest number, rounded up to one decimal number) based on the scope. Formulas (4.1)-(4.4) are from chapter 7.1 of [13].

$$\text{ISC}_{\text{Base}} = 1 - (1 - C)(1 - I)(1 - A) \tag{4.1}$$

$$\text{Impact} = \begin{cases} 6.42 \times \text{ISC}_{\text{Base}}, & \text{if Scope = Unchanged} \\ 7.52 \times (\text{ISC}_{\text{Base}} - 0.029) - \\ 3.25 \times (\text{ISC}_{\text{Base}} - 0.02)^{15}, & \text{if Scope = Changed} \end{cases} \tag{4.2}$$

$$\text{Exploitability} = 8.22 \times AV \times AC \times PR \times UI \tag{4.3}$$

$$\text{Base Score} = \begin{cases} 0, & \text{if Impact} \leq 0 \\ \text{RoundUp}(\min(\text{Impact} + \text{Exploitability}, 10)), & \text{if Scope = Unchanged} \\ \text{RoundUp}(\min(1.08 \times (\text{Impact} + \text{Exploitability}), 10)), & \text{if Scope = Changed} \end{cases} \tag{4.4}$$

Formula (4.1) helps reduce unnecessary computation when the values of C, I, and A are all zero (be set as "High"). In addition, the metric S is used to differentiate

the cases in Equations (4.2) and (4.4) to reflect various security situations.

Temporal Score formula :

In Equation (4.5), Exploit Code Maturity (ECM), Report Confidence (RC), and Remediation Level (RL) are all factors between 0.9 and 1.0. Equation (4.5) is from chapter 7.2 of [13].

$$\text{Temporal Score} = \text{RoundUp}(\text{Base Score} \times E \times RL \times RC) \tag{4.5}$$

4.3.1.2 Environmental Score

The Environmental Score does not directly influence the calculation of the Base Score. It is a variation of the basic score that allows users to modify the specific value of the metrics setting to facilitate different situations. Specifically, programmers can adjust the weights of MI (Modified Integrity), MC (Modified Confidentiality), and MA (Modified Availability) based on actual demands. The CR (Confidentiality Requirement), IR (Integrity Requirement), and AR (Availability Requirement) metrics indicate the importance of each security objective to the system and are still defined subjectively by programmers. Hence, the calculation methods of Equations (4.6) to (4.9) are the same as those in Equations (4.1) to (4.4), Although the specific variables are different. Formulas (4.6)-(4.9) are from chapter 7.3 of [13].

$$\text{MISS} = \min\left[1 - (1 - CR \times MC)(1 - IR \times MI)(1 - AR \times MA),\ 0.915\right] \tag{4.6}$$

$$\text{Modified Impact} = \begin{cases} 6.42 \times \text{MISS}, & \text{if Scope} = \text{Unchanged} \\ 7.52 \times (\text{MISS} - 0.029) & \\ -3.25 \times (\text{MISS} \times 0.9731 - 0.02)^{13}, & \text{if Scope} = \text{Changed} \end{cases} \tag{4.7}$$

$$\text{Modified Exploitability} = 8.22 \times MAV \times MAC \times MPR \times MUI \tag{4.8}$$

In formula (4.9), ES represents the Environmental Score and it is very similar to Equation (4.4).

$$\text{ES} = \begin{cases} 0, & \text{if ModifiedImpact} \leq 0 \\ \text{RoundUp}(\text{RoundUp}[\min([\text{ModifiedImpact} & \\ + \text{ModifiedExploitability}], 10)] \times ECM \times RL \times RC), & \text{if Scope} = \text{Unchanged} \\ \text{RoundUp}(\min(1.08 \times (\text{Impact} + \text{Exploitability}), 10)), & \text{if Scope} = \text{Changed} \end{cases} \tag{4.9}$$

The values used in the Basic Score are by default constant. If users modify some value, then the score they get is Environmental Score instead of Base Score. Therefore, users either use Base Score or Environmental Score, not both simultaneously. In other words, no new metrics are introduced in Environmental Metrics.

4.4 CVSS Evaluation

Although CVSS is the most widely used vulnerability prioritization model, our analysis revealed that certain aspects of its design introduce unnecessary complexity and semantic ambiguity, which can affect the consistency and interpretability of results. For example, the authors of [24] identified two redundant metrics Scope and Attack Complexity. In addition, CVSSv4.0 [15], the "Scope" metric is split into three components "SC, SI, SA" to address this ambiguity. Furthermore, CVSS does not account for real-world distributions of vulnerabilities, which limits its ability to accurately reflect practical risk.

Inconsistency Due to Ambiguity

Ideally, CVSS should produce consistent scores regardless of who applies it. However, some metric definitions in CVSS are vague, leading to inconsistent results. In [51], Wunder et al. identify inconsistencies in the application of AV, UI, and S metrics. These inconsistencies arise because users often have difficulty accurately determining the appropriate settings.

Real World Vulnerability Distributions Ignored

Spatial distributions and exploitation frequency are important characteristics of vulnerabilities; however, they are not reflected in CVSS metrics[13]. Roughly 80% of vulnerabilities are exploited before their official disclosure, as noted in [42]. Hence, the life cycle of a vulnerability is an aspect that reflects the risk level of vulnerability, that is, a longer time to discover means a higher risk of being exploited. Furthermore, finding four in [25], shows that 20% of vulnerability types account for 70% total number of vulnerabilities. This means that most software systems are affected by a small subset of vulnerability types. If resources are limited and cannot address all vulnerabilities, prioritizing these frequently exploited vulnerabilities is essential to protect the majority of users.
These findings demonstrate that real world exploitation patterns play a critical role in prioritizing vulnerabilities. However, they are not reflected in the CVSS framework, revealing a gap between CVSS metrics and practical vulnerability prioritization.

5

Methods

This chapter outlines the methodology employed to achieve the research objectives, including data collection and tool configuration. Additionally, it provides a detailed discussion of the improved model, along with an explanation of the chosen metrics.

5.1 Methodology

This thesis employs a hybrid methodology that combines evolution research and empirical study approaches. These methodologies involve systematic analysis and assessment [46] to achieve the research objectives of understanding, evaluating, and prioritizing vulnerabilities in MQTT broker platforms.

The empirical study involves collecting data from real-world environments to validate the findings, while the evaluation research focuses on assessing existing systems against predefined criteria [46].

This hybrid approach ensures both the practical insights gained from empirical studies and the theoretical rigor provided by evaluation research, making it well suited for a comprehensive assessment of MQTT broker security and for offering actionable recommendations to enhance their resilience.

5.1.1 Evaluation research

Literature and Contextual Review A systematic review of the academic literature and industry reports was conducted to establish a basic understanding of MQTT vulnerabilities and security mechanisms. Historical trends from vulnerability databases, such as the NVD, were analyzed to identify recurring security patterns and common attack vectors. Defining evaluation criteria based on established security benchmarks, such as CVSS scores, exploitability, and impact metrics, was also a crucial step in this process.

Comparative Analysis
Vulnerabilities across various MQTT broker platforms were identified and analyzed based on their architectures and security measures. For a subset of these identified vulnerabilities, scores were calculated using both the standard CVSS and the

proposed scoring model. Comparing these results highlights the effectiveness and contextual relevance of the proposed model.

Vulnerability Prioritization
To facilitate effective decision-making, a prioritization framework was developed that integrates empirical findings and evaluation benchmarks, including CVSS-based severity scores. Vulnerabilities were identified, their severity scores were calculated and ranked, and the most critical issues were highlighted.

5.1.2 Empirical study

Platform Selection
Five open-source broker platforms were selected for analysis. The selection criteria included open-source, public availability, and widely used in IoT projects to ensure broad representativeness and applicability.

Vulnerability Scanning and Analysis
Snyk generated a security report for each broker platform. The reports were then manually reviewed to identify and analyze vulnerability patterns. Furthermore, identified vulnerabilities were categorized by type, such as code-level bugs, configuration issues, and third-party dependency weaknesses.

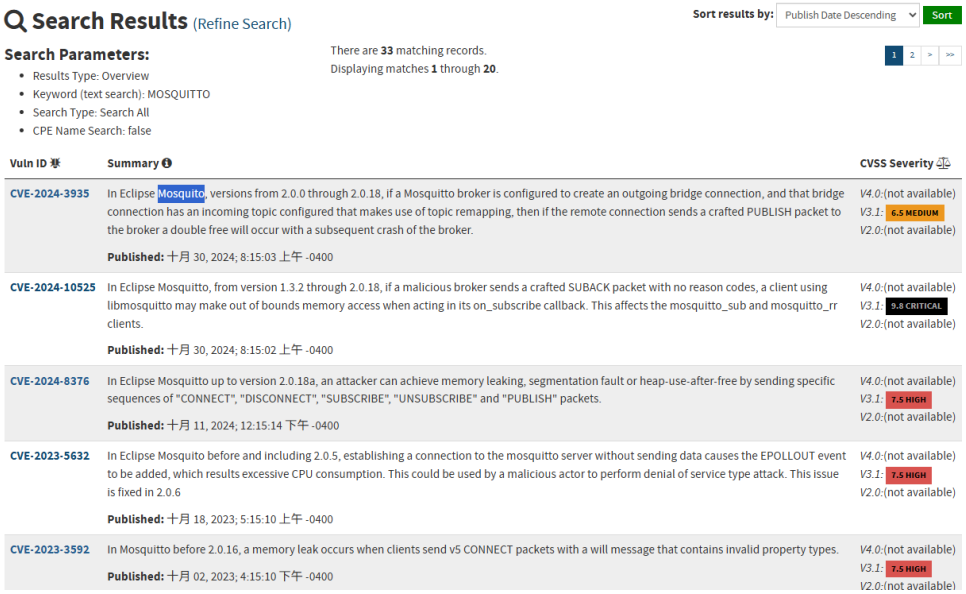
Empirical Validation
The proposed vulnerability scoring model was validated using real-world data by applying it to the vulnerabilities identified by Snyk during the scanning process. The resulting scores were then compared with traditional CVSS scores, as shown in Table 6.28, to assess the model's ability to accurately capture vulnerabilities in MQTT environments. This step confirmed the effectiveness of the proposed vulnerability scoring model in supporting vulnerability management in IoT systems.

5.2 Data Collection

The data in this thesis were collected from the NVD and from reports generated by Snyk. The source code of the broker platforms analyzed is publicly available and published on GitHub, as listed in the Appendix C.1.

5.2.1 Data from NVD

The NVD is a continuously updated repository that provides detailed information on cybersecurity vulnerabilities, including security checklists, descriptions of software flaws, product identifiers, and impact metrics [37]. Relevant vulnerability information was retrieved through targeted searches using the NVD interface. For example, to identify vulnerabilities related to “Mosquitto”, the broker’s name was entered into the NVD search engine. Figure 5.1 shows the search results for “Mosquitto”, including CVE identifiers, descriptions, and the corresponding CVSS scores.



Search Results (Refine Search)

Sort results by: Publish Date Descending Sort

Search Parameters:

- Results Type: Overview
- Keyword (text search): MOSQUITTO
- Search Type: Search All
- CPE Name Search: false

There are **33** matching records.
Displaying matches **1** through **20**.

1 2 > >>

Vuln ID	Summary	CVSS Severity
CVE-2024-3935	In Eclipse Mosquitto, versions from 2.0.0 through 2.0.18, if a Mosquitto broker is configured to create an outgoing bridge connection, and that bridge connection has an incoming topic configured that makes use of topic remapping, then if the remote connection sends a crafted PUBLISH packet to the broker a double free will occur with a subsequent crash of the broker. **Published:** 十月 30, 2024; 8:15:03 上午 -0400	*V4.0:*(not available) *V3.1:* 6.5 MEDIUM *V2.0:*(not available)
CVE-2024-10525	In Eclipse Mosquitto, from version 1.3.2 through 2.0.18, if a malicious broker sends a crafted SUBACK packet with no reason codes, a client using libmosquitto may make out of bounds memory access when acting in its on_subscribe callback. This affects the mosquitto_sub and mosquitto_rr clients. **Published:** 十月 30, 2024; 8:15:02 上午 -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2024-8376	In Eclipse Mosquitto up to version 2.0.18a, an attacker can achieve memory leaking, segmentation fault or heap-use-after-free by sending specific sequences of "CONNECT", "DISCONNECT", "SUBSCRIBE", "UNSUBSCRIBE" and "PUBLISH" packets. **Published:** 十月 11, 2024; 12:15:14 下午 -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2023-5632	In Eclipse Mosquito before and including 2.0.5, establishing a connection to the mosquitto server without sending data causes the EPOLLOUT event to be added, which results excessive CPU consumption. This could be used by a malicious actor to perform denial of service type attack. This issue is fixed in 2.0.6 **Published:** 十月 18, 2023; 5:15:10 上午 -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2023-3592	In Mosquitto before 2.0.16, a memory leak occurs when clients send v5 CONNECT packets with a will message that contains invalid property types. **Published:** 十月 02, 2023; 4:15:10 下午 -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)

Figure 5.1: Result of searching on NVD with key word "Mosquitto". The figure shows vulnerabilities related to Mosquitto broker platform.

5.2.2 Data from Snyk generated reports

Configuring Snyk for analysis

To perform the analysis using Snyk, the latest source code versions of each selected broker platform, listed in Appendix C.1, were forked into GitHub repositories. Note that for Snyk to successfully scan a repository, it must be owned by a user or organization; repositories with contributor access only cannot be scanned by Snyk.

The forked repository was then added to Snyk for static analysis. Upon adding the repository, Snyk automatically scanned it for vulnerabilities and generated a security report. Figure 5.2 illustrates the workflow for using Snyk.

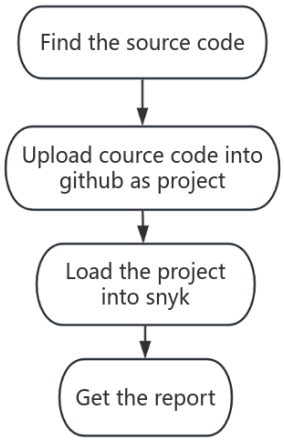


Figure 5.2: Step by step setup and configuration of Snyk.

Data collection from Snyk

The report generated by Snyk lists all identified issues, their severity score, and remediation measure to mitigate the risk of each vulnerability. The issues are categorized by their corresponding CWE-ID for easier reference. Figure 5.3 shows the results of the vulnerability scan on the Mosquetto broker platform.

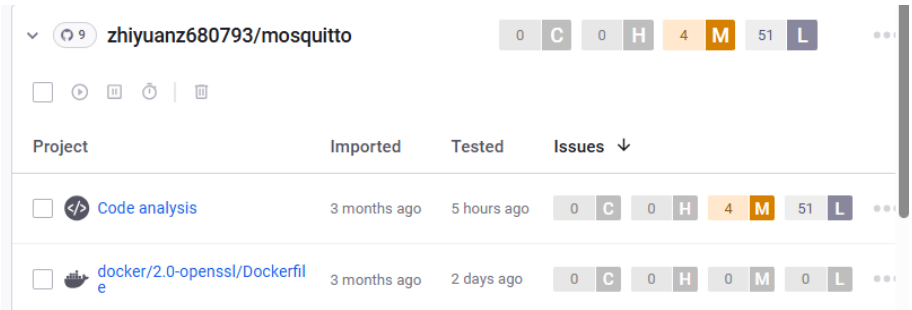


Figure 5.3: Result of scanning Mosquitto broker platform by Snyk

Figure 5.4 presents detailed information on a specific issue, "Path Traversal", where vulnerable components are highlighted and a possible remediation method is recommended. The security level is based on CVSS and a CWE code that refers to the reported issue in NVD. However, it is interesting that there is a score in the upper right corner of the figure, which is a scoring system used by Snyk for programmers' reference only.

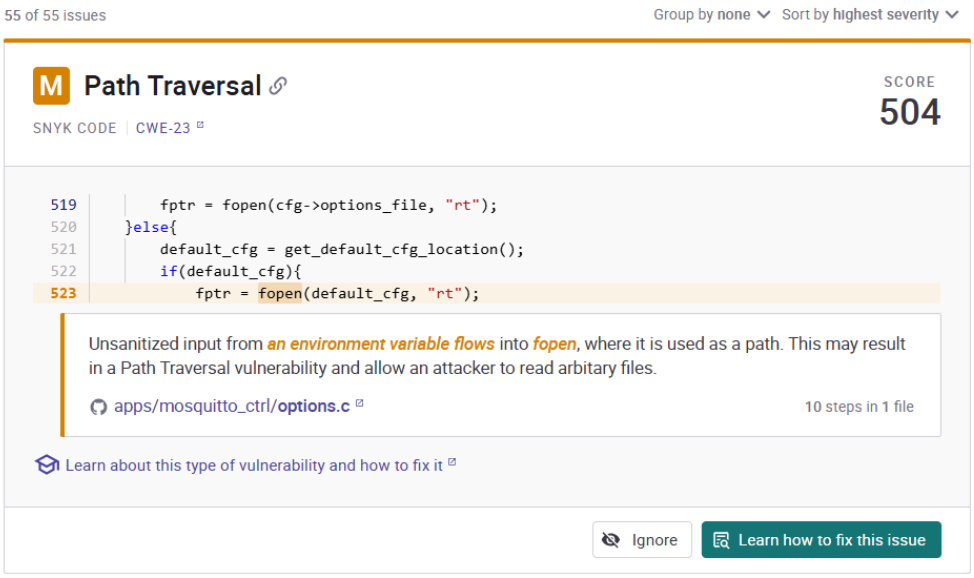


Figure 5.4: A specific issue snyk found in a broker platform and recommended step to mitigate the risk

Each report generated by Snyk was manually reviewed to get an overview of communication through broker platforms, identify vulnerabilities specific to each broker platform, and vulnerabilities that multiple brokers have in common.

6

Results

This section addresses the research questions introduced in the first chapter and highlights how the findings contribute to a broader understanding of vulnerabilities related to MQTT.

6.1 RQ1: How have vulnerabilities in MQTT broker platforms evolved over time and what trends can be observed in their types, frequency, and severity?

The MQTT protocol has become a cornerstone of IoT communication due to its efficiency and lightweight. However, as its adoption has increased, so has the attention of malicious actors seeking to exploit vulnerabilities in MQTT broker implementations.
With the growing adoption of MQTT broker platforms, understanding their security vulnerabilities and how they have evolved over time has become increasingly important. This section, corresponding to the first research question, highlights the trends in security weaknesses across popular MQTT brokers, identifies common attack vectors, and analyzes how these vulnerabilities have changed over time.
MQTT brokers have significantly strengthened their security posture due to increased awareness of IoT threats and advancements in cybersecurity practices. A historical analysis of vulnerabilities in MQTT brokers reveals several key trends that are crucial to informing the development of robust security in MQTT-based systems.
The security landscape of MQTT broker platforms has evolved significantly over time, reflecting a broader trend in cybersecurity, where threats continue to evolve in response defensive mechanisms. These vulnerability trends can be broken down into three distinct phases.

Early Stages (Pre-2015): Functionality Over Security

During the early stages of MQTT adoption, the protocol was utilized primarily in industrial IoT applications, where security was often an afterthought. MQTT protocol was designed primarily for efficiency rather than robust protection. One of

the primary security concerns during this era was authentication and authorization weaknesses, where many brokers allowed clients to connect without requiring credentials. Furthermore, in earlier versions of Eclipse Mosquitto (prior to version 1.4), authentication was not enabled by default [8]. This allowed attackers to easily connect and subscribe to topics, leading to data leakage and possible disruption of IoT networks [8].
Unencrypted communication was another vulnerability in which MQTT messages were often transmitted in plain text, making them susceptible to man-in-the-middle (MITM) attacks [26].
In environments where Transport Layer Security (TLS) was not enforced, attackers could intercept MQTT sessions, hijack connections, and maintain persistent access to broker resources. This lack of session protection made it easier for adversaries to manipulate topics, subscribe to sensitive data, or inject malicious payloads [23].

Open MQTT ports were also a vulnerability in which broker platforms were frequently exposed to the Internet with default configurations, enabling attackers to easily discover and exploit them [23]. The increasing adoption of IoT in consumer devices, such as connected cars and smart home devices, has attracted not only malicious actors but also the research community. Several studies have identified security vulnerabilities in MQTT broker platforms, prompting these platforms to implement enhanced security features.

Development Phase (2015-2020): Introduction of Security Features

With the increasing popularity of IoT and the growing awareness of the cybersecurity threats it introduces, MQTT brokers began to incorporate security mechanisms. One such mechanism was TLS encryption, in which transport layer security (TLS 1.2/1.3) was introduced to protect message integrity and confidentiality [43].
Stronger authentication mechanisms such as OAuth 2.0, JWT authentication, and API key-based access mechanisms were implemented to improve authentication [43].
Access Control Lists (ACLs) were another security feature introduced by adopting fine-grained permission systems to limit client access to specific topics [11].
In 2017, HiveMQ introduced advanced authentication mechanisms, along with integration with enterprise identity providers. Despite that, researchers detected misconfigurations in default ACLs settings, which could allow subscriptions to unauthorized topics [22].

Recent Trends (2021-Present): Advanced Threats and Mitigation

Despite significant security improvements, MQTT brokers continue to face sophisticated attacks. One of the most pressing concerns is Denial-of-Service (DoS) attacks, in which attackers exploit high connection limits and malformed MQTT packets to crash brokers [18][50]. DoS attacks usually target the broker's ability to manage client connections. Attackers often exploit the limited resource capacity of MQTT

brokers by initiating large-scale connection floods, where adversaries establish thousands of simultaneous connections and exhaust the broker's available resources, leading to complete shutdown or service failure [18].
Another critical vulnerability is memory corruption and resource exhaustion, resulting from improper memory management practices such as use-after-free and buffer overflow in some broker implemented [9].
Injecting malformed MQTT packets is also a major threat, which is the injection of oversize or corrupted data that disrupts the broker's message processing mechanisms, potentially causing unpredictable behavior or crashes [2]. A notable example of this occurred in the EMQX broker, where a flaw in its message queue processing allowed adversaries to flood the broker with high-retention messages. This vulnerability resulted in excessive memory consumption and service unavailability, demonstrating how improper message retention management could be exploited for DoS attacks that affect service availability [9].

Furthermore, new security challenges such as container escape have emerged as brokers are deployed in containerized environment. To address this threat, zero-trust architectures which are based on the principle "never trust, always verify" are utilized, enforcing stricter authentication and minimum-privilege policies [2].

Path traversal and arbitrary file access are also threats, especially when brokers fail to properly sanitize file paths in the configuration settings. For example, there is a memory leak in Mosquitto version 2.0.16, which attackers could exploit remotely. This happens when a client attempts to send many QoS 2 messages with duplicate IDs and does not respond to the PUBREC commands. The problem comes from not properly handling the EAGAIN error returned by the libc send function, which ends up causing the system to run out of memory. This vulnerability is listed in NVD with a CVE number (CVE-2023-28366), showing that attackers can manipulate broker configurations and specify file paths that result in unauthorized access to files. Such exploits can give adversaries access to read sensitive data or modify configurations, potentially threatening the entire MQTT infrastructure.

The security landscape of MQTT brokers has matured considerably, yet new vulnerabilities continue to emerge as attackers refine their techniques. Historical trends reveal a shift from early-phase issues, such as weak authentication and lack of encryption, toward more advanced threats, such as memory corruption and DoS attacks. The average severity and frequency of attacks have increased from medium in the early phase to high-critical in the recent phase, as shown in Table 6.1.

Table 6.1: Observed vulnerability trends across phases in MQTT Brokers.

Dimension	**Early Phase**	**Development Phase**	**Recent Phase**
Types	Configuration	Authentication, ACLs	Memory, DoS, Injection, Traversal
Frequency	Medium	Increasing	High (due to complexity)
Severity	Medium–High	High	High–Critical

These vulnerabilities highlight that, despite their widespread use and efficiency, MQTT brokers still remain susceptible to various exploitation methods.

6.2 RQ2: What are the critical differences in the vulnerability distribution across the major MQTT broker implementations and how can these differences inform more secure design choices?

We collect vulnerability data related to MQTT brokers from both NVD and Snyk.

6.2.1 Distributions of Vulnerabilities from NVD

The version information for each broker is listed in Table C.1. Using the broker names as search terms, we retrieved all corresponding vulnerabilities recorded in the NVD across all time periods. The NVD provides valuable information, including CVSS score, patches, and type descriptions. Specifically, for identifying vulnerability types, NVD listed CVE-IDs in the "Weakness Enumeration" section. The distribution of these types in the NVD is summarized in Figure 6.1.

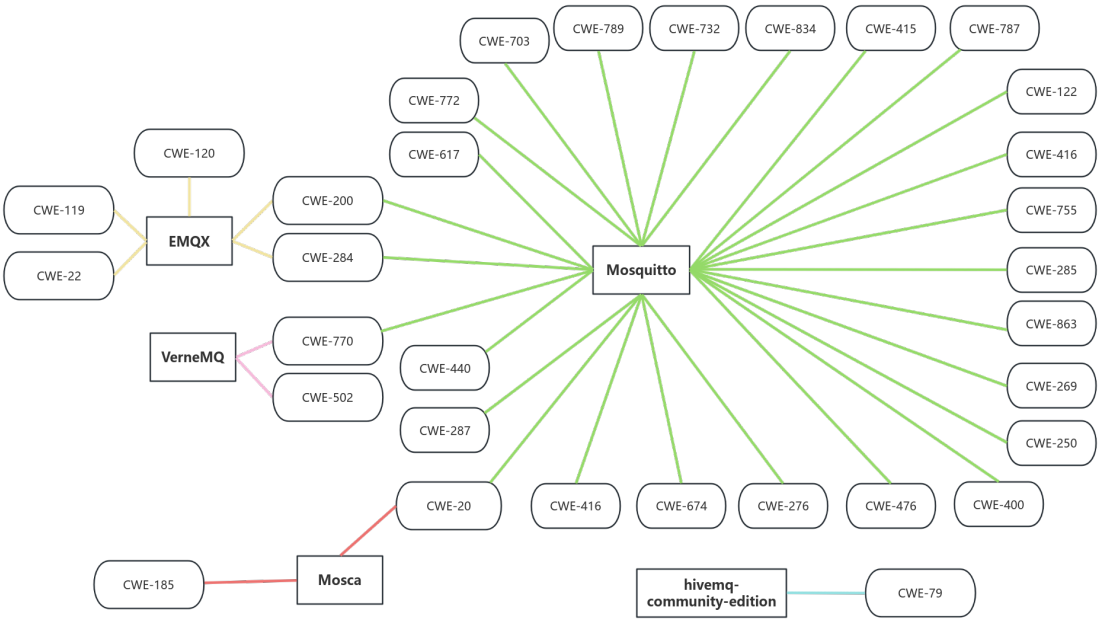


Figure 6.1: Distribution of vulnerabilities found in NVD, across five broker platforms. Rectangles represent broker names, circles represent vulnerabilities and the CWE inside circles represent types of vulnerabilities.

Mosquitto has the highest number of CWE-IDs, as seen in Figure 6.1, indicating that it is more likely to be affected by vulnerabilities. In contrast, the HiveMQ Community Edition has significantly fewer reported vulnerabilities than the other

brokers, with only CWE-79 listed in the NVD. Notably, it is the only broker that has no vulnerability in common with other brokers. Mosquitto shares CWE-200 and CWE-284 with EMQX, CWE-770 with VerneMQ, and CWE-20 with Mosca.

6.2.2 Vulnerability Distribution in Snyk Generated Reports

Figure 6.2 shows the distributions of vulnerabilities among five brokers (the core implementation of the broker, not the library). Among them, EMQX and the HiveMQ Community Edition have two vulnerabilities, which are less than other three brokers. In contrast, Mosquitto has eight distinct types of vulnerabilities. A notable recurring vulnerability is "Path Traversal", which three of the five brokers have this issue in common. This is reflected in our scoring system because if all conditions are equal, the vulnerability with higher frequency should have a higher priority score for detection than those with lower frequency.

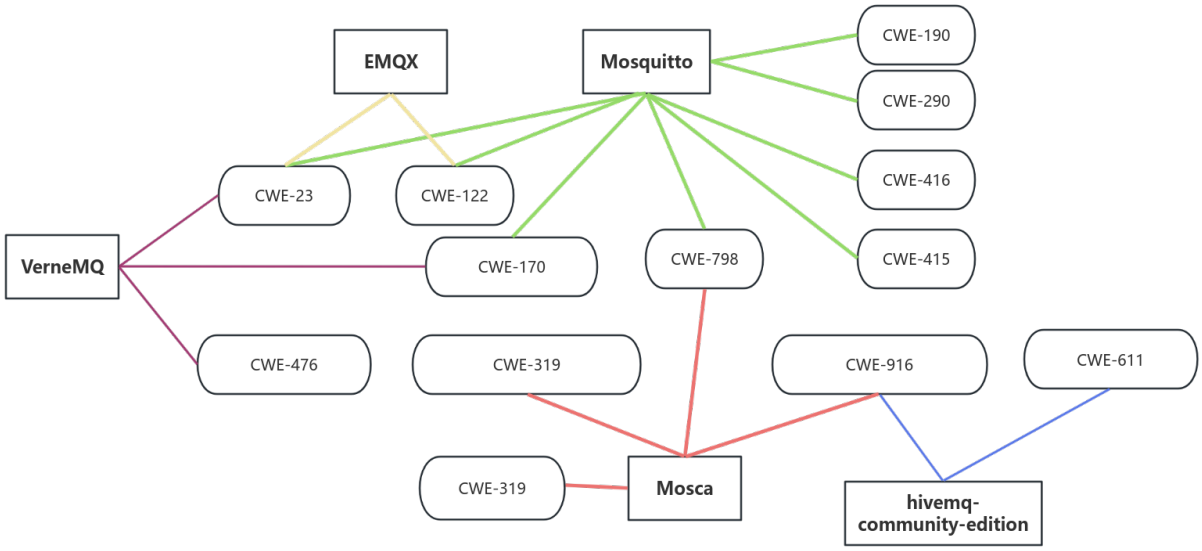


Figure 6.2: Distribution of vulnerabilities from Snyk generated report. Rectangles represent broker names, circles represent vulnerabilities.

Table 6.2 shows thirteen types of vulnerabilities and sixteen specific vulnerability instances. The specific types of CWE-ID are defined in the Terminology section. From the table, we observe that the CWE-23 (Path Traversal) appears in three brokers: Mosquitto, EMQX, and VerneMQ. Then CWE-122 (Buffer Overflow) is found in Mosquitto and EMQX, while CWE-170 (Improper Null Termination) is found in VerneMQ and Mosquitto. CWE-916 (Password Hash With Insufficient Computational Effort) also appears in two brokers: hivemq-community-edition and Mosca. All other vulnerabilities appear only once across all evaluated brokers. The distribution of vulnerabilities varies between the broker platforms, with some brokers populated with different types of vulnerability than others. Hence, the vulnerability distribution is a key factor that the proposed model incorporates when calculating the vulnerability score.

Table 6.2: Vulnerability distribution between platforms, their types and root cause.

CWE-ID	**Root Cause**	**Appears in brokers**
CWE-319	Cleartext transmission is caused by usage of HTTP protocol but not HTTPs.	Mosca
CWE-327	TLS validation is disabled because NODE_TLS_REJECT_UNAUTHORIZED is set to zero	Mosca
CWE-476	Possible NULL pointer from unvalidated return value.	VerneMQ
CWE-23	Unsanitized environment variable input to ‘fopen()‘.	Mosquitto, EMQX, VerneMQ
	Command line input flows into ‘open()‘ with no sanitization.	
	Command line input flows into ‘unlink()‘ with no sanitization.	
CWE-122	File input flows into ‘memcpy()‘ without bounds checking.	Mosquitto, EMQX
CWE-190	File input causes integer overflow via ‘+‘ operator.	Mosquitto
CWE-415	Possible double ‘delete‘ on the same pointer.	Mosquitto
CWE-798	Found hardcoded credential used in username.	Mosquitto, Mosca
	Found hardcoded credential used in connect.	
CWE-416	Use-after-free: freed string used in ‘strcmp()‘	Mosquitto
CWE-170	Missing null termination before ‘strlen()‘.	Mosquitto, VerneMQ
CWE-290	Hardcoded IP used in ‘strcmp()‘, can be spoofed.	Mosquitto
CWE-611	XML unmarshal allows XXE via external entity.	hivemq-community-edition
CWE-916	Use of insecure MD5 hash function.	hivemq-community-edition, Mosca

6.3 RQ3: How are the vulnerabilities in MQTT broker platforms related to the underlying protocol architecture and the third-party libraries they integrate?

The analysis of the broker platforms reveals that MQTT brokers are not standalone self-contained systems. They depend on the MQTT protocol architecture, primarily its publish-subscribe mechanism, as well as on third-party dependencies, including libraries, system-level packages for compression, authentication, and data parsing. Hence, the security of MQTT broker platforms is influenced by these two primary sources.

Architecture-Centric Vulnerabilities

MQTT is a lightweight protocol designed for devices with limited resources in IoT environments. However, its design lacks several essential built-in security features, such as enforced encryption or authentication mechanisms [1]. These architectural weaknesses compromise the fundamental rules of information security; CIA principles (Confidentiality, Integrity, and availability) of data in MQTT communications. The architectural-level vulnerabilities are discussed in detail in other sections of this thesis. Although these gaps necessitate the implementation of additional layers of security by MQTT brokers. This, in turn, introduces complex dependencies on external security libraries and utility packages, which is an issue explored in depth through the third research question (RQ3).

Vulnerabilities Tied to Third-Party Libraries

The Snyk scan of MQTT brokers uncovered several high-impact vulnerabilities in third-party components, often used indirectly through package chains. Below are a few illustrative examples from analysis of MQTT broker platforms.
One such vulnerability is DLL injection (CVE-2020-13110), which affects the Mosca Broker platform, as shown in the Appendix B.2, with a severity score of 8.4 (classified as high in CVSS). The broker platform itself does not implement Kerberos directly; its reliance on the "ascoltatori" component, which in turn depends on "Kerberos", introduces risk.
DLL Injection is a Windows-specific exploit technique that allows attackers to inject malicious code into trusted processes. Although the MQTT broker does not explicitly load untrusted DLLs, the vulnerability exists within a third-party authentication package, highlighting a supply chain risk.

Another critical vulnerability with a CVSS score of 9.8 is "Integer Overflow/Wraparound" as shown in the Appendix B.3, which is introduced to the EMQX broker by the "zlib/zlib1g" library. This is a system-level vulnerability that affects the "zlib" compression library. Although EMQX does not use zlib directly, it inherits the flaw

through the OS-level packages bundled in its docker environment (Docker image debian:12-slim).
This illustrates that vulnerabilities in containerized environments can extend beyond application code, affecting runtime libraries as well. Currently, no fix is available for this issue.

Regular Expression Denial of Service (ReDos) via Ansi-regex (CVE-2021-3807) is another vulnerability with a CVSS score of 7.5 (high) found in the Mosca broker platform.
This vulnerability is introduced to Mosca through transitive dependencies including ascoltatori, pino, levelup. This vulnerability can cause performance degradation or DoS through malicious input patterns that exploit slow regex evaluations. ReDoS vulnerabilities exploit inefficient regular expressions to freeze a process. These packages are widely used in Node.js ecosystems and often exist deep within dependency graphs, remaining unnoticed without automated scanning tools.

The examples above illustrate that security issues are not just about the broker source code; vulnerabilities can exist at several levels deep within the dependency tree of the MQTT broker platform. Indirect dependencies or transitive packages are often the weakest link. Table 6.3 lists some vulnerability types and the third-party library they originate from in detail.

Table 6.3: Summary of some vulnerabilities in MQTT brokers and the third-party library they originate from.

Vulnerability Type	**Source**	**Broker**	**CVSS Score**	**Risk Origin**
DLL Injection	kerberos via ascoltatori	Mosca	8.4 (High)	Application-level dependency
Integer Overflow / Wraparound	zlib1g in Docker base image	EMQX	9.8 (Critical)	OS-level library
Regular Expression DoS (ReDoS)	ansi-regex, semver	Mosca	7.5 (High)	Nested transitive dependencies

6.4 RQ4: What metrics significantly impact vulnerability prioritization and how can we design a better model that reflects the security of MQTT?

This section begins by discussing key metrics that significantly influence vulnerability prioritization. It then presents the rationale behind the design of a vulnerability scoring model tailored to the characteristics of MQTT Brokers. The model is then used to calculate real-world vulnerabilities, followed by a comparative evaluation against the scores calculated by the original CVSS. Finally, all modifications are summarized and visualized to provide a clear overview of the improvements.

In the original CVSS framework, the **Impact** and **Exploitability** are considered the two most critical metric groups. The Impact group reflects the severity of the consequences resulting from a successful exploitation, while the Exploitation group assesses how easily a vulnerability can be exploited. Together, these groups constitute the **Basic Score**, which serves as the foundation for assessing the overall severity of a vulnerability. Due to its fundamental role, many organizations recommend using it as the final CVSS score for the practical vulnerability assessment [45].

6.4.1 Designing an Improved Model to Reflect the Security of MQTT?

We begin by aiming to retain the most metrics of Impact and Exploitability because they represent the core dimensions of severity and availability of vulnerability. However, they do not fully capture the essential contextual characteristics in MQTT environments [13][15]. Based on the analysis conducted in this study, two additional metrics emerged as particularly important for effective vulnerability prioritization on MQTT broker platforms: **Frequency** and **Popularity**. Frequency refers to how often a specific type of vulnerability appears in public vulnerability databases, whereas popularity indicates the number of MQTT broker platforms affected by a given vulnerability.
These metrics were identified through the exploration of Research Questions 1 and 2, which revealed that vulnerabilities that are reported more frequently or affect multiple platforms pose a greater overall risk to the ecosystem.
Although the CVSS framework is robust, it lacks these contextual dimensions. As a result, it may not accurately prioritize vulnerabilities with broader implications across platforms or appear repeatedly over time. This gap highlights the need for a vulnerability scoring system that is more aligned with the operational realities of MQTT-based systems.
To address this limitation, we propose an improved vulnerability scoring model specifically tailored for MQTT brokers. The design of this enhanced model is motivated by the need to integrate the MQTT-specific context, namely frequency and popularity, while building upon the strengths of the existing CVSS framework. This section establishes the rationale for this improvement, while the following section

provides a detailed description of the structure of the model and the metric formulation. By incorporating additional context-aware metrics, the enhanced scoring model aims to provide an accurate and MQTT-specific evaluation of vulnerability severity, thus improving prioritization decisions and strengthening IoT security posture.

To overcome the limitations of the traditional CVSS scoring framework, a refined vulnerability scoring model is proposed, specifically designed for MQTT broker platforms. This model aims to deliver a more context-aware assessment by capturing not only the intrinsic characteristics of vulnerabilities, but also by integrating real-world factors such as remediation availability, frequency, and distribution of vulnerabilities across broker platforms. Frequency refers to how often a particular vulnerability is observed in public vulnerability databases, while popularity accounts for the number of broker platforms affected by the same issue. These two factors were identified as significant through the results of Research Questions 1 and 2, which demonstrated that vulnerabilities that are more frequent and widely distributed pose greater systemic risks and thus warrant higher prioritization.

Beyond these metrics, some adjustments are also made. In CVSS, Scope is grouped under Exploitability; however, due to its conceptual overlap with the Confidentiality, Integrity, and Availability (CIA) metrics, it has been moved to the Severity group. This modification ensures that Scope contributes more directly to the evaluation of a vulnerability's overall impact, rather than its likelihood of exploitation.
In simplifying the model, the Attack Vector metric has also been revised. Instead of the original four categories defined in CVSS, the proposed model consolidates these into two categories: Online and Local. This simplification reduces ambiguity and improves consistency, particularly in scenarios where scoring decisions need to be made quickly or at scale. Additionally, the Attack Complexity metric has been removed entirely, as its subjective interpretation often introduces inconsistencies and increases the likelihood of scoring errors, especially among less experienced assessors.
Another important addition is the introduction of a new metric called Fixability, which reflects how easily a vulnerability can be addressed once discovered. The presence of an official patch or remediation measure typically reduces the severity of a vulnerability in practice, and incorporating this into the model enables a more nuanced evaluation of security risks, especially in environments where patch management plays a critical role in system resilience.
Together, these refinements create a scoring framework that extends beyond static severity assessment and moves toward a more dynamic and MQTT-specific understanding of risk. Rather than treating all vulnerabilities with the same structural weight, the model emphasizes their practical implications, operational impact, and likelihood of recurrence across platforms. This enables developers, researchers, and security teams to make more informed decisions about which vulnerabilities require immediate attention and which can be prioritized based on contextual risk. The complete structure of the proposed scoring model, including the revised and added metrics, is visualized in Figure 6.3, which illustrates how each element contributes to the final severity score.

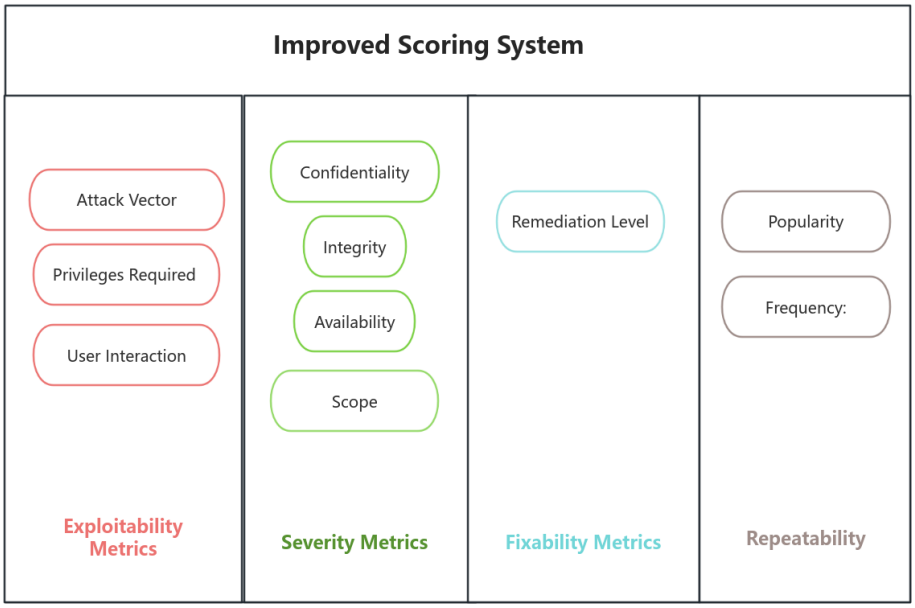


Figure 6.3: Metrics of improved scoring system

Vulnerabilities are classified into different severity levels based on their final scores. Higher levels generally require higher priority to be fixed. The severity levels in the new model are aligned with the levels defined in traditional CVSS, as shown in Table 4.2. However, a notable modification is the "Critical" level, which was adjusted to accommodate the extended scoring range. The maximum score in the new model may exceed ten, due to the inclusion of new context-specific metrics. Consequently, all vulnerabilities with scores above nine are classified as "Critical" level. Table 6.4 provides detailed information on the levels.

Table 6.4: Vulnerability levels of Improved Scoring System and numerical value.

Score	**Vulnerability Levels**
0	Safe
0 - 3.9	Low
4 - 6.9	Medium
7 - 8.9	High
Greater than 9	Critical

To enhance clarity and reduce complexity, abbreviations will be used in place of full metric names throughout the formulas. These abbreviations, along with their definitions, are presented in the Terminology table. The following sections provide a detailed explanation and rationale for the modifications introduced to each group of metrics in the improved vulnerability scoring system.

6.4.1.1 Severity

Severity reflects the level of risk once the vulnerability is successfully exploited. In the proposed model, severity is assessed using four metrics: confidentiality (C), integrity (I), availability (A), and scope (S). Confidentiality refers to the degree to which sensitive information may be exposed, integrity indicates the extent to which data may be modified or tampered with, and availability represents the likelihood of disruption or denial of access to services or systems. Scope measures whether the impact of a vulnerability extends beyond the security permissions of its initial context. For example, in Linux, a vulnerability in the user level could impact the superuser level, or a vulnerability within one user group can impact other user groups.

In CVSS, the impact metric contains confidentiality, integrity, and availability, which are core metrics to evaluate the severity of vulnerabilities. In the latest version, CVSS v4.0, Scope is split into three metrics, "Subsequent System Confidentiality", "Subsequent System Integrity", and "Subsequent System Availability", shown in Figure 6.4. This change was introduced to eliminate the ambiguity of the Scope [51]. Additionally, Scope is moved from Exploitability to Impact. This change reflects that Scope, like confidentiality, integrity, and availability, is an inherent property of vulnerability and does not change over time; hence, it is now categorized as Impact rather than Exploitability.

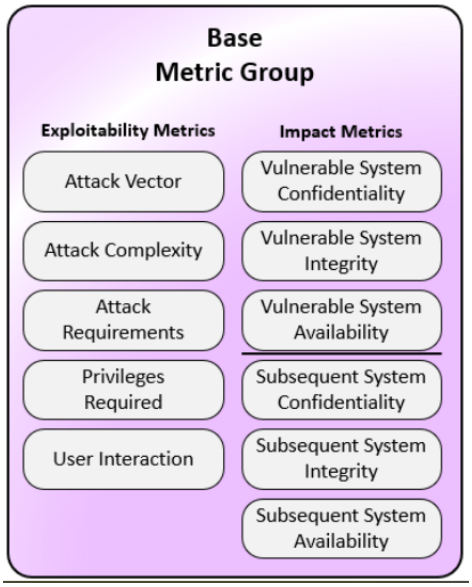


Figure 6.4: Replaced Scope in CVSS 4.0 [15]

The severity metrics values are set as shown in Tables 6.5, 6.6, 6.7, and 6.8. These values are the same as the corresponding setting in CVSS.

Equation (4.1) contains C, I and A, we add the fourth element, "scope" and get Equation (5.1). If all metric of C, I, A is equal to zero, then the severity score is equal to zero, which means that the vulnerability cannot leak and modify the data

Table 6.5: Scoring of Confidentiality (C)

Level	Ratio
High	0.56
Low	0.22
None	0

Table 6.6: Scoring of Integrity (I)

Level	Ratio
High	0.56
Low	0.22
None	0

Table 6.7: Scoring of Availability (A)

Level	Ratio
High	0.56
Low	0.22
None	0

Table 6.8: Scoring of Scope (S)

Level	Ratio
Exceeded (E)	0.56
UnExceeded (UE)	0.22

and also does not affect the system's availability, that is, causing crashes, slowing down performance, or causing services to temporarily or permanently stop.

If only Availability (A) is zero, it means that the system's availability is not affected, but there is still the possibility of data leakage or modification. If only C is zero, this means that the data is not leaked but can be modified, and the system availability is affected. If only I is zero, that is, the data cannot be modified, but the data may leak and system's availability is also possibly affected.

$$\text{Severity Score} = \begin{cases} 0 & \text{If C, I, and A are all zero} \\ 1 - (1 - C) \times (1 - I) \times (1 - A) \times (1 - S) & \text{else} \end{cases} \tag{5.1}$$

6.4.1.2 Exploitability

In the proposed model, the exploitability metrics retain their definitions as in CVSS. The privileges required (PR) and user integration (UI) metrics remain unchanged, while the attack complexity (AC) is removed, and the values of the attack vector (AV) are revised.

The attack complexity metric was removed because it is vague and redundant, as noted in [24], where the authors aimed to reduce the number of metrics to avoid the calculation errors caused by incorrect user input. They use a decision tree algorithm to classify and identify redundant metrics, finding that S and AC have higher redundancy and can be removed. Following this rationale, the proposed model omits the AC metric to maintain consistency and reduce complexity.

Table 6.9: Attack vector(AV) metric and its four options according to CVSS [13]

Option	Description
Network	The vulnerability is accessible for the entire the Internet.
Adjacent	The vulnerability is accessible for the Internet, but attack is only logically adjacent in topology
Local	The vulnerability is cannot be exploited remotely.
Physical	Attackers can only exploit the vulnerability offline by physical impact, for example touch or high temperature.

Table 6.9 shows four options for the attack vector in CVSS. As noted in [51], the authors highlight inconsistencies in the value setting, especially between "Network" and "Adjacent". For example, different programmers might assign different values in the attack vector (AV) for the same vulnerability. To address this inconsistency, we merge "Network" and "Adjacent" into a single category labeled "Online", associated with the network stack. Similarly, we merge "Local" and "Physical" into a single category labeled "Local", because vulnerabilities in these categories cannot be exploited remotely. As a result, the Exploitability group in the proposed model consists of three metrics: PR, AV and UI. The possible values for these three metrics are listed in Table 6.10, 6.11 and 6.12. The metric values at each level are consistent with those defined in CVSS.

Table 6.10: Values of PR

Level	Ratio
None	0.85
Low (L)	0.62
High (H)	0.27

Table 6.11: Values of AV

Level	Ratio
Local	0.55
Online	0.88

Table 6.12: Values of UI

Level	Ratio
None	0.85
Required (R)	0.62

Similar to Equation (4.3), the Exploitability score is calculated as a constant plus three configurable factors.

$$\text{Exploitability Score} = \text{PR} \times \text{AV} \times \text{UI} \tag{5.2}$$

6.4.1.3 Fixability

The Report Confidence (RC) metric was removed from the Temporal Metrics group because, in the proposed model, all vulnerabilities are identified and confirmed by the trusted scanning tool and the trusted database before being recorded. This confirmation process ensures the validity of each reported vulnerability and reduces the uncertainty about its existence. Therefore, the RC metric, which normally indicates confidence in the accuracy of a reported vulnerability, is unnecessary in this context. Removing RC simplifies the computation and evaluation procedure while maintaining the reliability of the vulnerability data.

Furthermore, the Exploit Code Maturity (ECM) metric was removed due to the difficulty in reliably assessing the maturity level of the exploit code. As shown in Figure 6.5, assigning a value requires the evaluator to compare "information regarding the availability of exploitation code/processes and the state of exploitation techniques" [12], with the corresponding data in the figure. However, this information must be manually collected by users, which can result in incomplete coverage of real-world scenarios. In addition, such information may not be updated in a timely manner, potentially leading to outdated or inconsistent assessments. To minimize such uncertainties and maintain consistency in the assessment, this metric was removed from our proposed vulnerability scoring model.

Metric Value	Description
Not Defined (X)	Reliable threat intelligence is not available to determine Exploit Maturity characteristics. This is the default value and is equivalent to Attacked (A) for the purposes of the calculation of the score by assuming the worst case.
Attacked (A)	Based on available threat intelligence either of the following must apply: Attacks targeting this vulnerability (attempted or successful) have been reported Solutions to simplify attempts to exploit the vulnerability are publicly or privately available (such as exploit toolkits)
Proof-of-Concept (P)	Based on available threat intelligence each of the following must apply: Proof-of-concept exploit code is publicly available No knowledge of reported attempts to exploit this vulnerability No knowledge of publicly available solutions used to simplify attempts to exploit the vulnerability (i.e., the "Attacked" value does not apply)
Unreported (U)	Based on available threat intelligence each of the following must apply: No knowledge of publicly available proof-of-concept exploit code No knowledge of reported attempts to exploit this vulnerability No knowledge of publicly available solutions used to simplify attempts to exploit the vulnerability (i.e., neither the "POC" nor "Attacked" values apply)

Figure 6.5: Exploit Maturity and its setting value [12]

Furthermore, the Remediation Level (RL) reflects the availability of fixes and is kept in the proposed scoring system. We follow the RL setting as in CVSS and the metric value is shown in Table 6.13.

The remediation level indicates how fixable the vulnerability is, and the overall remediability factor is represented in Equation (5.3).

$$\text{Remediability Factor} = \text{Remediation Level} \tag{5.3}$$

Table 6.13: Scoring of Remediation Levels

Remediation Level	Description	Score
OfficialFix	Official patch is available to completely fix.	0.95
Temporary Fix	The official fix is temporary but not permanent.	0.96
Workaround	There is an unofficial solution available.	0.97
Unavailable	There is no solution available.	1

6.4.1.4 Repeatability

Repeatability is a new group of metrics that is incorporated in the proposed model and contains two aspects: **Frequency of Recent Exploitated Activity** whether the vulnerability was actively exploited recently. **Extent of Distribution** the extent of its distribution in MQTT broker implementations.

These two metrics are incorporated because a higher level of **Recent Exploitation Activity** indicates that the vulnerability has been more actively and frequently targeted by attackers in recent times. This not only suggests a higher likelihood that the vulnerability is exploited again, but also reflects the current lack of effective defense strategies against it. For **Extent of Distribution**, a higher score indicates that vulnerability is more widely present across different MQTT brokers. This implies a broader impact and a greater risk of attack, potentially affecting multiple systems, platforms, and even entire industries.

Traditional CVSS does not explicitly account for these factors through specific metrics. While CVSS includes Temporal Metrics, they are used to measure the current state of remediation techniques rather than the vulnerability itself. Furthermore, CVSS includes "Environmental Metrics" allowing users to adjust the basic impact values based on their environment. However, it does not introduce new metrics related to frequency or distribution.

Thus, in the proposed model, frequency and popularity are incorporated to represent the extent of a vulnerability's temporal and spatial distribution. **Frequency** is defined as the number of times a specific type of vulnerability was recorded in the NVD in a period of one month. **Popularity** is defined as the number of brokers (of 5 brokers analyzed in this thesis) affected by the same vulnerability in a period of one month. Initially, values close to one are assigned to these metrics to minimize their impact on the final score, as their influence may not be clear at the beginning. These values can be adjusted later if necessary. The specific metric values are presented in Tables 6.14 and 6.15.

Table 6.14: Scoring of Frequency

Frequency	**Description**	**Score**
High (H)	Over 100	1.2
Medium (M)	Over 40	1.15
Low (L)	Over 20	1.1
Very Little (VL)	1 or more than 1	1.05
None	0	1

Table 6.15: Scoring of Popularty

Popularity	**Description**	**Score**
High (H)	4 or 5	1.2
Medium (M)	3	1.15
Low (L)	2	1.1
Very Little (VL)	1	1.05
None (N)	0	1

The formula for calculating repeatability is shown in Equation (5.4).

$$\text{Repeatability Factor} = \text{Popularity} \times \text{Frequency} \tag{5.4}$$

The final score is calculated using Equation (5.5). If any of the multiplicative factors is zero, the resulting score will also be zero. In CVSS, the maximum score is 10; thus, a constant value of 16.3407 is used in the calculation to ensure that the final score does not exceed this limit in the absence of additional metrics.

$$\begin{aligned}\text{Final Score} = 16.3407 \times \text{Severity Factor} \times \text{Exploitability Factor} \\ \times \text{Remediability Factor} \times \text{Repeatability Factor}\end{aligned} \tag{5.5}$$

6.4.2 Key Differences Between the New Model and CVSS

The equations and metric groups used in the proposed vulnerability scoring model have been discussed in detail in previous sections. This section summarizes all modifications made to the original CVSS framework, presenting a side-by-side comparison with the proposed model 6.16. To enhance clarity, Figure 6.6 illustrates the key improvements and structural differences introduced in the proposed model. We also show the computation of Scope in the original CVSS and the proposed model.

Changes In Metric Setting

Figure 6.6 visualizes all the changes introduced in the proposed model. Steps 1 to 6 correspond to the specific improvements listed in rows 1 to 6 of Table 6.16.

Table 6.16: Limitations of CVSS and improvements made in the proposed model

No.	Limitations of CVSS	Improvements in the proposed model
1	AC is redundant [24].	Removed AC
2	Attack Vector is hard to set and lead to inconsistent score [51].	Metrics was simplified by consolidating its four options into two.
3	Exploit Code Maturity (ECM) is hard to test	Removed ECM.
4	Report Confidence is not necessary.	Removed Report Confidence.
5	Scope is used as condition of computing (4.2).	Moved Scope into impact metrics.
6	Frequency and Popularity are not considered.	Introduced Frequency and Popularity in the proposed model.

We removed Attack Complexity (AC), Remediation Level (RC) and Exploit Code Maturity (ECM). The Attack Vector (AV) metric was simplified by consolidating its four options into two. In addition, Scope(S) was reclassified from Exploitability group to Severity group. Finally, the proposed model introduces two new context-specific metrics: Frequency and Popularity, which address critical gaps in assessing vulnerabilities within MQTT environments.

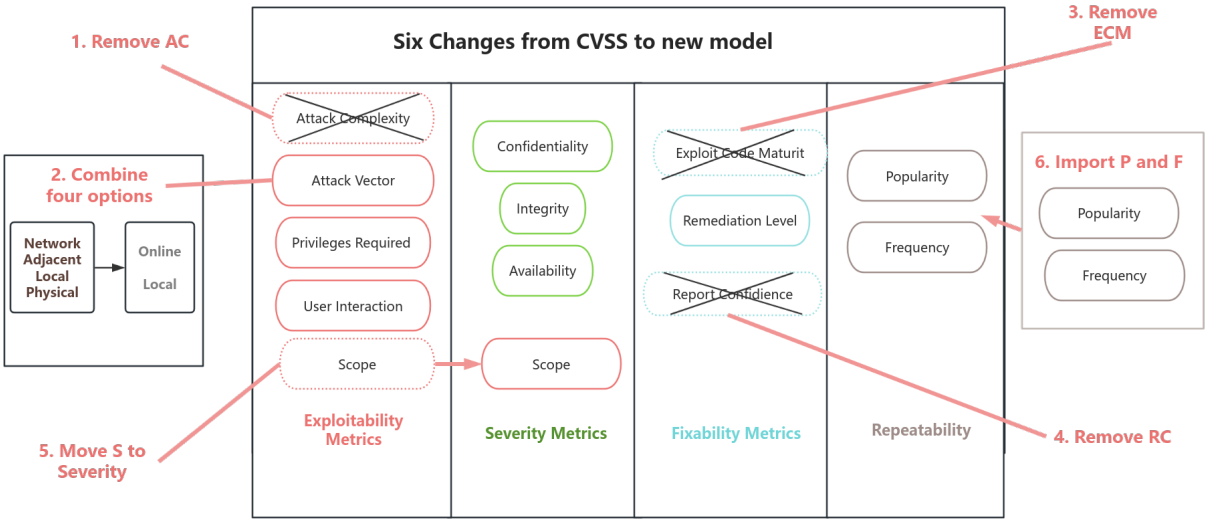


Figure 6.6: Visualization of Six significant changes listed in the table 6.16

Changes In Computation

The modification of metrics in the proposed model results in corresponding changes to the computation process. In particular, there are some differences in how scores are calculated compared to the original CVSS. One key change involves the treatment of Scope metric. In Equation (4.2), Scope is used as a condition in the calculation of the impact score. In CVSS v4.0, Scope has been integrated into severity metrics, along with Confidentiality (C), Integrity (I), and Availability (A). Following this approach, we also reclassified Scope from the Exploitability group to the Severity group. This adjustment is reflected in Formula (5.1), which illustrates how Equations (4.1) and (4.2) are combined in the revised scoring model after the reclassification of Scope.

$$\text{ISC}_{\text{Base}} = 1 - (1 - C)(1 - I)(1 - A) \tag{4.1}$$

$$\text{Impact Score} = \begin{cases} 6.42 \times \text{ISC}_{\text{Base}}, & \text{if Scope} = \text{Unchanged} \\ 7.52 \times (\text{ISC}_{\text{Base}} - 0.029) - 3.25 \times (\text{ISC}_{\text{Base}} - 0.02)^{15}, & \text{if Scope} = \text{Changed} \end{cases} \tag{4.2}$$

$$\text{Severity Score} = \begin{cases} 0, & \text{if C, I, and A are all zero} \\ 1 - (1 - C)(1 - I)(1 - A)(1 - S), & \text{else} \end{cases} \tag{5.1}$$

In Equation (5.1), if the values of C, I, and A are all zero, then the severity score becomes zero. This behavior is consistent with the condition Impact<=0 in Equation (4.4).

In Equation (4.4), CVSS discuss different situations based on the values of Impact and Scope. However, in the proposed model it is only discussed once as in (5.1). The final score is derived using a simple multiplication-based linear equation (see (5.5)).

$$\text{Base Score} = \begin{cases} 0, & \text{if Impact} \leq 0 \\ \text{RoundUp}(\min(\text{Impact} + \text{Exploitability}, 10)), & \text{if Scope} = \text{Unchanged} \\ \text{RoundUp}(\min(1.08 \times (\text{Impact} + \text{Exploitability}), 10)), & \text{if Scope} = \text{Changed} \end{cases} \tag{4.4}$$

$$\begin{aligned} \text{Final Score} = 16.3407 \times \text{Severity Factor} \times \text{Exploitability Factor} \\ \times \text{Remediability Factor} \times \text{Repeatability Factor} \end{aligned} \tag{5.5}$$

6.4.3 Calculating Vulnerability Score with proposed Vulnerability Scoring Model

To clarify the process, this section calculates the severity of some selected vulnerabilities using the proposed model and subsequently compares the result with those from the traditional CVSS vulnerability scoring system.

6.4.3.1 Path Traversal

Path traversal is a vulnerability found in the Mosquitto broker platform and is selected as an example to calculate the vulnerability score using the improved scoring system. In Figure 6.7, the top-left corner shows the report of a path traversal vulnerability in Mosquitto broker, while the top-right corner shows how the vulnerability is manifested in the source code. Lines 519 and 523 show that the function pointer "fptr" is used to store the file path. The lower portion of the figure reveals that the variable “optional_file” is determined by the second part of the command-line instruction entered by the user. For example, if a user enters “./my_program ../../etc/passwd”, the segment “../../etc/passwd” may expose the password information or any information depending on the user’s input. This behavior exemplifies a classic path traversal vulnerability, where an attacker can manipulate input to access unintended files or directories.

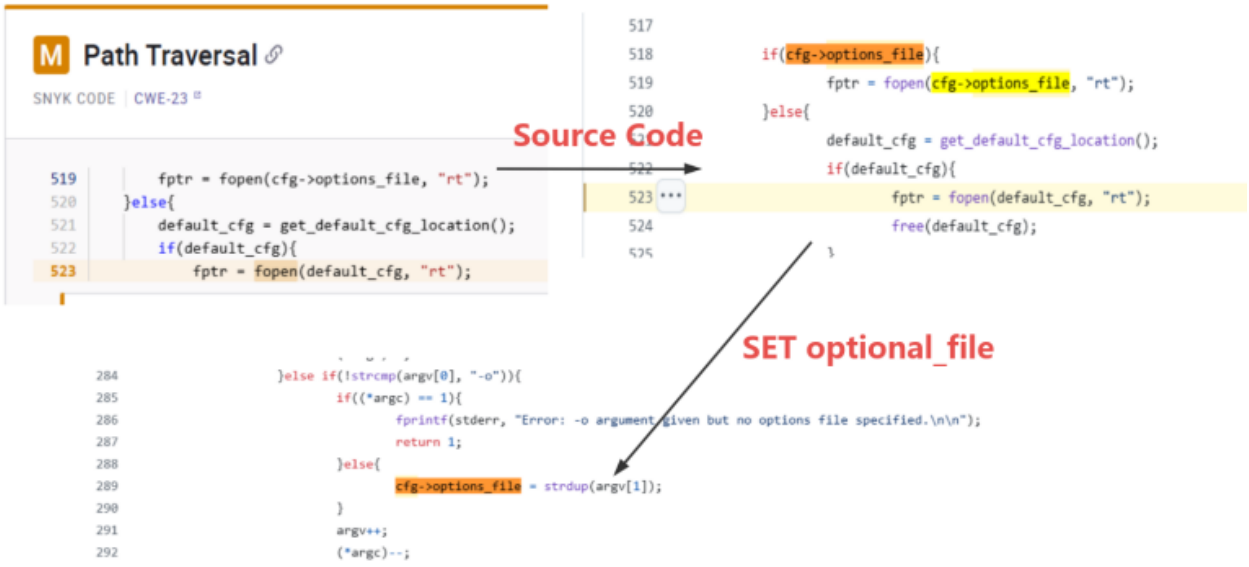


Figure 6.7: Path Traversal in mosquitto, and where in the source code Path Traversal could be exploited

We found similar cases as CVE-2019-10743, where attackers used specially crafted ZIP archive files (including path traversal strings, like "../../file.exe".). When these files are decompressed, they may extract executable content into any directory, potentially leading to serious security breaches. Another example is CVE-2022-45918, where attackers can request previous logs and manipulate the query parameter with “../../../../../../../../../etc/passwd” to access and read arbitrary files on the server.

The above two cases and their corresponding CVSS configurations and scores can be found in the NVD. Detailed metric configurations for these vulnerabilities are shown in Table 6.17 and 6.18.

Table 6.17: Setting values of CVE-2019-10743, in[34], CVSS Version 3x

Metrics	**Setting**
AV	Local
AC	Low
PR	None
UI	Required
S	UnExceeded
C	None
I	High
A	None

Table 6.18: Setting values of CVE-2022-45918, in[33], CVSS Version 3x

Metrics	**Setting**
AV	Network
AC	Local
PR	Low
UI	None
S	UnExceed
C	High
I	None
A	None

The above two cases were used to validate our proposed model. Table 6.19 shows the configuration of the Exploitability and Severity metrics in the improved scoring model.

For the AV metric, the attack command is sent to the server remotely, so it should be classified as “Online”, while for the PR metric, our case closely resembles CVE-2019-10743. In CVE-2019-10743, the compressed file is automatically decompressed and executed on the server side, requiring no special privileges beyond the ability to upload the file. Similarly, in our case, the attacker only needs to modify the command input, which is then automatically parsed by the server. Therefore, the PR metric is set to “None”.

In terms of UI metric, CVE-2022-45918 is marked as “Required” because there is a prerequisite: “the SCORM debugger must be enabled for the entire ILIAS platform.” [4]. However, in our case, as long as the server is running, it can parse the instructions without any user interface. Therefore, the UI is set to “None”.

For the S metric, in our case, the impact of the vulnerability only influences the components of the server, so it is set to “UnExceeded”. Metric C focuses on whether the data are exposed, while Metric I focuses on whether the data are modified. Therefore, in CVE-2019-10743, the decompressed malicious executable file could potentially modify system files, so its integrity is “High” and confidentiality is “None”.

In contrast, CVE-2022-45918 has a high risk of exposing information to the attacker, so its I metric is “None” and C metric is “High”. In our case, as seen in line 519, the “str” pointer has only “read” permission. Therefore, both I and A are set to “None”. However, C is set to “High” because an attacker could input a command such as “xxxxx /../pwd xxx”, which could result in unauthorized file reads (for example, from “/../pwd” path).

Table 6.19: Exploitability and Severity Metric Setting

Metrics	Setting
AV	Online
PR	None
UI	None
S	UnExceeded
C	High
I	None
A	None

Table 6.20: Fixability and Repeparability Metric Setting

Metrics	Setting
RL	OfficialFix
F	High
P	Medium

Table 6.20 shows the configuration of Fixability and Repeatability. For Fixability, we consider whether there is an official method or patch to fix. In [17], the authors propose an algorithm for building a whitelist, a set of allowed paths or resources, to prohibit unauthorized access. Similarly, the Synk report [44] recommends maintaining a set of permitted file system paths and comparing the user input against that set. Based on these recommendations, we set the RL metric to "Official Fix".

For the P metric, we consider the appearance of vulnerability type in different broker platforms. For path traversal, it appears on three broker platforms; therefore, we set the popularity to "Medium".

Finally, for the F metric, we perform an exact match search for the term "path traversal" in the NVD. We tracked the number of occurrences of the "path traversal" over a period of one month (20 March to 20 April) and it occurred 108 times (Listed in Appendix, from figureD.1 to D.11). Thus, the frequency is set to "High". With all metrics defined, we now substitute the metrics values in formula (5.5) to compute the final vulnerability score, as shown in Equation (5.6):

$$\begin{aligned}\text{Final Score} = 16.3407 \times \text{Severity Factor} \times \text{Exploitability Factor} \times \text{Remediability} \\ \text{Factor} \times \text{Repeatability Factor} = 16.3407 \times (1 - (1 - C) \times (1 - I) \\ \times (1 - A) \times (1 - S)) \times PR \times AV \times UI \times RL \times P \times F = 16.3407 \\ \times (1 - (1 - 0.56) \times (1 - 0) \times (1 - 0) \times (1 - 0.22)) \times 0.85 \\ \times 0.88 \times 0.85 \times 0.95 \times 1.15 \times 1.2 = 8.94596 \end{aligned} \tag{5.6}$$

Thus, in our proposed scoring model, the final vulnerability score is 8.94596, and the severity level is "High". In contrast, the same vulnerability receives a score of 6.5 with a severity level of "Medium" under the traditional CVSS [33]. This shows that the proposed vulnerability scoring model effectively captures the risk level. A detailed discussion of this comparison is provided in the Evaluation section.

6.4.3.2 BufferOverflow

According to the report generated by Snyk, the Mosquitto broker contains a buffer overflow vulnerability. We evaluated this vulnerability using our proposed vulnerability scoring model. Figure 6.8 shows the source code from the Mosquitto GitHub repository, where on line 72, the function memcpy() copies data from the buffer "buf" to the memory region "cfg.message[pos]", with "rlen" specifying the number of bytes. Line 62 limits "rlen" to a maximum of 1024 bytes.

```
53  int load_stdin(void)
54  {
55          size_t pos = 0, rlen;
56          char buf[1024];
57          char *aux_message = NULL;
58
59          cfg.pub_mode = MSGMODE_STDIN_FILE;
60
61          while(!feof(stdin)){
62                  rlen = fread(buf, 1, 1024, stdin);
63                  aux_message = realloc(cfg.message, pos+rlen);
64                  if(!aux_message){
65                          err_printf(&cfg, "Error: Out of memory.\n");
66                          free(cfg.message);
67                          return 1;
68                  } else
69                  {
70                          cfg.message = aux_message;
71                  }
72                  memcpy(&(cfg.message[pos]), buf, rlen);
73                  pos += rlen;
74          }
75          if(pos > MQTT_MAX_PAYLOAD){
76                  err_printf(&cfg, "Error: Message length must be less than %u bytes.\n\n", MQTT_MAX_PAYLOAD);
77                  free(cfg.message);
78                  return 1;
79          }
```

Figure 6.8: Souce code of Mosquitto broker platform where it leads to buffer overflow.

In each iteration of the while loop, "realloc" is used to expand the "cfg.message" buffer to accommodate new data. However, the while loop (lines 61 to line 74) lacks a check to ensure that the total size of "cfg message" does not exceed "MQTT_MAX _PAYLOAD" limit.

Although there is a conditional check "if condition", it is not enough as it is executed after the while loop, by which point the data may already have been written to the file system. As a result, if the input exceeds the length limit, it can cause memory corruption, leading to a potential risk of buffer overflow vulnerability.

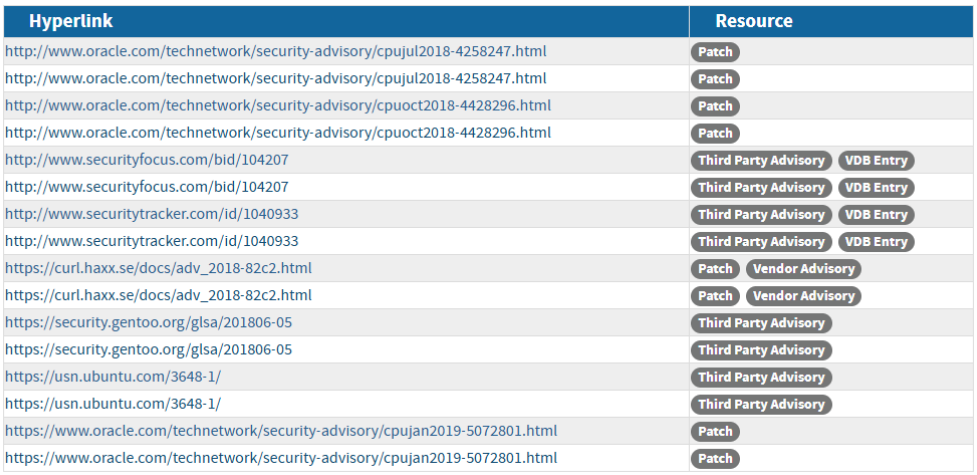
Before computing the final vulnerability score, each metric must be correctly configured. To guide this process, we refer to CVE-2018-1000300 from the NVD [32], which has a CVSS score of 8.8 and is classified as "High" severity level. This vulnerability involves a buffer overflow in the "curl" command. Specifically, during FTP (File Transfer Protocol) transfers, "curl" internally retains a "closure handle"

to manage the termination of FTP connections. However, responses from the FTP server may exceed the default buffer length: 16KB, which potentially results in buffer overwriting in the buffer allocated in the closure handle [39].

Table 6.21: CVE-2018-1000300 Metric Setting [32],CVSS version 3x

Metrics	Setting
AV	Network
AC	Local
PR	None
UI	None
S	UnExceeded
C	High
I	High
A	High

Attackers may be able to write arbitrary content in memory, leading to possible data tampering, data leakage, or denial of service. Therefore, the I, C, and A metrics are set to "High". Table 6.21 shows the CVSS metric configuration, based on which we can define the values of AV, PR, UI, S, C, I and A for the Mosquitto broker, as shown in Table 6.22.



Hyperlink	Resource
http://www.oracle.com/technetwork/security-advisory/cpujul2018-4258247.html	Patch
http://www.oracle.com/technetwork/security-advisory/cpujul2018-4258247.html	Patch
http://www.oracle.com/technetwork/security-advisory/cpuoct2018-4428296.html	Patch
http://www.oracle.com/technetwork/security-advisory/cpuoct2018-4428296.html	Patch
http://www.securityfocus.com/bid/104207	Third Party Advisory VDB Entry
http://www.securityfocus.com/bid/104207	Third Party Advisory VDB Entry
http://www.securitytracker.com/id/1040933	Third Party Advisory VDB Entry
http://www.securitytracker.com/id/1040933	Third Party Advisory VDB Entry
https://curl.haxx.se/docs/adv_2018-82c2.html	Patch Vendor Advisory
https://curl.haxx.se/docs/adv_2018-82c2.html	Patch Vendor Advisory
https://security.gentoo.org/glsa/201806-05	Third Party Advisory
https://security.gentoo.org/glsa/201806-05	Third Party Advisory
https://usn.ubuntu.com/3648-1/	Third Party Advisory
https://usn.ubuntu.com/3648-1/	Third Party Advisory
https://www.oracle.com/technetwork/security-advisory/cpujan2019-5072801.html	Patch
https://www.oracle.com/technetwork/security-advisory/cpujan2019-5072801.html	Patch

Figure 6.9: NVD page for Buffer Overflow official patches

CVSS does not include metrics for "Fixiability" and "Repeatability", therefore, these values must be assigned based on their definitions. Table 6.23 shows the setting of RL, F and P. The NVD page [32] has listed many official patches, as shown in figure 6.9, therefore, we set it as "Official Fix".
The metric P is set to "Low", since two brokers (EMQX and Mosquitto) contain buffer overflow. To define the metric F, we conducted a targeted searched for "Buffer

Overflow" on NVD's search engine and found 183 items from 20 March to 20 April (in Appendix 5, from Figure E.1to E.21), therefore, it was set to "Critical".

Table 6.22: Exploitability and Severity Setting of Buffer Overflow

Metrics	Setting
AV	Online
PR	None
UI	None
S	UnExceeded
C	High
I	High
A	High

Table 6.23: Fixability and Repearability Metric Setting

Metrics	Setting
RL	OfficialFix
F	High
P	Low

Substituting the values into the Equation (5.5), we can calculate the final score for the buffer overflow vulnerability, which is 12.1627, the level is "Critical".

6.4.3.3 Double Free

In the Mosquitto broker platform, **Double Free** is another type of vulnerability, as seen in the source code in Figure 6.10. It is programmed in C++, where the "delete" operator is used to release the memory space previously allocated using function "new()". However, in lines 31 and 33, delete is called twice on the same pointer "mosq". After the first "delete", "mosq" becomes a wild pointer. The second call to the "delete()" function to release the same memory will result in undefined behavior, as "mosq" may point to an arbitrary location in memory, which can lead to issues such as segmentation faults or program crash.

```
17      {
18              struct mosquittopp_test *mosq;
19
20              int port = atoi(argv[1]);
21
22              mosqpp::lib_init();
23
24              mosq = new mosquittopp_test("01-no-clean-session", false);
25
26              mosq->connect("localhost", port, 60);
27
28              while(run == -1){
29                      mosq->loop();
30              }
31              delete mosq;
32
33              delete mosq;
34              mosqpp::lib_cleanup();
35
36              return run;
37      }
```

Figure 6.10: Mosquitto source Code that leads to Double Free vulnerability

To calculate the vulnerability score with our proposed model and define the appropriate metrics values, we examined a "Double Free" example from the NVD (CVE-2022-49541), which has a vulnerability score of 7.8 according to traditional CVSS [35]. Figure 6.11 shows the relevant source code related to CVE-2022-49541 and the corresponding patch implementation [41].

The red-highlighted section in Figure 6.11 shows the vulnerable code that leads to "Double Free" vulnerability. Specifically, the super block "sb" invokes a destructor function in its associated memory, including "cifs_sb". However, the "if" condition attempts to free "cifs_sb" again, which leads to Double Free.

```
@@ -838,7 +838,7 @@ cifs_smb3_do_mount(struct file_system_type *fs_type,
              int flags, struct smb3_fs_context *old_ctx)
 {
        int rc;
-       struct super_block *sb;
+       struct super_block *sb = NULL;
        struct cifs_sb_info *cifs_sb = NULL;
        struct cifs_mnt_data mnt_data;
        struct dentry *root;
@@ -934,9 +934,11 @@ out_super:
        return root;
 out:
        if (cifs_sb) {
-               kfree(cifs_sb->prepath);
-               smb3_cleanup_fs_context(cifs_sb->ctx);
-               kfree(cifs_sb);
+               if (!sb || IS_ERR(sb)) {  /* otherwise kill_sb will handle */
+                       kfree(cifs_sb->prepath);
+                       smb3_cleanup_fs_context(cifs_sb->ctx);
+                       kfree(cifs_sb);
+               }
        }
        return root;
 }
```

Figure 6.11: Patch of CVE-2022-49541[41]

The idea is to ensure that the memory is freed only once, and then the green section shows the patched code. When "*sb" is NULL, it indicates that the superblock was never initialized, and therefore, its destructor function is not called. To free the "cifs_sb" correctly, the "if condition" is modified to "!sb || IS_ERR(sb)", which ensures that the "free()" operator is executed only when "sb" is either uninitialized or not successfully initialized. This modification ensures that the "free()" operation is executed only if the destructor function is not invoked, thereby successfully resolving double free vulnerability.

The NVD provides CVSS metric settings as shown in Table 6.24 which we adopt in our scoring model. For the RL, figure 6.11 shows a committed patch, therefore, RL is set to "Official Fix". Regarding the metric P, only one of the five brokers (Mosquitto) is affected by this issue, and therefore, it is set to "Very Little". Finally, for metric F, we identified five relevant records in the NVD between March 20 and April 20 (in Appendix 6, Figure F.1), so F is also set to "Very Little". The complete configuration is shown in Table 6.25. Substituting the values into Equation (5.5), we calculate the final score of the "Double Free" vulnerability as 4.63113, which is

Table 6.24: Metrics setting of CVE-2022-49541 in CVSS [35],CVSS Version 3x

Metrics	Setting
AV	Local
AC	Low
PR	Low
UI	None
S	UnExceeded
C	High
I	High
A	High

Table 6.25: Metric setting in Improved Scoring System

Metrics	Setting
AV	Local
PR	Low
UI	None
S	UnExceeded
C	High
I	High
A	High
RL	OfficialFix
F	VeryLittle
P	VeryLittle

classified as “Medium” severity level.

6.4.3.4 Hard Coded Credentials

Hard coded credentials refers to usernames, passwords, API keys, tokens, or other secrets that are directly written in the source code instead of being securely stored and retrieved from configuration files or environment variables. As shown in Figure 6.12, on line 98, variables “username” and “password” are used to store the client’s authentication information, which is later verified by the broker when a client attempts to connect to the broker. Since Mosquitto is open-source and publicly accessible, attackers can easily extract these credentials from the code and impersonate a legitimate user to connect to the MQTT Broker. Thereby, gaining unauthorized access to the MQTT broker.

```
96    rc = 1
97    keepalive = 10
98    connect_packet_admin = mosq_test.gen_connect("ctrl-test", keepalive=keepalive, username="admin", password="admin")
99    connack_packet_admin = mosq_test.gen_connack(rc=0)
100
```

Figure 6.12: Mosquitto source code that contains hard coded credentials[41].

A similar vulnerability (CVE-2024-41794) is identified in the NVD, in which attackers can gain root privileges through remote access to the device’s operating system. Notably, CVE-2024-41794 carries the highest possible CVSS score of 10, indicating a critical severity level [36]. Table 6.26 shows the CVSS metric configuration for CVE-2024-41794, which we adopt to calculate the vulnerability score with our scoring model. Table 6.27 shows the metrics configuration in our scoring model.

Table 6.26: Metrics setting of CVE-2024-41794 in CVSS [36],CVSS Version 3x

Metrics	Setting
AV	Network
AC	Low
PR	None
UI	None
S	Exceeded
C	High
I	High
A	High

Table 6.27: Metric setting in Improved Scoring System

Metrics	Setting
AV	Online
PR	None
UI	None
S	Exceeded
C	High
I	High
A	High
RL	OfficialFix
F	VeryLittle
P	Low

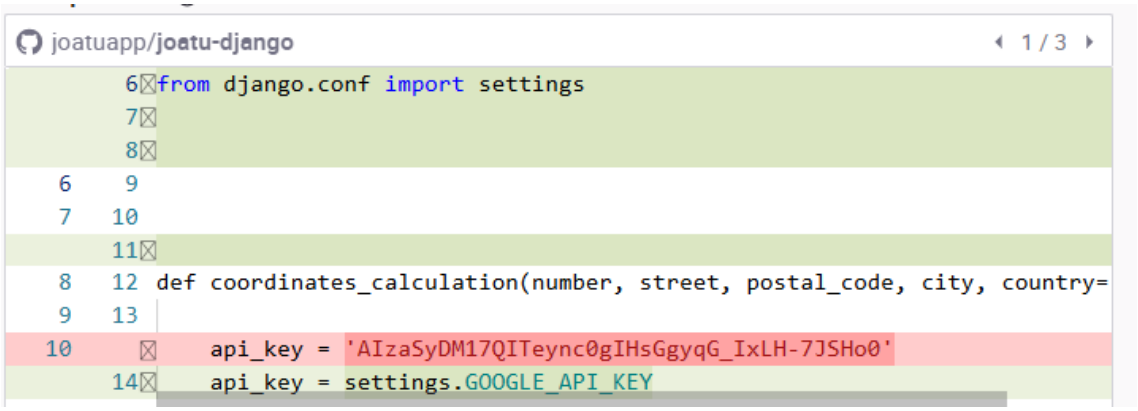


Figure 6.13: Hard coded credentials and fix suggested by Snyk.

For the RL metric, a common method to address the "hard coded credentials" is to store keys and passwords outside the source code. For example, Snyk provides a recommended fix, shown in Figure 6.13, which removes sensitive hard-coded information and moves it to Django's settings configuration file "settings.py". This file is not publicly accessible and is deployed on the server. The server retrieves configurations from Django's environment variables, rather than from static code. Therefore, RL is set to "Official Fix".

The metric P is set to "Low", since two of five brokers (Mosquitto and Mosca) have this issue. Metric F is set to "Very little", since there were three records related to hard-coded credentials between March 20 and April 20 (see Appendix G.1). Substituting the above values into Equation (5.5), the final calculated score is 10.9725, which falls into the "Critical" severity level.

6.4.3.5 Evaluation of improved scoring model

So far, we have calculated four types of vulnerabilities in MQTT brokers: Path Traversal (CWE-23), Buffer Overflow (CWE-122), Double Free (CWE-415), and Hard coded Credentials (CWE-798). Table 6.28 presents the corresponding vulnerability scores as calculated by both the traditional CVSS and our improved scoring model.

Table 6.28: Scores in CVSS and Improved Scoring System

CWE-ID	CVE-ID	**F x P**	Score in CVSS	**Score in improved scoring system**
CWE-23 Path Traversal	CVE-2022-45918	1.2 x 1.15	6.5 (Medium)	8.94596 (High)
CWE-122 BufferFlow	CVE-2018-1000300	1.2 x 1.1	9.8 (Critical)	12.1627 (Critical)
CWE-415 Double Free	CVE-2022-49541	1.05 x 1.05	7.8 (High)	4.63113 (Medium)
CWE-798 Hardcode Credentials	CVE-2024-4179	1.05 x 1.05	10.0 (Critical)	10.9725 (Critical)

Notably, CVE-2022-45918 (CWE-23) shows an increase in score from 6.5 in CVSS to 8.94596 in our proposed scoring model, with its severity level rising from "Medium" to "High". CVE-2018-1000300 (CWE-122), classified as a critical vulnerability in CVSS with a vulnerability score of 9.8, also remains in the "critical" level in our model, with an increase of approximately 20%, from 9.8 to 12.1627. However, in the case of CVE-2022-49541, it has a drop of approximately 40%, from 7.8 to 4.63113. Finally, for CVE-2024-4179, it increases moderately from 10 to 10.9725 and remains at the "critical" severity level.

The newly incorporated metrics F and P, their influence on the final score, and their settings for each CVE-ID are shown in Table 6.28. The score increase observed in the first, third, and fourth rows reflects the positive contribution of F and P, aligning with the goal of our improved model. However, for the third row (CVE-2022-49541), although F and P introduce a slight upward adjustment, but the final score still decreases. This is because the PR metric is set to "Low", meaning that attackers cannot directly exploit the vulnerability without any user interaction (for example, no need to enter a password). This significantly lowers the final score, outweighing the marginal increase introduced by F and P.

Overall, the improved scoring model emphasizes the real-world impact of F and P, increasing the vulnerability score for widely distributed and frequently occurring vulnerabilities. At the same time, the influence of PR metric is also strengthened in our model, causing a notable score reduction for some vulnerabilities, due to the

absence of privilege required.

In general, the traditional CVSS framework emphasizes exploitability and impact, which can lead to an overestimation of the severity of vulnerabilities that are unlikely to be exploited in practice or are easily remediated. In contrast, the proposed scoring model introduces additional evaluation metrics by incorporating real-world factors such as Fixability and Repeatability. As a result, it provides a more realistic assessment of the severity of a vulnerability.

7

Discussion

This chapter discusses the key findings of the study and the improved vulnerability scoring model including its limitations. Furthermore, this chapter outlines recommendations for future work aimed at improving security of MQTT broker platforms.

7.1 Key Findings

Our research underscores the importance of proactive security assessments in IoT deployments. Even protocols such as MQTT, designed for efficiency and simplicity, can become vectors for critical vulnerabilities if the underlying software implementations are not adequately vetted.
The findings of this study highlight the ongoing security challenges in MQTT and its broker platforms. Although these broker platforms are designed to support efficient and lightweight communication for IoT systems, our analysis revealed security issues that stem not only from broker architecture but also from the use of third-party libraries and plugins. The results affirm concerns raised in the previous literature regarding the security risks broker platforms have.

Furthermore, the use of the vulnerability scanning tool "Snyk" proved to be beneficial in identifying known vulnerabilities. Snyk detected multiple outdated and insecure dependencies that might not be evident through code inspection alone. However, the analysis also revealed limitations of such tools. For example, it was highly dependent on up-to-date vulnerability databases. This suggests that relying only on static analysis is inadequate and should be complemented by manual reviews and dynamic analysis for more comprehensive coverage.

Another observation during the analysis was that the vulnerability distribution was uneven between the broker platforms, as shown in Figure 6.1. While some broker platforms had a relatively higher number of reported issues, others had a lower number. This unevenness could be influenced by factors such as code quality or the popularity of the project, an area area that warrants further investigation in future research.

7.2 Improved Vulnerability Scoring Model

Our research underscores the importance of secure architectural decisions and proposes an improved vulnerability prioritization model to improve the resilience of the IoT infrastructure. One of the key contributions of this thesis is the development of an improved vulnerability scoring model. While the traditional CVSS is valuable for standardizing severity assessment, our analysis shows that it does not take into account contextual factors, particularly in the IoT domain.

Unlike the traditional CVSS, our proposed model incorporates additional factors such as Popularity (how many broker platforms are affected by the same vulnerability within a given time frame) and Frequency (the number of times a vulnerability appears in the NVD within a given time frame). These metrics make our vulnerability scoring model context aware and tailored to the characteristics of IoT systems and MQTT broker platforms. However, incorporating new metrics introduces certain challenges. Specifically, the traditional CVSS scoring range of 0 to 10, is no longer directly applicable and needs to be adjusted. In our model, the "critical" severity level, which is defined as scores between 9 and 10 in traditional CVSS, may now include scores that exceed 10. Although this limitation has minimal impact on the model's effectiveness in vulnerability assessment.

In addition, we revised several CVSS metric configurations to improve clarity and reduce potential errors. This includes merging options in the "Attack Vector" metric, relocating "Scope" from Exploitability to Severity, and simplifying the configuration by removing "Attack Complexity". Through these enhancements, our proposed model offers a more realistic prioritization approach to vulnerabilities specific to MQTT broker platforms. The proposed scoring model addresses the limitations of traditional CVSS methods by incorporating real-world relevance, thereby providing more accurate and targeted support for the assessment of security risk.

7.3 Future Work

In the future, additional validations are required for the new scoring model to ensure that the Frequency and Popularity metrics remain effective across a broader range of vulnerabilities.
Future research can also explore the application of this model to larger and more diverse vulnerability datasets, covering more broker platforms, and extending to different time frames. This helps to evaluate the versatility and scalability of the scoring method.
Moreover, with the rise of AI, the integration of ML techniques into the scoring model presents a promising direction for future work. For example, using existing CVSS configurations provided in the NVD as a training set, a ML model can be trained to predict the metric values, helping developers reduce human error.
Finally, the continued development and empirical evaluation of such scoring frameworks is essential to advance vulnerability assessment methods that are more in line with the dynamics and complexity of modern IoT environments.

8

Conclusion

This thesis has explored the security landscape of MQTT broker platforms by analyzing their vulnerabilities and evaluating their severity, evolution, and distribution across widely used open-source implementations such as EMQX, Mosca, Vernemq, Mosquitto, and Hivemq.
The findings reveal that while MQTT is widely adopted in IoT systems for its lightweight publish-subscribe communication model, it remains vulnerable to a wide range of security threats. This study identified key trends in the evolution of MQTT-related vulnerabilities, including frequent occurrences of Denial of Service (DoS) attacks, inadequate authentication mechanisms, and security risks stemming from third-party dependencies.

A major contribution of this study is the development of an improved vulnerability prioritization model. Unlike the traditional CVSS, the proposed model incorporates two novel metrics: **Popularity** refers to how many broker platforms are affected by the same vulnerability within a given time frame. **Frequency** refers to the number of times a vulnerability appears in the National Vulnerability Database (NVD) within a given time frame. These enhancements reflect real-world relevance and exposure, providing a practical approach to vulnerability prioritization.

Furthermore, the model modifies some existing CVSS metrics to simplify the metric configuration process, reduce scoring errors, and better reflect the threat landscape specific to the MQTT and IoT ecosystems. In addition, by integrating empirical vulnerability distribution data with a tailored scoring approach, this framework allows security analysts and developers to allocate resources more efficiently and address the most impactful issues first.

In conclusion, this thesis not only maps the current vulnerability landscape of MQTT brokers but also introduces a refined risk assessment model that is better aligned with the unique characteristics of IoT security requirements. Future research may expand this model across larger datasets to evaluate its effectiveness and accuracy in diverse environments.

Bibliography

[1] *Security risks in MQTT-based Industrial IoT Applications*, 2022.

[2] N. S. Alotaibi, H. I. Sayed Ahmed, S. O. M. Kamel, and G. F. ElKabbany. Secure enhancement for mqtt protocol using distributed machine learning framework. *Sensors*, 24(5):1638, 2024.

[3] Syaiful Andy, Budi Rahardjo, and Bagus Hanindhito. Attack scenarios and security analysis of mqtt communication protocol in iot system. In *2017 4th International conference on electrical engineering, computer science and informatics (EECSI)*, pages 1–6. IEEE, 2017.

[4] Niklas Schilling (Office Munich) | SEC Consult Vulnerability Lab Anna Hartig (Office Bochum), Constantin Schwarz (Office Bochum). Multiple critical vulnerabilities in ilias elearning platform. *ILIAS eLearning platform*, 2022.

[5] N. Bakar, F. Khan, and S. Latif. Mqtt security challenges and vulnerabilities: A review. In *Proceedings of the 8th International Conference on IoT Security*, pages 120–130. IEEE, 2019.

[6] Vladimir Bashun and Aleksei Shadrunov. An evaluation of container security vulnerability detection tools. In *Proceedings of the 2024 International Conference on Cybersecurity and Privacy Protection*, 2024.

[7] DHS. CWE Official website. `https://cwe.mitre.org/index.html`, 2018. Accessed: 2025-06-01.

[8] Eclipse Mosquitto. Authentication Methods in Mosquitto. `https://mosquitto.org/documentation/authentication-methods/`, 2023. Accessed: 2025-05-19.

[9] EMQX Documentation. EMQX Enterprise v4 Change Log. `https://docs.emqx.com/en/emqx/latest/changes/changes-ee-v4.html`, 2025. Accessed: 2025-03-09.

[10] Emqx Github. Emqx-Public archive. `https://github.com/emqx/emqx`, 2025. Accessed: 2025-05-20.

[11] EMQX Team. Authorization in MQTT: Using ACLs to Control Access to MQTT Messaging, 2023. Accessed: 2025-03-02.

[12] FIRST. Specification Document Section 3.1: Exploit Maturity (E). https://www.first.org/cvss/v4-0/specification-document, 2023. Accessed: 2025-05-19.

[13] First.org. Common Vulnerability Scoring System v3.1: Specification Document. https://www.first.org/cvss/v3-1/specification-document, 2023. Section 1, Accessed: 2025-05-19.

[14] First.org. Common Vulnerability Scoring System v4.0: Version History. https://www.first.org/cvss/v4-0/faq, 2023. Section 4, Accessed: 2025-05-19.

[15] FIRST.org. Specification Document: CVSS v4.0 Metric Groups. https://www.first.org/cvss/v4-0/specification-document, 2023. Accessed: 2025-05-19.

[16] First.org. Common Vulnerability Scoring System v4.0: Specification Document. https://www.first.org/cvss/v4-0/specification-document, 2024. Section 1, Accessed: 2025-05-19.

[17] Michael Flanders. A simple and intuitive algorithm for preventing directory traversal attacks. *CoRR*, abs/1908.04502, 2019.

[18] NCC Group. Technical advisory: Mosquitto broker dos through a memory leak vulnerability, 2023. Accessed: March 9, 2025.

[19] Hivemq-community-edition Github . Hivemq-community-edition public . https://github.com/hivemq/hivemq-community-edition, 2025. Accessed: 2025-05-20.

[20] HiveMQ Team. introduction to mqtt quality of service (qos) levels, 2015. Accessed: 2025-05-17.

[21] HiveMQ Team. Mqtt essentials part 6: Mqtt quality of service (qos) levels, 2015. Accessed: 2025-05-17.

[22] HiveMQ Team. MQTT Security Fundamentals: Authorization. https://www.hivemq.com/blog/mqtt-security-fundamentals-authorization/, 2024. Accessed: 2025-03-09.

[23] IoT For All. 5 Worst IoT Hacking Vulnerabilities, 2021.

[24] Shaofeng Kai, Jinghua Zheng, Fan Shi, and Zhifan Lu. A cvss-based vulnerability assessment method for reducing scoring error. In *2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT)*, pages 25–32, 2021.

[25] Bingchang Liu, Guozhu Meng, Wei Zou, Qi Gong, Feng Li, Min Lin, Dandan Sun, Wei Huo, and Chao Zhang. A large-scale empirical study on vulnerability distribution within projects and the lessons learned. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*, pages 1547–1559, 2020.

[26] S. Moschoyiannis M. A. Ferrag, L. Maglaras and H. Janicke. A Brief Review on MQTT's Security Issues within the Internet of Things (IoT). *ResearchGate*, 2019.

[27] A. Mahmoud, J. Smith, and L. Zhang. Security analysis of mqtt broker implementations: A comparative study. *International Journal of Cybersecurity and IoT*, 12(3):45–60, 2022.

[28] Mosca Github. Mosca-Public archive. `https://github.com/moscajs/mosca`, 2020. Accessed: 2025-05-20.

[29] Mosquitto Github . mosquitto public . `https://github.com/eclipse-mosquitto/mosquitto`, 2025. Accessed: 2025-05-20.

[30] MQTT Organization. Mqtt - the standard for iot messaging, 2024. Accessed: 2024-12-01.

[31] Nitin Naik. Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. In *2017 IEEE International Systems Engineering Symposium (ISSE)*, pages 1–7, 2017.

[32] NVD NIST. CVE-2018-1000300 Detail. `https://nvd.nist.gov/vuln/detail/CVE-2018-1000300`, 2018. Accessed: 2025-05-19.

[33] NVD NIST. CVE-2022-45918 Detail. `https://nvd.nist.gov/vuln/detail/CVE-2022-45918`, 2018. Accessed: 2025-05-19.

[34] NVD NIST. CVE-2019-10743 Detail. `https://nvd.nist.gov/vuln/detail/CVE-2019-10743`, 2019. Accessed: 2025-05-19.

[35] NVD NIST. CVE-2022-49541 Detail. `https://nvd.nist.gov/vuln/detail/CVE-2022-49541`, 2022. Accessed: 2025-05-19.

[36] NVD NIST. CVE-2024-41794 Detail. `https://nvd.nist.gov/vuln/detail/CVE-2024-41794`, 2024. Accessed: 2025-05-19.

[37] NVD NIST. General Information. `https://nvd.nist.gov/general`, 2025. Accessed: 2025-05-19.

[38] Edoardo Di Paolo, Enrico Bassetti, and Angelo Spognardi. Security assessment of common open source mqtt brokers and clients, 2023.

[39] Project curl Security Advisory. FTP shutdown response buffer overflow. `https:`

//curl.se/docs/CVE-2018-1000300.html, 2018. Accessed: 2025-05-19.

[40] Kaspersky Research. 33 vulnerabilities in mqtt implementations affecting iot devices, 2023. Accessed: 2024-12-11.

[41] Ronnie Sahlberg. cifs: fix potential double free during failed mount. https://git.kernel.org/stable/c/8378a51e3f8140f60901fb27208cc7a6e47047b5, 2022. Accessed: 2025-05-19.

[42] Muhammad Shahzad, Muhammad Zubair Shafiq, and Alex X. Liu. A large scale exploratory analysis of software vulnerability life cycles. In *2012 34th International Conference on Software Engineering (ICSE)*, pages 771–781, 2012.

[43] Meena Singh, M.A. Rajan, V.L. Shivraj, and P. Balamuralidhar. Secure mqtt for internet of things (iot). In *2015 Fifth International Conference on Communication Systems and Network Technologies*, pages 746–751, 2015.

[44] Snyk. Directory traversal: Unintended disclosure of sensitive files. https://learn.snyk.io/lesson/directory-traversal/?ecosystem=javascript, 2024. Accessed: 2025-05-19.

[45] Jonathan Spring, Eric Hatleback, A Manion, and D Shic. Towards improving cvss. *SEI, CMU, Tech. Rep*, 2018.

[46] Klaas-Jan Stol and Brian Fitzgerald. The abc of software engineering research. *ACM Transactions on Software Engineering and Methodology*, 27(3):11:1–11:51, September 2018.

[47] Ivan Vaccari, Maurizio Aiello, and Enrico Cambiaso. SlowITe, a Novel Denial of Service Attack Affecting MQTT. *Sensors*, 20(10):2932, 2020.

[48] Vernemq Github . vernemq public . https://github.com/vernemq/vernemq, 2025. Accessed: 2025-05-20.

[49] Ruyi Wang, Ling Gao, Qian Sun, and Deheng Sun. An improved cvss-based vulnerability scoring mechanism. In *2011 Third International Conference on Multimedia Information Networking and Security*, pages 352–355, 2011.

[50] SISA Threat Watch. Iot sparks new ddos alert: The emergence of ddos 2.0, 2023. Accessed: 2024-12-11.

[51] Julia Wunder, Andreas Kurtz, Christian Eichenmüller, Freya Gassmann, and Zinaida Benenson. Shedding light on cvss scoring inconsistencies: A user-centric study on evaluating widespread security vulnerabilities. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 1102–1121, 2024.

[52] Bin Yuan, Zhanxiang Song, Yan Jia, Zhenyu Lu, Deqing Zou, Hai Jin, and Luyi Xing. Mqttactic: Security analysis and verification for logic flaws in mqtt implementations. In *2024 IEEE Symposium on Security and Privacy (SP)*,

pages 2385–2403. IEEE, 2024.

A

Appendix 1 - AI Usage

All ideas, experimental design, algorithm design and analysis, and interpretation of results were only conducted by us. The AI tools were not involved in generating any novel research contributions or analyzing original data; they were used exclusively as writing assistants.

The AI tools used include ChatGPT (GPT-4o and GPT-4o mini) and Grammarly. These tools assisted with grammar checking, formatting (e.g., citations and tables), and improving readability (e.g., coherence and transitions).

B

Appendix 2 Report of Snyk

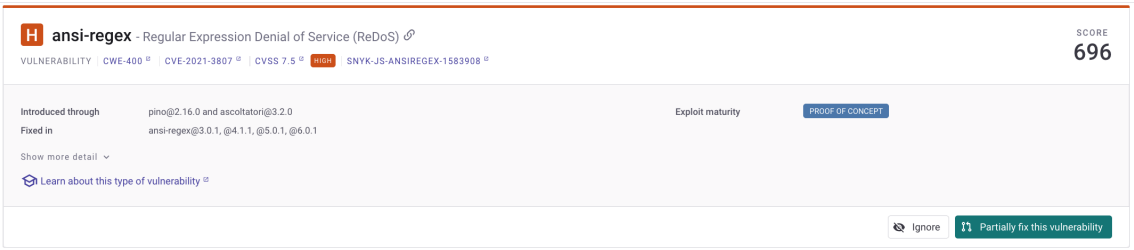


Figure B.1: Result of running Static analysis on broker platform

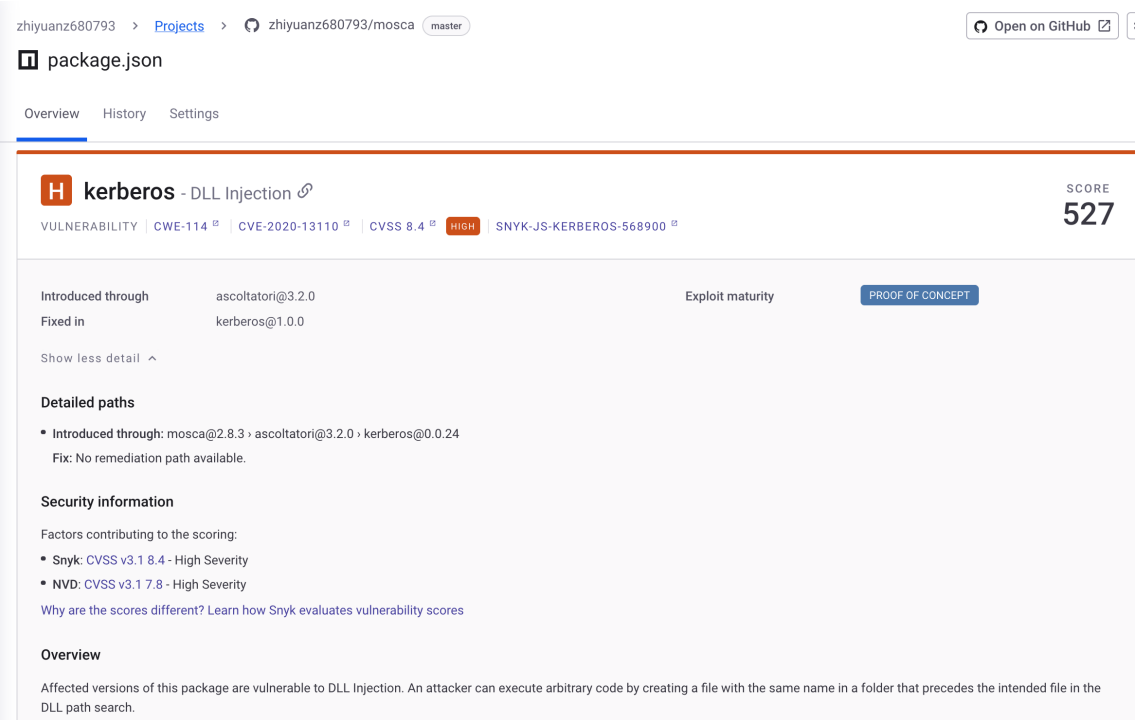


Figure B.2: Result of running static analysis on Mosca broker platform.

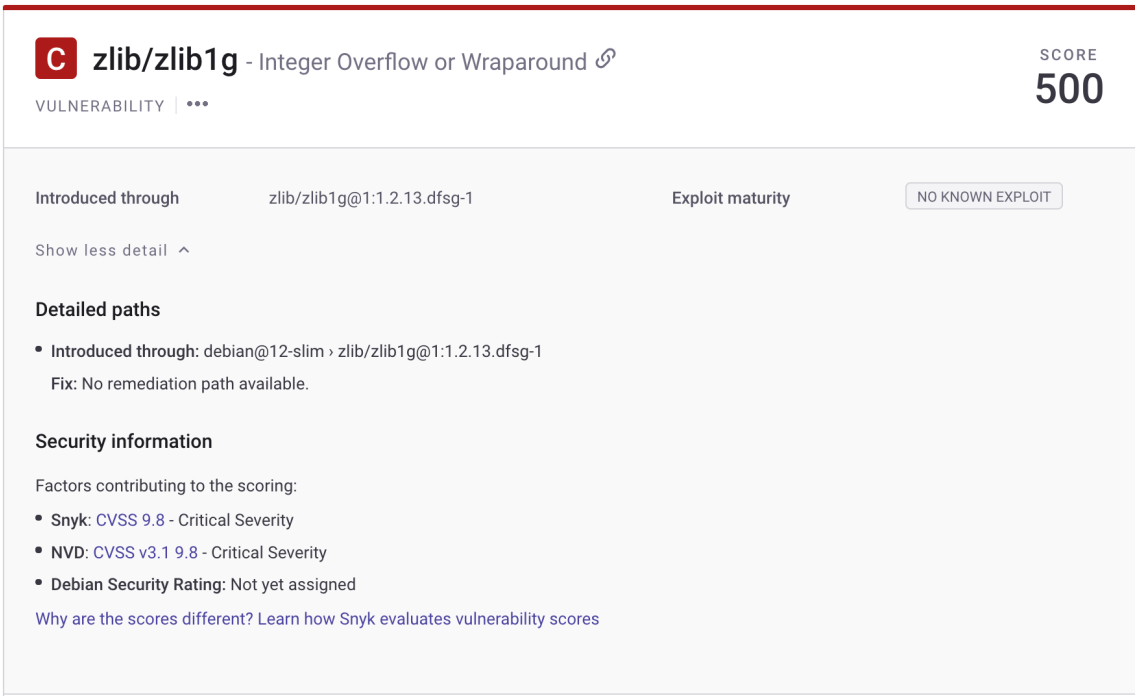


Figure B.3: Result of running static analysis on EMQX broker platform.

C

Appendix 3 Source Code of MQTT Brokers and Version Information

Table C.1: Source Code of MQTT Brokers and Version Information

Broker Name	**Is it the Latest Version**	**Version We Survey**	**Link**
Vernemq	Yes	2.0.1	[48]
Hivemq-community-edition	Yes	2025.2	[19]
Mosquitto	No	2.0.20	[29]
Mosca	Yes	2.8.3	[28]
Emqx	No	5.8.5	[10]

D

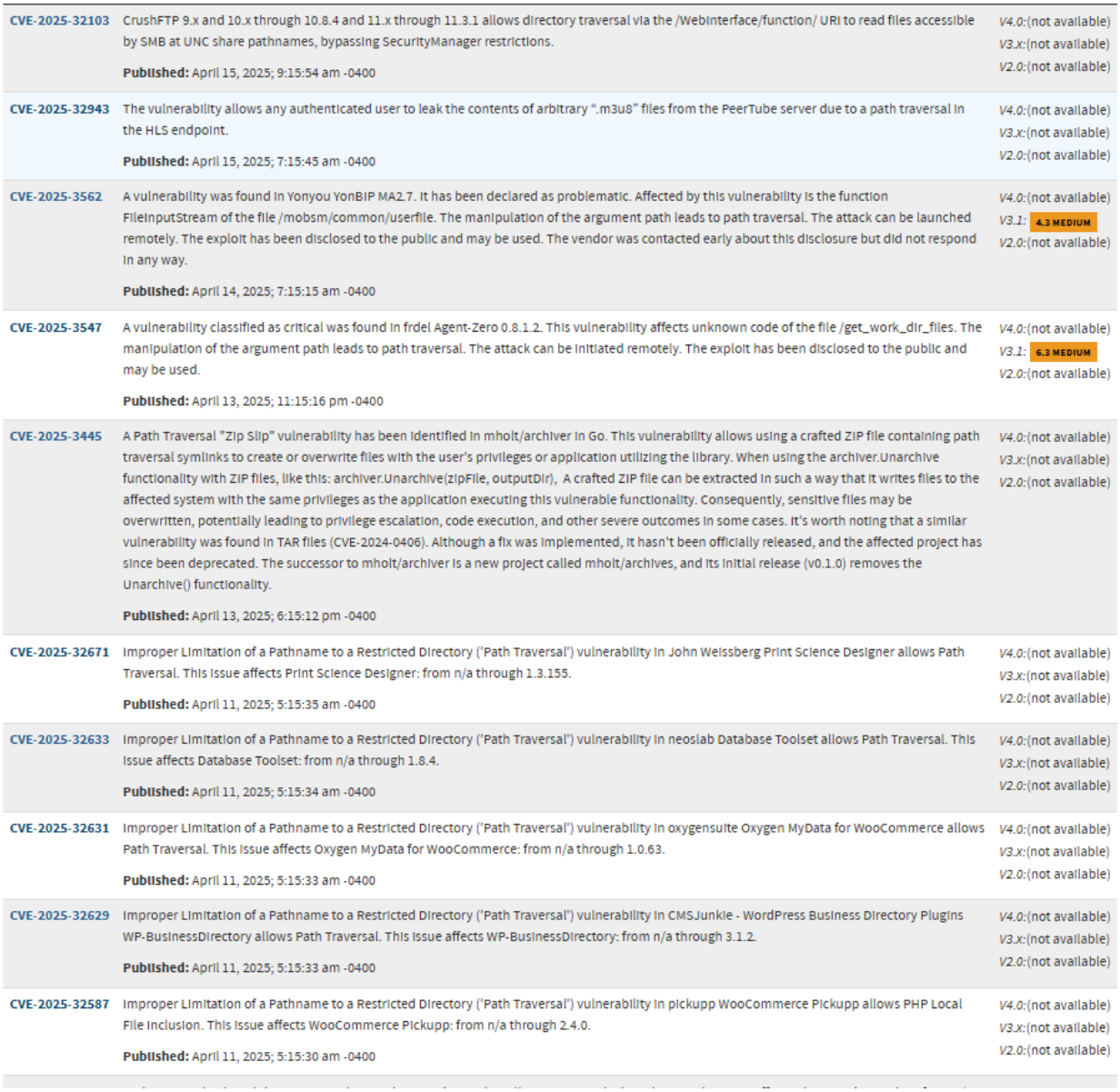
Appendix 4 Appearance of Path Traversal in NVD

(20th March 2025 to 20th April 2025)

Totally 108 path traversal vulnerabilities.

	Published: April 21, 2025; 11:15:21 pm -0400	
CVE-2025-39470	Path Traversal: '.../...//' vulnerability in ThimPress Ivy School allows PHP Local File Inclusion.This issue affects Ivy School: from n/a through 1.6.0. **Published:** April 18, 2025; 1:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-39568	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in Arture B.V. StoreContrl Woocommerce allows Path Traversal. This issue affects StoreContrl Woocommerce: from n/a through 4.1.3. **Published:** April 17, 2025; 12:15:58 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27299	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in WP Asia MyTicket Events allows Path Traversal. This issue affects MyTicket Events: from n/a through 1.2.4. **Published:** April 17, 2025; 12:15:36 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27283	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in rockgod100 Theme File Duplicator allows Path Traversal. This issue affects Theme File Duplicator: from n/a through 1.3. **Published:** April 17, 2025; 12:15:34 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-39598	Path Traversal vulnerability in Quý Lê 91 Administrator Z allows Path Traversal. This issue affects Administrator Z: from n/a through 2025.03.28. **Published:** April 16, 2025; 9:15:52 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-39544	Cross-Site Request Forgery (CSRF) vulnerability in Bill Minozzi WP Tools allows Path Traversal. This issue affects WP Tools: from n/a through 5.18. **Published:** April 16, 2025; 9:15:47 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3686	A vulnerability classified as problematic was found in misstt123 oasys 1.0. Affected by this vulnerability is the function image of the file /show. The manipulation leads to path traversal. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. This product does not use versioning. This is why information about affected and unaffected releases are unavailable. **Published:** April 16, 2025; 8:15:17 am -0400	*V4.0:*(not available) *V3.1:* 4.3 MEDIUM *V2.0:*(not available)
CVE-2025-30966	Path Traversal vulnerability in NotFound WPJobBoard allows Path Traversal. This issue affects WPJobBoard: from n/a through n/a. **Published:** April 15, 2025; 6:15:26 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27791	Collabora Online is a collaborative online office suite based on LibreOffice technology. In versions prior to 24.04.12.4, 23.05.19, and 22.05.25, there is a path traversal flaw in handling the CheckFileInfo BaseFileName field returned from WOPI servers. This allows for a file to be written anywhere the uid running Collabora Online can write, if such a response was supplied by a malicious WOPI server. By combining this flaw with a Time of Check, Time of Use DNS lookup issue with a WOPI server address under attacker control, it is possible to present such a response to be processed by a Collabora Online instance. This issue has been patched in versions 24.04.13.1, 23.05.19, and 22.05.25. **Published:** April 15, 2025; 3:16:07 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.1: First 9 of Path Traversal

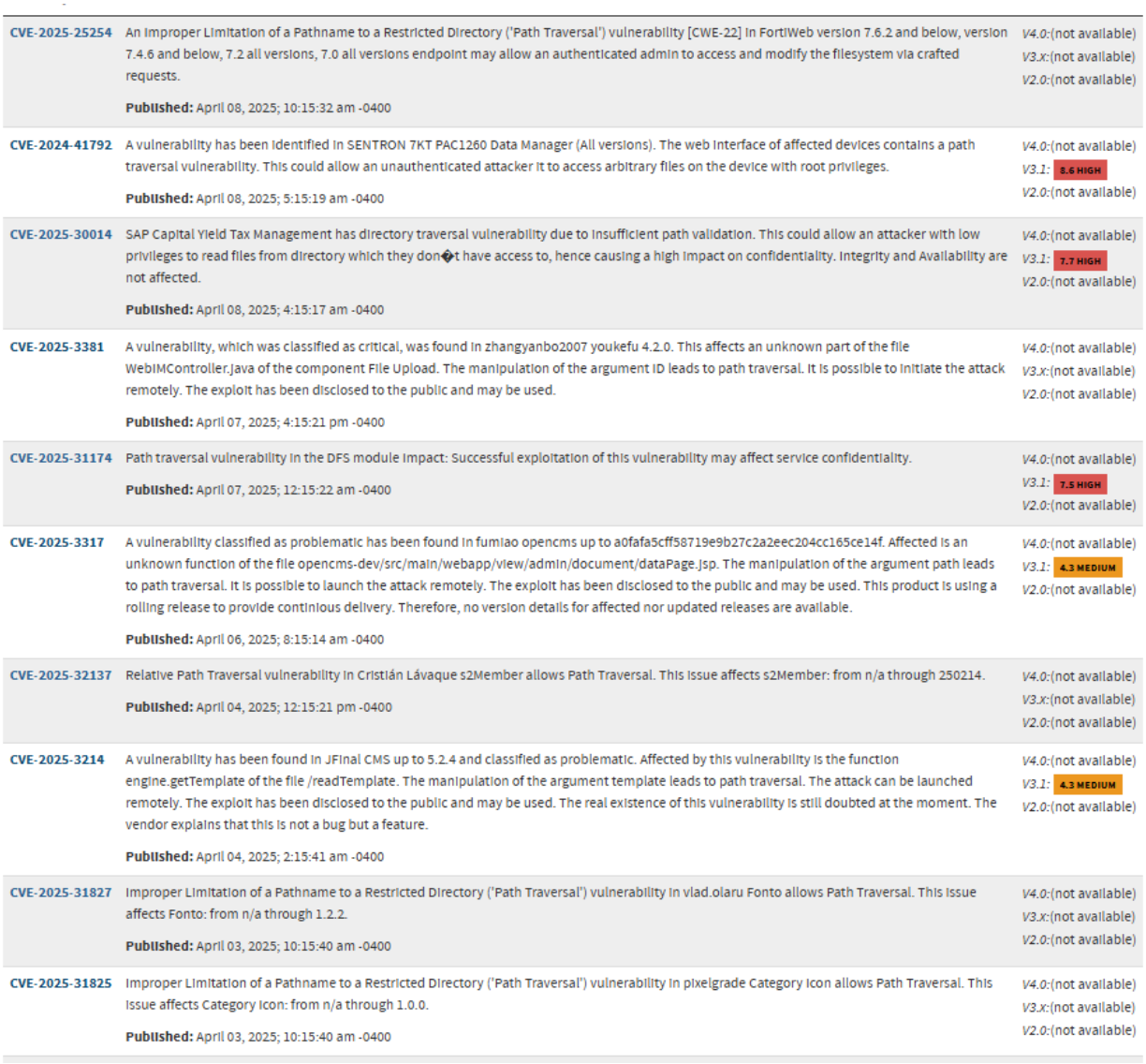


CVE-2025-32103	CrushFTP 9.x and 10.x through 10.8.4 and 11.x through 11.3.1 allows directory traversal via the /WebInterface/function/ URI to read files accessible by SMB at UNC share pathnames, bypassing SecurityManager restrictions. **Published:** April 15, 2025; 9:15:54 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32943	The vulnerability allows any authenticated user to leak the contents of arbitrary ".m3u8" files from the PeerTube server due to a path traversal in the HLS endpoint. **Published:** April 15, 2025; 7:15:45 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3562	A vulnerability was found in Yonyou YonBIP MA2.7. It has been declared as problematic. Affected by this vulnerability is the function FileInputStream of the file /mobsm/common/userfile. The manipulation of the argument path leads to path traversal. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** April 14, 2025; 7:15:15 am -0400	*V4.0:*(not available) *V3.1:* 4.3 MEDIUM *V2.0:*(not available)
CVE-2025-3547	A vulnerability classified as critical was found in frdel Agent-Zero 0.8.1.2. This vulnerability affects unknown code of the file /get_work_dir_files. The manipulation of the argument path leads to path traversal. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 13, 2025; 11:15:16 pm -0400	*V4.0:*(not available) *V3.1:* 6.3 MEDIUM *V2.0:*(not available)
CVE-2025-3445	A Path Traversal "Zip Slip" vulnerability has been identified in mholt/archiver in Go. This vulnerability allows using a crafted ZIP file containing path traversal symlinks to create or overwrite files with the user's privileges or application utilizing the library. When using the archiver.Unarchive functionality with ZIP files, like this: archiver.Unarchive(zipFile, outputDir), A crafted ZIP file can be extracted in such a way that it writes files to the affected system with the same privileges as the application executing this vulnerable functionality. Consequently, sensitive files may be overwritten, potentially leading to privilege escalation, code execution, and other severe outcomes in some cases. It's worth noting that a similar vulnerability was found in TAR files (CVE-2024-0406). Although a fix was implemented, it hasn't been officially released, and the affected project has since been deprecated. The successor to mholt/archiver is a new project called mholt/archives, and its initial release (v0.1.0) removes the Unarchive() functionality. **Published:** April 13, 2025; 6:15:12 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32671	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in John Weissberg Print Science Designer allows Path Traversal. This issue affects Print Science Designer: from n/a through 1.3.155. **Published:** April 11, 2025; 5:15:35 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32633	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in neoslab Database Toolset allows Path Traversal. This issue affects Database Toolset: from n/a through 1.8.4. **Published:** April 11, 2025; 5:15:34 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32631	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in oxygensuite Oxygen MyData for WooCommerce allows Path Traversal. This issue affects Oxygen MyData for WooCommerce: from n/a through 1.0.63. **Published:** April 11, 2025; 5:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32629	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in CMSJunkie - WordPress Business Directory Plugins WP-BusinessDirectory allows Path Traversal. This issue affects WP-BusinessDirectory: from n/a through 3.1.2. **Published:** April 11, 2025; 5:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32587	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in pickupp WooCommerce Pickupp allows PHP Local File Inclusion. This issue affects WooCommerce Pickupp: from n/a through 2.4.0. **Published:** April 11, 2025; 5:15:30 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.2: Second 10 of Path Traversal

CVE-2025-32585	Path Traversal vulnerability in Trusty Plugins Shop Products Filter allows PHP Local File Inclusion. This issue affects Shop Products Filter: from n/a through 1.2. **Published:** April 11, 2025; 5:15:29 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32509	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in WPMinds Simple WP Events allows Path Traversal. This issue affects Simple WP Events: from n/a through 1.8.17. **Published:** April 11, 2025; 5:15:23 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32807	A path traversal vulnerability in FusionDirectory before 1.5 allows remote attackers to read arbitrary files on the host that end with .png (and .svg or .xpm for some configurations) via the icon parameter of a GET request to geticon.php. **Published:** April 10, 2025; 8:15:27 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-31411	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in Aribhour Linet ERP-Woocommerce Integration allows Path Traversal.This issue affects Linet ERP-Woocommerce Integration: from n/a through 3.5.12. **Published:** April 10, 2025; 7:15:45 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32209	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in totalprocessing Total processing card payments for WooCommerce allows Path Traversal. This issue affects Total processing card payments for WooCommerce: from n/a through 7.1.5. **Published:** April 10, 2025; 4:15:17 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32205	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in plotnetdotcom Plotnet Forms. This issue affects Plotnet Forms: from n/a through 1.0.30. **Published:** April 10, 2025; 4:15:17 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30582	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in aytechnet DyaPress ERP/CRM allows PHP Local File Inclusion. This issue affects DyaPress ERP/CRM: from n/a through 18.0.2.0. **Published:** April 10, 2025; 4:15:14 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30290	ColdFusion versions 2023.12, 2021.18, 2025.0 and earlier are affected by an Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability that could lead to a security feature bypass. A high privileged attacker could exploit this vulnerability to bypass security protections and gain unauthorized write and delete access. Exploitation of this issue does not require user interaction and scope is changed. **Published:** April 08, 2025; 4:15:26 pm -0400	*V4.0:*(not available) *V3.1:* 8.7 HIGH *V2.0:*(not available)
CVE-2024-12556	Prototype Pollution in Kibana can lead to code injection via unrestricted file upload combined with path traversal. **Published:** April 08, 2025; 4:15:19 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-32017	Umbraco is a free and open source .NET content management system. Authenticated users to the Umbraco backoffice are able to craft management API request that exploit a path traversal vulnerability to upload files into a incorrect location. The issue affects Umbraco 14+ and is patched in 14.3.4 and 15.3.1. **Published:** April 08, 2025; 12:15:27 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.3: Third 10 of Path Traversal



CVE-2025-25254	An Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability [CWE-22] in FortiWeb version 7.6.2 and below, version 7.4.6 and below, 7.2 all versions, 7.0 all versions endpoint may allow an authenticated admin to access and modify the filesystem via crafted requests. **Published:** April 08, 2025; 10:15:32 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-41792	A vulnerability has been identified in SENTRON 7KT PAC1260 Data Manager (All versions). The web interface of affected devices contains a path traversal vulnerability. This could allow an unauthenticated attacker it to access arbitrary files on the device with root privileges. **Published:** April 08, 2025; 5:15:19 am -0400	*V4.0:*(not available) *V3.1:* 8.6 HIGH *V2.0:*(not available)
CVE-2025-30014	SAP Capital Yield Tax Management has directory traversal vulnerability due to insufficient path validation. This could allow an attacker with low privileges to read files from directory which they don't have access to, hence causing a high impact on confidentiality. Integrity and Availability are not affected. **Published:** April 08, 2025; 4:15:17 am -0400	*V4.0:*(not available) *V3.1:* 7.7 HIGH *V2.0:*(not available)
CVE-2025-3381	A vulnerability, which was classified as critical, was found in zhangyanbo2007 youkefu 4.2.0. This affects an unknown part of the file WebIMController.java of the component File Upload. The manipulation of the argument ID leads to path traversal. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 4:15:21 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-31174	Path traversal vulnerability in the DFS module Impact: Successful exploitation of this vulnerability may affect service confidentiality. **Published:** April 07, 2025; 12:15:22 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2025-3317	A vulnerability classified as problematic has been found in fumiao opencms up to a0fafa5cff58719e9b27c2a2eec204cc165ce14f. Affected is an unknown function of the file opencms-dev/src/main/webapp/view/admin/document/dataPage.jsp. The manipulation of the argument path leads to path traversal. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. This product is using a rolling release to provide continious delivery. Therefore, no version details for affected nor updated releases are available. **Published:** April 06, 2025; 8:15:14 am -0400	*V4.0:*(not available) *V3.1:* 4.3 MEDIUM *V2.0:*(not available)
CVE-2025-32137	Relative Path Traversal vulnerability in Cristián Lávaque s2Member allows Path Traversal. This issue affects s2Member: from n/a through 250214. **Published:** April 04, 2025; 12:15:21 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3214	A vulnerability has been found in JFinal CMS up to 5.2.4 and classified as problematic. Affected by this vulnerability is the function engine.getTemplate of the file /readTemplate. The manipulation of the argument template leads to path traversal. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. The real existence of this vulnerability is still doubted at the moment. The vendor explains that this is not a bug but a feature. **Published:** April 04, 2025; 2:15:41 am -0400	*V4.0:*(not available) *V3.1:* 4.3 MEDIUM *V2.0:*(not available)
CVE-2025-31827	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in vlad.olaru Fonto allows Path Traversal. This issue affects Fonto: from n/a through 1.2.2. **Published:** April 03, 2025; 10:15:40 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-31825	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in pixelgrade Category Icon allows Path Traversal. This issue affects Category Icon: from n/a through 1.0.0. **Published:** April 03, 2025; 10:15:40 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.4: Fourth 10 of Path Traversal

CVE-2025-31800	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in publitio Publitio allows Path Traversal. This issue affects Publitio: from n/a through 2.1.8. **Published:** April 03, 2025; 10:15:40 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-31554	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in docxpresso Docxpresso allows Absolute Path Traversal. This issue affects Docxpresso: from n/a through 2.6. **Published:** April 03, 2025; 10:15:36 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30596	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in NotFound include-file allows Path Traversal. This issue affects include-file: from n/a through 1. **Published:** April 03, 2025; 10:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2023-40714	A relative path traversal in Fortinet FortiSIEM versions 7.0.0, 6.7.0 through 6.7.2, 6.6.0 through 6.6.3, 6.5.1, 6.5.0 allows attacker to escalate privilege via uploading certain GUI elements **Published:** April 02, 2025; 4:15:13 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2023-46988	Path Traversal vulnerability in ONLYOFFICE Document Server before v8.0.1 allows a remote attacker to copy arbitrary files by manipulating the fileExt parameter in the /example/editor endpoint, leading to unauthorized access to sensitive files and potential Denial of Service (DoS). **Published:** April 01, 2025; 6:15:20 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30841	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in adamskaat Countdown & Clock allows Remote Code Inclusion. This issue affects Countdown & Clock: from n/a through 2.8.8. **Published:** April 01, 2025; 5:15:45 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-31131	YesWiki is a wiki system written in PHP. The squelette parameter is vulnerable to path traversal attacks, enabling read access to arbitrary files on the server. This vulnerability is fixed in 4.5.2. **Published:** April 01, 2025; 11:16:07 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2025-30910	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in CreativeMindsSolutions CM Download Manager allows Path Traversal. This issue affects CM Download Manager: from n/a through 2.9.6. **Published:** April 01, 2025; 2:15:54 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30882	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in JoomSky JS Help Desk allows Path Traversal. This issue affects JS Help Desk: from n/a through 2.9.1. **Published:** April 01, 2025; 2:15:54 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30878	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in JoomSky JS Help Desk allows Path Traversal. This issue affects JS Help Desk: from n/a through 2.9.2. **Published:** April 01, 2025; 2:15:53 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.5: Fifth 10 of Path Traversal

CVE-2025-30834	Path Traversal vulnerability in Bit Apps Bit Assist allows Path Traversal. This issue affects Bit Assist: from n/a through 1.5.4. **Published:** April 01, 2025; 2:15:52 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30793	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in Property Hive Houzez Property Feed allows Path Traversal. This issue affects Houzez Property Feed: from n/a through 2.5.4. **Published:** April 01, 2025; 2:15:51 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30594	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in NotFound Include URL allows Path Traversal. This issue affects Include URL: from n/a through 0.3.5. **Published:** April 01, 2025; 2:15:49 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2048	The Lana Downloads Manager WordPress plugin before 1.10.0 does not validate user input used in a path, which could allow users with an admin role to perform path traversal attacks and download arbitrary files on the server **Published:** April 01, 2025; 2:15:48 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3043	A vulnerability, which was classified as critical, has been found in GuoMinJim PersonManage 1.0. This issue affects the function preHandle of the file /login/. The manipulation of the argument Request leads to path traversal. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. This product takes the approach of rolling releases to provide continious delivery. Therefore, version details for affected and updated releases are not available. **Published:** March 31, 2025; 9:15:21 pm -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-30005	Xorcom CompletePBX is vulnerable to a path traversal via the Diagnostics reporting module, which will allow reading of arbitrary files and additionally delete any retrieved file in place of the expected report. This issue affects CompletePBX: all versions up to and prior to 5.2.35 **Published:** March 31, 2025; 1:15:41 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2292	Xorcom CompletePBX is vulnerable to an authenticated path traversal, allowing for arbitrary file reads via the Backup and Restore functionality.This issue affects CompletePBX: through 5.2.35. **Published:** March 31, 2025; 1:15:40 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3021	Path Traversal vulnerability in e-solutions e-management. This vulnerability could allow an attacker to access confidential files outside the expected scope via the 'file' parameter in the /downloadReport.php endpoint. **Published:** March 31, 2025; 7:15:39 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2961	A vulnerability classified as problematic was found in opensolon up to 3.1.0. This vulnerability affects the function render_mav of the file /aa of the component org.noear.solon.core.handle.RenderManager. The manipulation of the argument template with the input ../org/example/HelloApp.class leads to path traversal: '../filedir'. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** March 30, 2025; 6:15:15 pm -0400	*V4.0:*(not available) *V3.1:* 4.3 MEDIUM *V2.0:*(not available)
CVE-2025-2917	A vulnerability, which was classified as problematic, was found in ChestnutCMS up to 1.5.3. Affected is the function readFile of the file /dev-api/cms/file/read. The manipulation of the argument filePath leads to path traversal. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 2:15:17 pm -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)

Figure D.6: Sixth 10 of Path Traversal

CVE-2024-54362	Path Traversal vulnerability in NotFound GetShop ecommerce allows Path Traversal. This issue affects GetShop ecommerce: from n/a through 1.3. **Published:** March 28, 2025; 11:15:45 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2024-54291	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in NotFound PluginPass allows Manipulating Web Input to File System Calls. This issue affects PluginPass: from n/a through 0.9.10. **Published:** March 28, 2025; 11:15:45 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-27932	Improper limitation of a pathname to a restricted directory ('Path Traversal') issue exists in the file deletion process of the USB storage file-sharing function of HGW-BL1500HM Ver 002.002.003 and earlier. If this vulnerability is exploited, an attacker may delete a file on the device or cause a denial of service (DoS) condition. **Published:** March 28, 2025; 5:15:14 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-27726	Improper limitation of a pathname to a restricted directory ('Path Traversal') issue exists in the file download process of the USB storage file-sharing function of HGW-BL1500HM Ver 002.002.003 and earlier. If this vulnerability is exploited, the product's files may be obtained and/or altered by a crafted HTTP request to specific functions of the product from a device connected to the LAN side. **Published:** March 28, 2025; 5:15:14 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-27718	Improper limitation of a pathname to a restricted directory ('Path Traversal') issue exists in the file upload process of the USB storage file-sharing function of HGW-BL1500HM Ver 002.002.003 and earlier. If this vulnerability is exploited, the product's files may be obtained and/or altered or arbitrary code may be executed by a crafted HTTP request to specific functions of the product from a device connected to the LAN side. **Published:** March 28, 2025; 5:15:14 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-27716	Improper limitation of a pathname to a restricted directory ('Path Traversal') issue exists in the file/folder listing process of the USB storage file-sharing function of HGW-BL1500HM Ver 002.002.003 and earlier. If this vulnerability is exploited, the product's files may be obtained and/or altered by a crafted HTTP request to specific functions of the product from a device connected to the LAN side. **Published:** March 28, 2025; 5:15:14 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2024-12905	An Improper Link Resolution Before File Access ("Link Following") and Improper Limitation of a Pathname to a Restricted Directory ("Path Traversal"). This vulnerability occurs when extracting a maliciously crafted tar file, which can result in unauthorized file writes or overwrites outside the intended extraction directory. The issue is associated with index.js in the tar-fs package. This issue affects tar-fs: from 0.0.0 before 1.16.4, from 2.0.0 before 2.1.2, from 3.0.0 before 3.0.8. **Published:** March 27, 2025; 1:15:53 pm -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-30895	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in magepeopleteam WpEvently allows PHP Local File Inclusion. This issue affects WpEvently: from n/a through 4.2.9. **Published:** March 27, 2025; 7:15:50 am -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-25371	NASA cFS (Core Flight System) Aquila is vulnerable to path traversal in the OSAL module, allowing the override of any arbitrary file on the system. **Published:** March 25, 2025; 5:15:41 pm -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)
CVE-2025-30567	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') vulnerability in wp01ru WP01 allows Path Traversal. This issue affects WP01: from n/a through 2.6.2. **Published:** March 25, 2025; 3:15:46 pm -0400	*V4.0*:(not available) *V3.x*:(not available) *V2.0*:(not available)

Figure D.7: Seventh 10 of Path Traversal

CVE-2025-2744	A vulnerability, which was classified as critical, was found in zhijiantianya ruoyi-vue-pro 2.4.1. Affected is an unknown function of the file /admin-api/mp/material/upload-news-image of the component Material Upload Interface. The manipulation of the argument File leads to path traversal. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 25, 2025; 3:15:39 am -0400	*V4.0:*(not available) *V3.1:* 5.4 MEDIUM *V2.0:*(not available)
CVE-2025-2743	A vulnerability, which was classified as problematic, has been found in zhijiantianya ruoyi-vue-pro 2.4.1. This issue affects some unknown processing of the file /admin-api/mp/material/upload-temporary of the component Material Upload Interface. The manipulation of the argument File leads to path traversal. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 25, 2025; 3:15:38 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2742	A vulnerability classified as critical was found in zhijiantianya ruoyi-vue-pro 2.4.1. This vulnerability affects unknown code of the file /admin-api/mp/material/upload-permanent of the component Material Upload Interface. The manipulation of the argument File leads to path traversal. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 25, 2025; 3:15:38 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2716	A vulnerability classified as problematic was found in China Mobile P22g-Ciac 1.0.00.488. This vulnerability affects unknown code of the component Samba Path Handler. The manipulation leads to path traversal. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 24, 2025; 7:15:13 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2708	A vulnerability, which was classified as critical, was found in zhijiantianya ruoyi-vue-pro 2.4.1. This affects an unknown part of the file /admin-api/infra/file/upload of the component Backend File Upload Interface. The manipulation of the argument path leads to path traversal. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 24, 2025; 4:15:18 pm -0400	*V4.0:*(not available) *V3.1:* 5.4 MEDIUM *V2.0:*(not available)
CVE-2025-2749	An authenticated remote code execution in Kentico Xperience allows authenticated users Staging Sync Server to upload arbitrary data to path relative locations. This results in path traversal and arbitrary file upload, including content that can be executed server side leading to remote code execution.This issue affects Kentico Xperience through 13.0.178. **Published:** March 24, 2025; 3:15:52 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2707	A vulnerability, which was classified as critical, has been found in zhijiantianya ruoyi-vue-pro 2.4.1. Affected by this issue is some unknown functionality of the file /app-api/infra/file/upload of the component Front-End Store Interface. The manipulation of the argument path leads to path traversal. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 24, 2025; 3:15:50 pm -0400	*V4.0:*(not available) *V3.1:* 5.4 MEDIUM *V2.0:*(not available)
CVE-2025-27553	Relative Path Traversal vulnerability in Apache Commons VFS before 2.10.0. The FileObject API in Commons VFS has a 'resolveFile' method that takes a 'scope' parameter. Specifying 'NameScope.DESCENDENT' promises that "an exception is thrown if the resolved file is not a descendent of the base file". However, when the path contains encoded ".." characters (for example, "%2E%2E/bar.txt"), it might return file objects that are not a descendent of the base file, without throwing an exception. This issue affects Apache Commons VFS: before 2.10.0. Users are recommended to upgrade to version 2.10.0, which fixes the issue. **Published:** March 23, 2025; 11:15:13 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-1973	The Export and Import Users and Customers plugin for WordPress is vulnerable to Path Traversal in all versions up to, and including, 2.6.2 via the download_file() function. This makes it possible for authenticated attackers, with Administrator-level access and above, to read the contents of arbitrary log files on the server, which can contain sensitive information. **Published:** March 22, 2025; 8:15:26 am -0400	*V4.0:*(not available) *V3.1:* 4.9 MEDIUM *V2.0:*(not available)

Figure D.8: Eighth 9 of Path Traversal

CVE-2025-30343	A directory traversal issue was discovered in OpenSlides before 4.2.5. Files can be uploaded to OpenSlides meetings and organized in folders. The interface allows users to download a ZIP archive that contains all files in a folder and its subfolders. If an attacker specifies the title of a file or folder as a relative or absolute path (e.g., ../../../etc/passwd), the ZIP archive generated for download converts that title into a path. Depending on the extraction tool used by the user, this might overwrite files locally outside of the chosen directory. **Published:** March 21, 2025; 2:15:26 am -0400	V4.0:(not available) V3.1: 6.5 MEDIUM V2.0:(not available)
CVE-2024-9597	A Path Traversal vulnerability exists in the `/wipe_database` endpoint of parisneo/lollms version v12, allowing an attacker to delete any directory on the system. The vulnerability arises from improper validation of the `key` parameter, which is used to construct file paths. An attacker can exploit this by sending a specially crafted HTTP request to delete arbitrary directories. **Published:** March 20, 2025; 6:15:49 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-9415	A Path Traversal vulnerability exists in the file upload functionality of transformeroptimus/superagi version 0.0.14. This vulnerability allows an attacker to upload an arbitrary file to the server, potentially leading to remote code execution or overwriting any file on the server. **Published:** March 20, 2025; 6:15:48 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-8898	A path traversal vulnerability exists in the `install` and `uninstall` API endpoints of parisneo/lollms-webui version V12 (Strawberry). This vulnerability allows attackers to create or delete directories with arbitrary paths on the system. The issue arises due to insufficient sanitization of user-supplied input, which can be exploited to traverse directories outside the intended path. **Published:** March 20, 2025; 6:15:44 am -0400	V4.0:(not available) V3.1: 9.8 CRITICAL V2.0:(not available)
CVE-2024-8859	A path traversal vulnerability exists in mlflow/mlflow version 2.15.1. When users configure and use the dbfs service, concatenating the URL directly into the file protocol results in an arbitrary file read vulnerability. This issue occurs because only the path part of the URL is checked, while parts such as query and parameters are not handled. The vulnerability is triggered if the user has configured the dbfs service, and during usage, the service is mounted to a local directory. **Published:** March 20, 2025; 6:15:44 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-8769	A vulnerability in the `LockManager.release_locks` function in aimhubio/aim (commit bb76afe) allows for arbitrary file deletion through relative path traversal. The `run_hash` parameter, which is user-controllable, is concatenated without normalization as part of a path used to specify file deletion. This vulnerability is exposed through the `Repo._close_run()` method, which is accessible via the tracking server instruction API. As a result, an attacker can exploit this to delete any arbitrary file on the machine running the tracking server. **Published:** March 20, 2025; 6:15:44 am -0400	V4.0:(not available) V3.1: 9.1 CRITICAL V2.0:(not available)
CVE-2024-8581	A vulnerability in the `upload_app` function of parisneo/lollms-webui V12 (Strawberry) allows an attacker to delete any file or directory on the system. The function does not implement user input filtering with the `filename` value, causing a Path Traversal error. **Published:** March 20, 2025; 6:15:43 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-8551	A path traversal vulnerability exists in the save-workflow and load-workflow functionality of modelscope/agentscope versions prior to the fix. This vulnerability allows an attacker to read and write arbitrary JSON files on the filesystem, potentially leading to the exposure or modification of sensitive information such as configuration files, API keys, and hardcoded passwords. **Published:** March 20, 2025; 6:15:43 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-8537	A path traversal vulnerability exists in the modelscope/agentscope application, affecting all versions. The vulnerability is present in the /delete-workflow endpoint, allowing an attacker to delete arbitrary files from the filesystem. This issue arises due to improper input validation, enabling the attacker to manipulate file paths and delete sensitive files outside of the intended directory. **Published:** March 20, 2025; 6:15:42 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-8438	A path traversal vulnerability exists in modelscope/agentscope version v.0.0.4. The API endpoint `/api/file` does not properly sanitize the `path` parameter, allowing an attacker to read arbitrary files on the server. **Published:** March 20, 2025; 6:15:42 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)
CVE-2024-8248	A vulnerability in the normalizePath function in mintplex-labs/anything-llm version git 296f041 allows for path traversal, leading to arbitrary file read and write in the storage directory. This can result in privilege escalation from manager to admin. The issue is fixed in version 1.2.2. **Published:** March 20, 2025; 6:15:41 am -0400	V4.0:(not available) V3.x:(not available) V2.0:(not available)

« ‹ 4 5 6 7 8 9 10 11 12 13 › »

Figure D.9: Nineth 11 of Path Traversal

CVE-2024-8060	OpenWebUI version 0.3.0 contains a vulnerability in the audio API endpoint `/audio/api/v1/transcriptions` that allows for arbitrary file upload. The application performs insufficient validation on the `file.content_type` and allows user-controlled filenames, leading to a path traversal vulnerability. This can be exploited by an authenticated user to overwrite critical files within the Docker container, potentially leading to remote code execution as the root user. **Published:** March 20, 2025; 6:15:40 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-7776	A vulnerability in the `download_model` function of the onnx/onnx framework, before and including version 1.16.1, allows for arbitrary file overwrite due to inadequate prevention of path traversal attacks in malicious tar files. This vulnerability can be exploited by an attacker to overwrite files in the user's directory, potentially leading to remote command execution. **Published:** March 20, 2025; 6:15:37 am -0400	*V4.0:*(not available) *V3.1:* 9.1 CRITICAL *V2.0:*(not available)
CVE-2024-6583	A path traversal vulnerability exists in the latest version of stanfirard/quivr. This vulnerability allows an attacker to upload files to arbitrary paths in an S3 bucket by manipulating the file path in the upload request. **Published:** March 20, 2025; 6:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-6483	A vulnerability in the `runs/delete-batch` endpoint of aimhubio/aim version 3.19.3 allows for arbitrary file or directory deletion through path traversal. The endpoint does not mitigate path traversal when handling user-specified run-names, which are used to specify log/metadata files for deletion. This can be exploited to delete arbitrary files or directories, potentially causing denial of service or data loss. **Published:** March 20, 2025; 6:15:32 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-5752	A path traversal vulnerability exists in stitional/devika, specifically in the project creation functionality. In the affected version beacf6edaa205a5a5370525407a6db45137873b3, the project name is not validated, allowing an attacker to create a project with a crafted name that traverses directories. This can lead to arbitrary file overwrite when the application generates code and saves it to the specified project directory, potentially resulting in remote code execution. **Published:** March 20, 2025; 6:15:32 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-12389	A path traversal vulnerability exists in binary-husky/gpt_academic version git 310122f. The application supports the extraction of user-provided 7z files without proper validation. The Python py7zr package used for extraction does not guarantee that files will remain within the intended extraction directory. An attacker can exploit this vulnerability to perform arbitrary file writes, which can lead to remote code execution. **Published:** March 20, 2025; 6:15:28 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-12217	A vulnerability in the gradio-app/gradio repository, version git 67e4044, allows for path traversal on Windows OS. The implementation of the blocked_path functionality, which is intended to disallow users from reading certain files, is flawed. Specifically, while the application correctly blocks access to paths like 'C:/tmp/secret.txt', it fails to block access when using NTFS Alternate Data Streams (ADS) syntax, such as 'C:/tmp/secret.txt::$DATA'. This flaw can lead to unauthorized reading of blocked file paths. **Published:** March 20, 2025; 6:15:27 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-12216	A vulnerability in the `ImageClassificationDataset.from_csv()` API of the `dmlc/gluon-cv` repository, version 0.10.0, allows for arbitrary file write. The function downloads and extracts `tar.gz` files from URLs without proper sanitization, making it susceptible to a TarSlip vulnerability. Attackers can exploit this by crafting malicious tar files that, when extracted, can overwrite files on the victim's system via path traversal or faked symlinks. **Published:** March 20, 2025; 6:15:27 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-11170	A vulnerability in danny-avila/librechat version git 81f2936 allows for path traversal due to improper sanitization of file paths by the multer middleware. This can lead to arbitrary file write and potentially remote code execution. The issue is fixed in version 0.7.6. **Published:** March 20, 2025; 6:15:24 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-11037	A path traversal vulnerability exists in binary-husky/gpt_academic at commit 679352d, which allows an attacker to bypass the blocked_paths protection and read the config.py file containing sensitive information such as the OpenAI API key. This vulnerability is exploitable on Windows operating systems by accessing a specific URL that includes the absolute path of the project. **Published:** March 20, 2025; 6:15:23 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.10: Tenth 10 of Path Traversal

CVE-2024-10986	GPT Academic version 3.83 is vulnerable to a Local File Read (LFI) vulnerability through its HotReload function. This function can download and extract tar.gz files from arxiv.org. Despite implementing protections against path traversal, the application overlooks the Tarslip triggered by symlinks. This oversight allows attackers to read arbitrary local files from the victim server. **Published:** March 20, 2025; 6:15:22 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10902	In eosphoros-ai/db-gpt version v0.6.0, the web API `POST /v1/personal/agent/upload` is vulnerable to Arbitrary File Upload with Path Traversal. This vulnerability allows unauthorized attackers to upload arbitrary files to the victim's file system at any location. The impact of this vulnerability includes the potential for remote code execution (RCE) by writing malicious files, such as a malicious `__init__.py` in the Python's `/site-packages/` directory. **Published:** March 20, 2025; 6:15:21 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10833	eosphoros-ai/db-gpt version 0.6.0 is vulnerable to an arbitrary file write through the knowledge API. The endpoint for uploading files as 'knowledge' is susceptible to absolute path traversal, allowing attackers to write files to arbitrary locations on the target server. This vulnerability arises because the 'doc_file.filename' parameter is user-controllable, enabling the construction of absolute paths. **Published:** March 20, 2025; 6:15:20 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10831	In eosphoros-ai/db-gpt version 0.6.0, the endpoint for uploading files is vulnerable to absolute path traversal. This vulnerability allows an attacker to upload arbitrary files to arbitrary locations on the target server. The issue arises because the `file_key` and `doc_file.filename` parameters are user-controllable, enabling the construction of paths outside the intended directory. This can lead to overwriting essential system files, such as SSH keys, for further exploitation. **Published:** March 20, 2025; 6:15:20 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10830	A Path Traversal vulnerability exists in the eosphoros-ai/db-gpt version 0.6.0 at the API endpoint `/v1/resource/file/delete`. This vulnerability allows an attacker to delete any file on the server by manipulating the `file_key` parameter. The `file_key` parameter is not properly sanitized, enabling an attacker to specify arbitrary file paths. If the specified file exists, the application will delete it. **Published:** March 20, 2025; 6:15:20 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10648	A path traversal vulnerability exists in the Gradio Audio component of gradio-app/gradio, as of version git 98cbcae. This vulnerability allows an attacker to control the format of the audio file, leading to arbitrary file content deletion. By manipulating the output format, an attacker can reset any file to an empty file, causing a denial of service (DOS) on the server. **Published:** March 20, 2025; 6:15:18 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10513	A path traversal vulnerability exists in the 'document uploads manager' feature of mintplex-labs/anything-llm, affecting the latest version prior to 1.2.2. This vulnerability allows users with the 'manager' role to access and manipulate the 'anythingllm.db' database file. By exploiting the vulnerable endpoint '/api/document/move-files', an attacker can move the database file to a publicly accessible directory, download it, and subsequently delete it. This can lead to unauthorized access to sensitive data, privilege escalation, and potential data loss. **Published:** March 20, 2025; 6:15:17 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10361	An arbitrary file deletion vulnerability exists in danny-avila/librechat version v0.7.5-rc2, specifically within the /api/files endpoint. This vulnerability arises from improper input validation, allowing path traversal techniques to delete arbitrary files on the server. Attackers can exploit this to bypass security mechanisms and delete files outside the intended directory, including critical system files, user data, or application resources. This vulnerability impacts the integrity and availability of the system. **Published:** March 20, 2025; 6:15:16 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-10019	A vulnerability in the `start_app_server` function of parisneo/lollms-webui V12 (Strawberry) allows for path traversal and OS command injection. The function does not properly sanitize the `app_name` parameter, enabling an attacker to upload a malicious `server.py` file and execute arbitrary code by exploiting the path traversal vulnerability. **Published:** March 20, 2025; 6:15:14 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure D.11: 11th 9 of Path Traversal

E

Appendix 5 Appearance of Buffer Overflow in NVD

(20th March 2025 to 20th April 2025)

Totally 183.

CVE-2025-3820	A vulnerability was found in Tenda W12 and i24 3.0.0.4(2887)/3.0.0.5(3644) and classified as critical. Affected by this issue is the function cgiSysUplinkCheckSet of the file /bin/httpd. The manipulation of the argument hostIp1/hostIp2 leads to stack-based buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 19, 2025; 5:15:45 pm -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3803	A vulnerability was found in Tenda W12 and i24 3.0.0.4(2887)/3.0.0.5(3644). It has been rated as critical. This issue affects the function cgiSysScheduleRebootSet of the file /bin/httpd. The manipulation of the argument rebootDate leads to stack-based buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 19, 2025; 11:15:47 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3802	A vulnerability was found in Tenda W12 and i24 3.0.0.4(2887)/3.0.0.5(3644). It has been declared as critical. This vulnerability affects the function cgiPingSet of the file /bin/httpd. The manipulation of the argument pingIP leads to stack-based buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 19, 2025; 11:15:46 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3791	A vulnerability classified as critical was found in symisc UnQLite up to 957c377cb691a4f617db9aba5cc46d90425071e2. This vulnerability affects the function jx9MemObjStore of the file /data/src/benchmarks/unqlite/unqlite.c. The manipulation leads to heap-based buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. Continious delivery with rolling releases is used by this product. Therefore, no version details of affected nor updated releases are available. **Published:** April 18, 2025; 11:15:59 am -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-29625	A buffer overflow vulnerability in Astrolog v7.70 allows attackers to execute arbitrary code or cause a Denial of Service (DoS) via an overly long environment variable passed to FileOpen function. **Published:** April 18, 2025; 11:15:58 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3786	A vulnerability was found in Tenda AC15 up to 15.03.05.19 and classified as critical. This issue affects the function fromSetWirelessRepeat of the file /goform/WifiExtraSet. The manipulation of the argument mac leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 18, 2025; 5:15:15 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3785	A vulnerability has been found in D-Link DWR-M961 1.1.36 and classified as critical. This vulnerability affects unknown code of the file /boafrm/formStaticDHCP of the component Authorization Interface. The manipulation of the argument Hostname leads to stack-based buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. Upgrading to version 1.1.49 is able to address this issue. It is recommended to upgrade the affected component. **Published:** April 18, 2025; 5:15:15 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-42599	Active! mail 6 BuildInfo: 6.60.05008561 and earlier contains a stack-based buffer overflow vulnerability. Receiving a specially crafted request created and sent by a remote unauthenticated attacker may lead to arbitrary code execution and/or a denial-of-service (DoS) condition. **Published:** April 18, 2025; 12:15:30 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3763	A vulnerability classified as critical has been found in SourceCodester Phone Management System 1.0. This affects the function main of the component Password Handler. The manipulation of the argument s leads to buffer overflow. Local access is required to approach this attack. The exploit has been disclosed to the public and may be used. **Published:** April 17, 2025; 3:16:15 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)

Figure E.1: First 9 of Buffer Overflow

CVE-2025-3762	A vulnerability was found in PCMan FTP Server 2.0.7. It has been rated as critical. Affected by this issue is some unknown functionality of the component MPUT Command Handler. The manipulation leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 17, 2025; 3:16:15 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-25455	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via wanMTU2. **Published:** April 17, 2025; 2:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-25454	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via wanSpeed2. **Published:** April 17, 2025; 2:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-25457	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via cloneType2. **Published:** April 17, 2025; 12:15:34 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29047	Buffer Overflow vulnerability inALFA WiFi CampPro router ALFA_CAMPRO-co-2.29 allows a remote attacker to execute arbitrary code via the hiddenIndex in the function StorageEditUser **Published:** April 17, 2025; 11:15:55 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29046	Buffer Overflow vulnerability inALFA WiFi CampPro router ALFA_CAMPRO-co-2.29 allows a remote attacker to execute arbitrary code via the GAPSMinute3 key value **Published:** April 17, 2025; 11:15:55 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.2: Second 6 of Buffer Overflow

CVE-2025-3786	A vulnerability was found in Tenda AC15 up to 15.03.05.19 and classified as critical. This issue affects the function fromSetWirelessRepeat of the file /goform/WifiExtraSet. The manipulation of the argument mac leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 18, 2025; 5:15:15 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3785	A vulnerability has been found in D-Link DWR-M961 1.1.36 and classified as critical. This vulnerability affects unknown code of the file /boafrm/formStaticDHCP of the component Authorization Interface. The manipulation of the argument Hostname leads to stack-based buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. Upgrading to version 1.1.49 is able to address this issue. It is recommended to upgrade the affected component. **Published:** April 18, 2025; 5:15:15 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-42599	Active! mail 6 BuildInfo: 6.60.05008561 and earlier contains a stack-based buffer overflow vulnerability. Receiving a specially crafted request created and sent by a remote unauthenticated attacker may lead to arbitrary code execution and/or a denial-of-service (DoS) condition. **Published:** April 18, 2025; 12:15:30 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3763	A vulnerability classified as critical has been found in SourceCodester Phone Management System 1.0. This affects the function main of the component Password Handler. The manipulation of the argument s leads to buffer overflow. Local access is required to approach this attack. The exploit has been disclosed to the public and may be used. **Published:** April 17, 2025; 3:16:15 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-3762	A vulnerability was found in PCMan FTP Server 2.0.7. It has been rated as critical. Affected by this issue is some unknown functionality of the component MPUT Command Handler. The manipulation leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 17, 2025; 3:16:15 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-25455	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via wanMTU2. **Published:** April 17, 2025; 2:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-25454	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via wanSpeed2. **Published:** April 17, 2025; 2:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-25457	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via cloneType2. **Published:** April 17, 2025; 12:15:34 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29047	Buffer Overflow vulnerability inALFA WiFi CampPro router ALFA_CAMPRO-co-2.29 allows a remote attacker to execute arbitrary code via the hiddenIndex in the function StorageEditUser **Published:** April 17, 2025; 11:15:55 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29046	Buffer Overflow vulnerability inALFA WiFi CampPro router ALFA_CAMPRO-co-2.29 allows a remote attacker to execute arbitrary code via the GAPSMinute3 key value **Published:** April 17, 2025; 11:15:55 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.3: Third 10 of Buffer Overflow

CVE-2025-29045	Buffer Overflow vulnerability in ALFA_CAMPRO-co-2.29 allows a remote attacker to execute arbitrary code via the newap_text_0 key value **Published:** April 17, 2025; 11:15:54 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29044	Buffer Overflow vulnerability in Netgear- R61 router V1.0.1.28 allows a remote attacker to execute arbitrary code via the QUERY_STRING key value **Published:** April 17, 2025; 11:15:54 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3728	A vulnerability classified as critical was found in SourceCodester Simple Hotel Booking System 1.0. This vulnerability affects the function Login. The manipulation of the argument uname leads to buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 5:15:48 pm -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-3727	A vulnerability classified as critical has been found in PCMan FTP Server 2.0.7. This affects an unknown part of the component STATUS Command Handler. The manipulation leads to buffer overflow. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 5:15:48 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3619	Heap buffer overflow in Codecs in Google Chrome on Windows prior to 135.0.7049.95 allowed a remote attacker to potentially exploit heap corruption via a crafted HTML page. (Chromium security severity: Critical) **Published:** April 16, 2025; 5:15:47 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3726	A vulnerability was found in PCMan FTP Server 2.0.7. It has been rated as critical. Affected by this issue is some unknown functionality of the component CD Command Handler. The manipulation leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 4:15:20 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3725	A vulnerability was found in PCMan FTP Server 2.0.7. It has been declared as critical. Affected by this vulnerability is an unknown functionality of the component MIC Command Handler. The manipulation leads to buffer overflow. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 4:15:20 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3724	A vulnerability was found in PCMan FTP Server 2.0.7. It has been classified as critical. Affected is an unknown function of the component DIR Command Handler. The manipulation leads to buffer overflow. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 4:15:19 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3723	A vulnerability was found in PCMan FTP Server 2.0.7 and classified as critical. This issue affects some unknown processing of the component MDTM Command Handler. The manipulation leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 4:15:19 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)

Figure E.4: Fourth 9 of Buffer Overflow

CVE-2025-22104	In the Linux kernel, the following vulnerability has been resolved: ibmvnic: Use kernel helpers for hex dumps Previously, when the driver was printing hex dumps, the buffer was cast to an 8 byte long and printed using string formatters. If the buffer size was not a multiple of 8 then a read buffer overflow was possible. Therefore, create a new ibmvnic function that loops over a buffer and calls hex_dump_to_buffer instead. This patch address KASAN reports like the one below: ibmvnic 30000003 env3: Login Buffer: ibmvnic 30000003 env3: 01000000af000000 <...> ibmvnic 30000003 env3: 2e6d62692e736261 ibmvnic 30000003 env3: 65050003006d6f63 == BUG: KASAN: slab-out-of-bounds in ibmvnic_login+0xacc/0xffc [ibmvnic] Read of size 8 at addr c0000001331a9aa8 by task ip/17681 <...> Allocated by task 17681: <...> ibmvnic_login+0x2f0/0xffc [ibmvnic] ibmvnic_open+0x148/0x308 [ibmvnic] __dev_open+0x1ac/0x304 <...> The buggy address is located 168 bytes inside of allocated 175-byte region [c0000001331a9a00, c0000001331a9aaf) <...> == ibmvnic 30000003 env3: 000000000033766e **Published:** April 16, 2025; 11:16:04 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.5: Fifth 1 of Buffer Overflow

CVE-2025-22059	In the Linux kernel, the following vulnerability has been resolved: udp: Fix multiple wraparounds of sk->sk_rmem_alloc. __udp_enqueue_schedule_skb() has the following condition: if (atomic_read(&sk->sk_rmem_alloc) > sk->sk_rcvbuf) goto drop; sk->sk_rcvbuf is initialised by net.core.rmem_default and later can be configured by SO_RCVBUF, which is limited by net.core.rmem_max, or SO_RCVBUFFORCE. If we set INT_MAX to sk->sk_rcvbuf, the condition is always false as sk->sk_rmem_alloc is also signed int. Then, the size of the incoming skb is added to sk->sk_rmem_alloc unconditionally. This results in integer overflow (possibly multiple times) on sk->sk_rmem_alloc and allows a single socket to have skb up to net.core.udp_mem[1]. For example, if we set a large value to udp_mem[1] and INT_MAX to sk->sk_rcvbuf and flood packets to the socket, we can see multiple overflows: # cat /proc/net/sockstat | grep UDP: UDP: inuse 3 mem 7956736 <-- (7956736 << 12) bytes > INT_MAX * 15 ^- PAGE_SHIFT # ss -uam State Recv-Q ... UNCONN -1757018048 ... <-- flipping the sign repeatedly skmem: (r2537949248,rb2147483646,t0,tb212992,f1984,w0,o0,bl0,d0) Previously, we had a boundary check for INT_MAX, which was removed by commit 6a1f12dd85a8 ("udp: relax atomic operation on sk->sk_rmem_alloc"). A complete fix would be to revert it and cap the right operand by INT_MAX: rmem = atomic_add_return(size, &sk->sk_rmem_alloc); if (rmem > min(size + (unsigned int)sk->sk_rcvbuf, INT_MAX)) goto uncharge_drop; but we do not want to add the expensive atomic_add_return() back just for the corner case. Casting rmem to unsigned int prevents multiple wraparounds, but we still allow a single wraparound. # cat /proc/net/sockstat | grep UDP: UDP: inuse 3 mem 524288 <-- (INT_MAX + 1) >> 12 # ss -uam State Recv-Q ... UNCONN -2147482816 ... <-- INT_MAX + 831 bytes skmem:(r2147484480,rb2147483646,t0,tb212992,f3264,w0,o0,bl0,d14468947) So, let's define rmem and rcvbuf as unsigned int and check skb->truesize only when rcvbuf is large enough to lower the overflow possibility. Note that we still have a small chance to see overflow if multiple skbs to the same socket are processed on different core at the same time and each size does not exceed the limit but the total size does. Note also that we must ignore skb->truesize for a small buffer as explained in commit 363dc73acacb ("udp: be less conservative with sock rmem accounting"). **Published:** April 16, 2025; 11:15:59 am -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)
CVE-2025-3693	A vulnerability was found in Tenda W12 3.0.0.5. It has been rated as critical. Affected by this issue is the function cgiWifiRadioSet of the file /bin/httpd. The manipulation leads to stack-based buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 10:15:28 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3683	A vulnerability was found in PCMan FTP Server 2.0.7. It has been declared as critical. This vulnerability affects unknown code of the component SIZE Command Handler. The manipulation leads to buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 7:15:43 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3682	A vulnerability was found in PCMan FTP Server 2.0.7. It has been classified as critical. This affects an unknown part of the component PASV Command Handler. The manipulation leads to buffer overflow. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 7:15:43 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3681	A vulnerability was found in PCMan FTP Server 2.0.7 and classified as critical. Affected by this issue is some unknown functionality of the component MODE Command Handler. The manipulation leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 6:15:15 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3680	A vulnerability has been found in PCMan FTP Server 2.0.7 and classified as critical. Affected by this vulnerability is an unknown functionality of the component LANG Command Handler. The manipulation leads to buffer overflow. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 6:15:15 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)

Figure E.6: Sixth 6 of Buffer Overflow

CVE-2025-3679	A vulnerability, which was classified as critical, was found in PCMan FTP Server 2.0.7. Affected is an unknown function of the component HOST Command Handler. The manipulation leads to buffer overflow. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 6:15:15 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3678	A vulnerability, which was classified as critical, has been found in PCMan FTP Server 2.0.7. This issue affects some unknown processing of the component HELP Command Handler. The manipulation leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 16, 2025; 5:15:28 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-25458	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via serverName2. **Published:** April 15, 2025; 7:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-25453	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via serviceName2. **Published:** April 15, 2025; 7:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2497	A maliciously crafted DWG file, when parsed through Autodesk Revit, can cause a Stack-Based Buffer Overflow vulnerability. A malicious actor can leverage this vulnerability to execute arbitrary code in the context of the current process. **Published:** April 15, 2025; 5:15:56 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-25456	Tenda AC10 V4.0si_V16.03.10.20 is vulnerable to Buffer Overflow in AdvSetMacMtuWan via mac2. **Published:** April 15, 2025; 3:16:07 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-28136	TOTOLINK A800R V4.1.2cu.5137_B20200730 was found to contain a buffer overflow vulnerability in the downloadFile.cgi. **Published:** April 15, 2025; 10:15:41 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-24797	Meshtastic is an open source mesh networking solution. A fault in the handling of mesh packets containing invalid protobuf data can result in an attacker-controlled buffer overflow, allowing an attacker to hijack execution flow, potentially resulting in remote code execution. This attack does not require authentication or user interaction, as long as the target device rebroadcasts packets on the default channel. This vulnerability fixed in 2.6.2. **Published:** April 14, 2025; 8:15:14 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3588	A vulnerability, which was classified as problematic, has been found in joelittlejohn jsonschema2pojo 1.2.2. This issue affects the function apply of the file org/jsonschema2pojo/rules/SchemaRule.java of the component JSON File Handler. The manipulation leads to stack-based buffer overflow. Attacking locally is a requirement. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** April 14, 2025; 5:15:18 pm -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-3277	An integer overflow can be triggered in SQLite's `concat_ws()` function. The resulting, truncated integer is then used to allocate a buffer. When SQLite then writes the resulting string to the buffer, it uses the original, untruncated size and thus a wild Heap Buffer overflow of size ~4GB can be triggered. This can result in arbitrary code execution. **Published:** April 14, 2025; 1:15:27 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.7: Seventh 10 of Buffer Overflow

CVE-2025-31344	Heap-based Buffer Overflow vulnerability in openEuler giflib on Linux. This vulnerability is associated with program files gif2rgb.C. This issue affects giflib: through 5.2.2. **Published:** April 14, 2025; 4:15:13 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3549	A vulnerability, which was classified as critical, was found in Open Asset Import Library Assimp 5.4.3. Affected is the function Assimp::MD3Importer::ValidateSurfaceHeaderOffsets of the file code/AssetLib/MD3/MD3Loader.cpp of the component File Handler. The manipulation leads to heap-based buffer overflow. The attack needs to be approached locally. The exploit has been disclosed to the public and may be used. **Published:** April 13, 2025; 11:15:16 pm -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-3548	A vulnerability, which was classified as critical, has been found in Open Asset Import Library Assimp up to 5.4.3. This issue affects the function aiString::Set in the library include/assimp/types.h of the component File Handler. The manipulation leads to heap-based buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. It is recommended to apply a patch to fix this issue. **Published:** April 13, 2025; 11:15:16 pm -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-3538	A vulnerability was found in D-Link DI-8100 16.07.26A1. It has been rated as critical. This issue affects the function auth_asp of the file /auth.asp of the component jhttpd. The manipulation of the argument callback leads to stack-based buffer overflow. The attack needs to be approached within the local network. The exploit has been disclosed to the public and may be used. **Published:** April 13, 2025; 3:15:14 pm -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)

Figure E.8: Eighth 4 of Buffer Overflow

CVE-2024-56406	A heap buffer overflow vulnerability was discovered in Perl. Release branches 5.34, 5.36, 5.38 and 5.40 are affected, including development versions from 5.33.1 through 5.41.10. When there are non-ASCII bytes in the left-hand-side of the `tr` operator, `S_do_trans_invmap` can overflow the destination pointer `d`. $ perl -e '$_ = "\x{FF}" x 1000000; tr/\xFF/\x{100}/;' Segmentation fault (core dumped) It is believed that this vulnerability can enable Denial of Service and possibly Code Execution attacks on platforms that lack sufficient defenses. **Published:** April 13, 2025; 10:15:14 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-23388	A Stack-based Buffer Overflow vulnerability in SUSE rancher allows for denial of service.This issue affects rancher: from 2.8.0 before 2.8.13, from 2.9.0 before 2.9.7, from 2.10.0 before 2.10.3. **Published:** April 11, 2025; 7:15:42 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3512	There is a Heap-based Buffer Overflow vulnerability in QTextMarkdownImporter. This requires an incorrectly formatted markdown file to be passed to QTextMarkdownImporter to trigger the overflow.This issue affects Qt from 6.8.0 to 6.8.4. Versions up to 6.6.0 are known to be unaffected, and the fix is in 6.8.4 and later. **Published:** April 11, 2025; 4:15:15 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30644	A Heap-based Buffer Overflow vulnerability in the flexible PIC concentrator (FPC) of Juniper Networks Junos OS on EX2300, EX3400, EX4100, EX4300, EX4300MP, EX4400, EX4600, EX4650-48Y, and QFX5k Series allows an attacker to send a specific DHCP packet to the device, leading to an FPC crash and restart, resulting in a Denial of Service (DoS). Continued receipt and processing of this packet will create a sustained Denial of Service (DoS) condition. Under a rare timing scenario outside the attacker's control, memory corruption may be observed when DHCP Option 82 is enabled, leading to an FPC crash and affecting packet forwarding. Due to the nature of the heap-based overflow, exploitation of this vulnerability could also lead to remote code execution within the FPC, resulting in complete control of the vulnerable component. This issue affects Junos OS on EX2300, EX3400, EX4100, EX4300, EX4300MP, EX4400, EX4600, EX4650-48Y, and QFX5k Series: * All versions before 21.4R3-S9, * from 22.2 before 22.2R3-S5, * from 22.4 before 22.4R3-S5, * from 23.2 before 23.2R2-S3, * from 23.4 before 23.4R2-S3, * from 24.2 before 24.2R2. **Published:** April 09, 2025; 4:15:27 pm -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2025-32464	HAProxy 2.2 through 3.1.6, in certain uncommon configurations, has a sample_conv_regsub heap-based buffer overflow because of mishandling of the replacement of multiple short patterns with a longer one. **Published:** April 08, 2025; 11:15:16 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29988	Dell Client Platform BIOS contains a Stack-based Buffer Overflow Vulnerability. A high privileged attacker with local access could potentially exploit this vulnerability, leading to arbitrary code execution. **Published:** April 08, 2025; 11:15:15 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30299	Adobe Framemaker versions 2020.8, 2022.6 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 3:15:50 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-30298	Adobe Framemaker versions 2020.8, 2022.6 and earlier are affected by a Stack-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 3:15:49 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-30295	Adobe Framemaker versions 2020.8, 2022.6 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 3:15:49 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)

Figure E.9: Ninth 9 of Buffer Overflow

CVE-2025-27752	Heap-based buffer overflow in Microsoft Office Excel allows an unauthorized attacker to execute code locally. **Published:** April 08, 2025; 2:16:04 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27490	Heap-based buffer overflow in Windows Bluetooth Service allows an authorized attacker to elevate privileges locally. **Published:** April 08, 2025; 2:15:59 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27487	Heap-based buffer overflow in Remote Desktop Client allows an authorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:59 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27481	Stack-based buffer overflow in Windows Telephony Service allows an unauthorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:58 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27478	Heap-based buffer overflow in Windows Local Security Authority (LSA) allows an authorized attacker to elevate privileges locally. **Published:** April 08, 2025; 2:15:58 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27477	Heap-based buffer overflow in Windows Telephony Service allows an unauthorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:57 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27199	Animate versions 24.0.7, 23.0.10 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 2:15:55 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-27198	Photoshop Desktop versions 25.12.1, 26.4.1 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 2:15:55 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-27196	Premiere Pro versions 25.1, 24.6.4 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 2:15:55 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-27195	Media Encoder versions 25.1, 24.6.4 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 2:15:55 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-27193	Bridge versions 14.1.5, 15.0.2 and earlier are affected by a Heap-based Buffer Overflow vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file. **Published:** April 08, 2025; 2:15:54 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)

Figure E.10: 10th 11 of Buffer Overflow

CVE-2025-26688	Stack-based buffer overflow in Microsoft Virtual Hard Drive allows an authorized attacker to elevate privileges locally. **Published:** April 08, 2025; 2:15:53 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-26674	Heap-based buffer overflow in Windows Media allows an authorized attacker to execute code locally. **Published:** April 08, 2025; 2:15:51 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-26668	Heap-based buffer overflow in Windows Routing and Remote Access Service (RRAS) allows an unauthorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:50 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-26666	Heap-based buffer overflow in Windows Media allows an authorized attacker to execute code locally. **Published:** April 08, 2025; 2:15:49 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-21222	Heap-based buffer overflow in Windows Telephony Service allows an unauthorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:45 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-21221	Heap-based buffer overflow in Windows Telephony Service allows an unauthorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:45 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-21205	Heap-based buffer overflow in Windows Telephony Service allows an unauthorized attacker to execute code over a network. **Published:** April 08, 2025; 2:15:45 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3289	A local code execution vulnerability exists in the Rockwell Automation Arena® due to a stack-based memory buffer overflow. The flaw is result of improper validation of user-supplied data. If exploited a threat actor can disclose information and execute arbitrary code on the system. To exploit the vulnerability a legitimate user must open a malicious DOE file. **Published:** April 08, 2025; 12:15:29 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3287	A local code execution vulnerability exists in the Rockwell Automation Arena® due to a stack-based memory buffer overflow. The flaw is result of improper validation of user-supplied data. If exploited a threat actor can disclose information and execute arbitrary code on the system. To exploit the vulnerability a legitimate user must open a malicious DOE file. **Published:** April 08, 2025; 12:15:28 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3409	A vulnerability classified as critical has been found in Nothings stb up to f056911. This affects the function stb_include_string. The manipulation of the argument path_to_includes leads to stack-based buffer overflow. It is possible to initiate the attack remotely. This product does not use versioning. This is why information about affected and unaffected releases are unavailable. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** April 08, 2025; 1:15:40 am -0400	*V4.0:*(not available) *V3.1:* 6.3 MEDIUM *V2.0:*(not available)

Figure E.11: 11th 10 of Buffer Overflow

CVE-2025-29769	libvips is a demand-driven, horizontally threaded image processing library. The heifsave operation could incorrectly determine the presence of an alpha channel in an input when it was not possible to determine the colour interpretation, known internally within libvips as "multiband". There aren't many ways to create a "multiband" input, but it is possible with a well-crafted TIFF image. If a "multiband" TIFF input image had 4 channels and HEIF-based output was requested, this led to libvips creating a 3 channel HEIF image without an alpha channel but then attempting to write 4 channels of data. This caused a heap buffer overflow, which could crash the process. This vulnerability is fixed in 8.16.1. **Published:** April 07, 2025; 4:15:21 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29482	Buffer Overflow vulnerability in libheif 1.19.7 allows a local attacker to execute arbitrary code via the SAO (Sample Adaptive Offset) processing of libde265. **Published:** April 07, 2025; 4:15:20 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29481	Buffer Overflow vulnerability in libbpf 1.5.0 allows a local attacker to execute arbitrary code via the bpf_object__init_prog` function of libbpf. **Published:** April 07, 2025; 4:15:20 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29480	Buffer Overflow vulnerability in gdal 3.10.2 allows a local attacker to cause a denial of service via the OGRSpatialReference::Release function. **Published:** April 07, 2025; 4:15:20 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29087	In SQLite 3.44.0 through 3.49.0 before 3.49.1, the concat_ws() SQL function can cause memory to be written beyond the end of a malloc-allocated buffer. If the separator argument is attacker-controlled and has a large string (e.g., 2MB or more), an integer overflow occurs in calculating the size of the result buffer, and thus malloc may not allocate enough memory. **Published:** April 07, 2025; 4:15:20 pm -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2025-3380	A vulnerability, which was classified as critical, has been found in PCMan FTP Server 2.0.7. Affected by this issue is some unknown functionality of the component FEAT Command Handler. The manipulation leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 3:15:57 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3379	A vulnerability classified as critical was found in PCMan FTP Server 2.0.7. Affected by this vulnerability is an unknown functionality of the component EPSV Command Handler. The manipulation leads to buffer overflow. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 3:15:57 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3378	A vulnerability classified as critical has been found in PCMan FTP Server 2.0.7. Affected is an unknown function of the component EPRT Command Handler. The manipulation leads to buffer overflow. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 2:15:45 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3377	A vulnerability was found in PCMan FTP Server 2.0.7. It has been rated as critical. This issue affects some unknown processing of the component ENC Command Handler. The manipulation leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 2:15:45 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3376	A vulnerability was found in PCMan FTP Server 2.0.7. It has been declared as critical. This vulnerability affects unknown code of the component CONF Command Handler. The manipulation leads to buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 1:15:39 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)

Figure E.12: 12th 10 of Buffer Overflow

CVE-2025-3375	A vulnerability was found in PCMan FTP Server 2.0.7. It has been classified as critical. This affects an unknown part of the component CDUP Command Handler. The manipulation leads to buffer overflow. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 1:15:39 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3374	A vulnerability was found in PCMan FTP Server 2.0.7 and classified as critical. Affected by this issue is some unknown functionality of the component CCC Command Handler. The manipulation leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 12:15:27 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3373	A vulnerability has been found in PCMan FTP Server 2.0.7 and classified as critical. Affected by this vulnerability is an unknown functionality of the component SITE CHMOD Command Handler. The manipulation leads to buffer overflow. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 12:15:27 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3372	A vulnerability, which was classified as critical, was found in PCMan FTP Server 2.0.7. Affected is an unknown function of the component MKDIR Command Handler. The manipulation leads to buffer overflow. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 11:15:46 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3371	A vulnerability, which was classified as critical, has been found in PCMan FTP Server 2.0.7. This issue affects some unknown processing of the component DELETE Command Handler. The manipulation leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 11:15:46 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3360	A flaw was found in GLib. An integer overflow and buffer under-read occur when parsing a long invalid ISO 8601 timestamp with the g_date_time_new_from_iso8601() function. **Published:** April 07, 2025; 9:15:43 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3349	A vulnerability, which was classified as critical, has been found in PCMan FTP Server 2.0.7. This issue affects some unknown processing of the component SYST Command Handler. The manipulation leads to buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 7:15:53 am -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3346	A vulnerability was found in Tenda AC7 15.03.06.44. It has been rated as critical. Affected by this issue is the function formSetPPTPServer of the file /goform/SetPptpServerCfg. The manipulation of the argument pptp_server_start_ip/pptp_server_end_ip leads to buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 07, 2025; 6:15:15 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2024-58116	Buffer overflow vulnerability in the SVG parsing module of the ArkUI framework Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:18 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2024-58115	Buffer overflow vulnerability in the SVG parsing module of the ArkUI framework Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:18 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2024-58110	Buffer overflow vulnerability in the codec module Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:16 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)

Figure E.13: 13th 11 of Buffer Overflow

CVE-2024-58109	Buffer overflow vulnerability in the codec module Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:15 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2024-58108	Buffer overflow vulnerability in the codec module Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:15 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2024-58107	Buffer overflow vulnerability in the codec module Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:15 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2024-58106	Buffer overflow vulnerability in the codec module Impact: Successful exploitation of this vulnerability may affect availability. **Published:** April 07, 2025; 12:15:15 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2025-3328	A vulnerability was found in Tenda AC1206 15.03.06.23. It has been classified as critical. Affected is the function form_fast_setting_wifi_set of the file /goform/fast_setting_wifi_set. The manipulation of the argument ssid/timeZone leads to buffer overflow. It is possible to launch the attack remotely. The exploit has been disclosed to the public and may be used. Other parameters might be affected as well. **Published:** April 06, 2025; 9:15:42 pm -0400	*V4.0:* 8.7 HIGH *V3.1:* 8.8 HIGH *V2.0:* 9.0 HIGH
CVE-2025-3266	A vulnerability, which was classified as critical, has been found in qinguoyi TinyWebServer up to 1.0. Affected by this issue is some unknown functionality of the file /http/http_conn.cpp. The manipulation of the argument name/password leads to stack-based buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 04, 2025; 4:15:18 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3259	A vulnerability, which was classified as critical, has been found in Tenda RX3 16.03.13.11. This issue affects the function formSetDeviceName of the file /goform/SetOnlineDevName. The manipulation of the argument devName leads to stack-based buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 04, 2025; 2:15:49 pm -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-29476	Buffer Overflow vulnerability in compress_chunk_fuzzer with oss-fuzz on commit 16450518afddcb3139de627157208e49bfef6987 in c-blosc2 v.2.17.0 and before. **Published:** April 04, 2025; 2:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3194	Versions of the package bigint-buffer from 0.0.0 are vulnerable to Buffer Overflow in the toBigIntLE() function. Attackers can exploit this to crash the application. **Published:** April 04, 2025; 1:15:45 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.14: 14th 9 of Buffer Overflow

CVE-2025-3203	A vulnerability classified as problematic was found in Tenda W18E 16.01.0.11. Affected by this vulnerability is the function formSetAccountList of the file /goform/setModules. The manipulation of the argument Password leads to stack-based buffer overflow. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** April 03, 2025; 11:15:14 pm -0400	*V4.0:* 5.3 MEDIUM *V3.1:* 4.3 MEDIUM *V2.0:* 4.0 MEDIUM
CVE-2025-3196	A vulnerability, which was classified as critical, was found in Open Asset Import Library Assimp 5.4.3. Affected is the function Assimp::MD2Importer::InternReadFile in the library code/AssetLib/MD2/MD2Loader.cpp of the component Malformed File Handler. The manipulation of the argument Name leads to stack-based buffer overflow. The attack needs to be approached locally. The exploit has been disclosed to the public and may be used. It is recommended to upgrade the affected component. **Published:** April 03, 2025; 10:15:18 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29462	A buffer overflow vulnerability has been discovered in Tenda Ac15 V15.13.07.13. The vulnerability occurs when the webCgiGetUploadFile function calls the socketRead function to process HTTP request messages, resulting in the overwriting of a buffer on the stack. **Published:** April 03, 2025; 4:15:24 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3166	A vulnerability classified as critical was found in code-projects Product Management System 1.0. This vulnerability affects the function search_item of the component Search Product Menu. The manipulation of the argument target leads to stack-based buffer overflow. Local access is required to approach this attack. The exploit has been disclosed to the public and may be used. **Published:** April 03, 2025; 1:15:31 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-22457	A stack-based buffer overflow in Ivanti Connect Secure before version 22.7R2.6, Ivanti Policy Secure before version 22.7R1.4, and Ivanti ZTA Gateways before version 22.8R2.2 allows a remote unauthenticated attacker to achieve remote code execution. **Published:** April 03, 2025; 12:15:35 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-3161	A vulnerability was found in Tenda AC10 16.03.10.13 and classified as critical. This issue affects the function ShutdownSetAdd of the file /goform/ShutdownSetAdd. The manipulation of the argument list leads to stack-based buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** April 03, 2025; 11:15:53 am -0400	*V4.0:*(not available) *V3.1:* 8.8 HIGH *V2.0:*(not available)
CVE-2025-3159	A vulnerability, which was classified as critical, was found in Open Asset Import Library Assimp 5.4.3. This affects the function Assimp::ASE::Parser::ParseLV4MeshBonesVertices of the file code/AssetLib/ASE/ASEParser.cpp of the component ASE File Handler. The manipulation leads to heap-based buffer overflow. The attack needs to be approached locally. The exploit has been disclosed to the public and may be used. The identifier of the patch is e8a6286542924e628e02749c4f5ac4f91fdae71b. It is recommended to apply a patch to fix this issue. **Published:** April 03, 2025; 10:15:46 am -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-3158	A vulnerability, which was classified as critical, has been found in Open Asset Import Library Assimp 5.4.3. Affected by this issue is the function Assimp::LWO::AnimResolver::UpdateAnimRangeSetup of the file code/AssetLib/LWO/LWOAnimation.cpp of the component LWO File Handler. The manipulation leads to heap-based buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. **Published:** April 03, 2025; 10:15:46 am -0400	*V4.0:*(not available) *V3.1:* 5.3 MEDIUM *V2.0:*(not available)
CVE-2025-32050	A flaw was found in libsoup. The libsoup append_param_quoted() function may contain an overflow bug resulting in a buffer under-read. **Published:** April 03, 2025; 10:15:43 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-21997	In the Linux kernel, the following vulnerability has been resolved: xsk: fix an integer overflow in xp_create_and_assign_umem() Since the i and pool->chunk_size variables are of type 'u32', their product can wrap around and then be cast to 'u64'. This can lead to two different XDP buffers pointing to the same memory area. Found by InfoTeCS on behalf of Linux Verification Center (linuxtesting.org) with SVACE. **Published:** April 03, 2025; 4:15:15 am -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)

Figure E.15: 15th 10 of Buffer Overflow

CVE-2025-3148	A vulnerability was found in codeprojects Product Management System 1.0 and classified as problematic. This issue affects some unknown processing of the component Login. The manipulation of the argument Str1 leads to buffer overflow. Attacking locally is a requirement. The exploit has been disclosed to the public and may be used. **Published:** April 03, 2025; 3:15:41 am -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-3139	A vulnerability was found in code-projects Bus Reservation System 1.0 and classified as critical. Affected by this issue is the function Login of the component Login Form. The manipulation of the argument Str1 leads to buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. **Published:** April 03, 2025; 12:15:39 am -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2024-45064	A buffer overflow vulnerability exists in the FileX internal RAM interface functionality of STMicroelectronics X-CUBE-AZRTOS-WL 2.0.0. A specially crafted set of network packets can lead to code execution. An attacker can send a sequence of requests to trigger this vulnerability. **Published:** April 02, 2025; 10:15:43 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30356	CryptoLib provides a software-only solution using the CCSDS Space Data Link Security Protocol - Extended Procedures (SDLS-EP) to secure communications between a spacecraft running the core Flight System (cFS) and a ground station. In 1.3.3 and earlier, a heap buffer overflow vulnerability persists in the Crypto_TC_ApplySecurity function due to an incomplete validation check on the fl (frame length) field. Although CVE-2025-29912 addressed an underflow issue involving fl, the patch fails to fully prevent unsafe calculations. As a result, an attacker can still craft malicious frames that cause a negative tf_payload_len, which is then interpreted as a large unsigned value, leading to a heap buffer overflow in a memcpy call. **Published:** April 01, 2025; 6:15:21 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-29070	A heap buffer overflow vulnerability has been identified in thesmooth2() in cmsgamma.c in lcms2-2.16 which allows a remote attacker to cause a denial of service. NOTE: the Supplier disputes this because "this is not exploitable as this function is never called on normal color management, is there only as a helper for low-level programming and investigation." **Published:** April 01, 2025; 5:15:44 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29069	A heap buffer overflow vulnerability has been identified in the lcms2-2.16. The vulnerability exists in the UnrollChunkyBytes function in cmspack.c, which is responsible for handling color space transformations. NOTE: this is disputed by the Supplier because the finding identified a bug in a third-party calling program, not in lcms. **Published:** April 01, 2025; 4:15:17 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-28398	D-LINK DI-8100 16.07.26A1 is vulnerable to Buffer Overflow in the ipsec_net_asp function via the remot_ip parameter. **Published:** April 01, 2025; 10:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-28395	D-LINK DI-8100 16.07.26A1 is vulnerable to Buffer Overflow in the ipsec_road_asp function via the host_ip parameter. **Published:** April 01, 2025; 10:15:32 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-24266	A buffer overflow was addressed with improved bounds checking. This issue is fixed in macOS Ventura 13.7.5, macOS Sequoia 15.4, macOS Sonoma 14.7.5. An app may be able to cause unexpected system termination. **Published:** March 31, 2025; 7:15:23 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-24237	A buffer overflow was addressed with improved bounds checking. This issue is fixed in visionOS 2.4, macOS Ventura 13.7.5, iOS 18.4 and iPadOS 18.4, iPadOS 17.7.6, macOS Sequoia 15.4, macOS Sonoma 14.7.5. An app may be able to cause unexpected system termination. **Published:** March 31, 2025; 7:15:20 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.16: 16th 10 of Buffer Overflow

CVE-2025-24228	A buffer overflow issue was addressed with improved memory handling. This issue is fixed in macOS Ventura 13.7.5, macOS Sequoia 15.4, macOS Sonoma 14.7.5. An app may be able to execute arbitrary code with kernel privileges. **Published:** March 31, 2025; 7:15:20 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-24209	A buffer overflow issue was addressed with improved memory handling. This issue is fixed in tvOS 18.4, Safari 18.4, iPadOS 17.7.6, iOS 18.4 and iPadOS 18.4, macOS Sequoia 15.4. Processing maliciously crafted web content may lead to an unexpected process crash. **Published:** March 31, 2025; 7:15:18 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-24157	A buffer overflow issue was addressed with improved memory handling. This issue is fixed in macOS Ventura 13.7.5, macOS Sequoia 15.4, macOS Sonoma 14.7.5. An app may be able to cause unexpected system termination or corrupt kernel memory. **Published:** March 31, 2025; 7:15:16 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-54809	Netgear Inc WNR854T 1.5.2 (North America) contains a stack-based buffer overflow vulnerability in the parse_st_header function due to use of a request header parameter in a strncpy where size is determined based on the input specified. By sending a specially crafted packet, an attacker can take control of the program counter and hijack control flow of the program to execute arbitrary system commands. **Published:** March 31, 2025; 5:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-54808	Netgear WNR854T 1.5.2 (North America) contains a stack-based buffer overflow vulnerability in the SetDefaultConnectionService function due to an unconstrained use of sscanf. The vulnerability allows for control of the program counter and can be utilized to achieve arbitrary code execution. **Published:** March 31, 2025; 5:15:48 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-54802	In Netgear WNR854T 1.5.2 (North America), the UPNP service (/usr/sbin/upnp) is vulnerable to stack-based buffer overflow in the M-SEARCH Host header. **Published:** March 31, 2025; 5:15:47 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-3007	A vulnerability was found in Novastar CX40 up to 2.44.0. It has been rated as critical. This issue affects the function getopt of the file /usr/nova/bin/netconfig of the component NetFilter Utility. The manipulation of the argument cmd/netmask/pipeout/nettask leads to stack-based buffer overflow. The exploit has been disclosed to the public and may be used. The vendor was contacted early about this disclosure but did not respond in any way. **Published:** March 31, 2025; 3:15:43 pm -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)
CVE-2023-33302	A buffer copy without checking size of input ('classic buffer overflow') in Fortinet FortiMail webmail and administrative interface version 6.4.0 through 6.4.4 and before 6.2.6 and FortiNDR administrative interface version 7.2.0 and before 7.1.0 allows an authenticated attacker with regular webmail access to trigger a buffer overflow and to possibly execute unauthorized code or commands via specifically crafted HTTP requests. **Published:** March 31, 2025; 11:15:41 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2924	A vulnerability, which was classified as problematic, was found in HDF5 up to 1.14.6. This affects the function H5HL__fl_deserialize of the file src/H5HLcache.c. The manipulation of the argument free_block leads to heap-based buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 4:15:26 pm -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)
CVE-2025-2923	A vulnerability, which was classified as problematic, has been found in HDF5 up to 1.14.6. Affected by this issue is the function H5F_addr_encode_len of the file src/H5Fint.c. The manipulation of the argument pp leads to heap-based buffer overflow. Attacking locally is a requirement. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 3:15:24 pm -0400	*V4.0:*(not available) *V3.1:* 3.3 LOW *V2.0:*(not available)
CVE-2025-2915	A vulnerability classified as problematic was found in HDF5 up to 1.14.6. This vulnerability affects the function H5F__accum_free of the file src/H5Faccum.c. The manipulation of the argument overlap_size leads to heap-based buffer overflow. Attacking locally is a requirement. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 1:15:30 pm -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)

Figure E.17: 17th 11 of Buffer Overflow

CVE-2025-2914	A vulnerability classified as problematic has been found in HDF5 up to 1.14.6. This affects the function H5FS__sinfo_serialize_sct_cb of the file src/H5FScache.c. The manipulation of the argument sect leads to heap-based buffer overflow. Local access is required to approach this attack. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 1:15:30 pm -0400	*V4.0:*(not available) *V3.1:* 3.3 LOW *V2.0:*(not available)
CVE-2025-2912	A vulnerability was found in HDF5 up to 1.14.6. It has been declared as problematic. Affected by this vulnerability is the function H5O_msg_flush of the file src/H5Omessage.c. The manipulation of the argument oh leads to heap-based buffer overflow. The attack needs to be approached locally. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 12:15:30 pm -0400	*V4.0:*(not available) *V3.1:* 3.3 LOW *V2.0:*(not available)
CVE-2025-28221	Tenda W6_S v1.0.0.4_510 has a Buffer Overflow vulnerability in the set_local_time function, which allows remote attackers to cause web server crash via parameter time passed to the binary through a POST request. **Published:** March 28, 2025; 10:15:20 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-28220	Tenda W6_S v1.0.0.4_510 has a Buffer Overflow vulnerability in the setcfm function, which allows remote attackers to cause web server crash via parameter funcpara1 passed to the binary through a POST request. **Published:** March 28, 2025; 10:15:20 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2023-53010	In the Linux kernel, the following vulnerability has been resolved: bnxt: Do not read past the end of test names Test names were being concatenated based on a offset beyond the end of the first name, which tripped the buffer overflow detection logic: detected buffer overflow in strnlen [...] Call Trace: bnxt_ethtool_init.cold+0x18/0x18 Refactor struct hwrm_selftest_qlist_output to use an actual array, and adjust the concatenation to use snprintf() rather than a series of strncat() calls. **Published:** March 27, 2025; 1:15:50 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2022-49754	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: Fix a buffer overflow in mgmt_mesh_add() Smatch Warning: net/bluetooth/mgmt_util.c:375 mgmt_mesh_add() error: __memcpy() 'mesh_tx->param' too small (48 vs 50) Analysis: 'mesh_tx->param' is array of size 48. This is the destination. u8 param[sizeof(struct mgmt_cp_mesh_send) + 29]; // 19 + 29 = 48. But in the caller 'mesh_send' we reject only when len > 50. len > (MGMT_MESH_SEND_SIZE + 31) // 19 + 31 = 50. **Published:** March 27, 2025; 1:15:40 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2025-28135	TOTOLINK A810R V4.1.2cu.5182_B20201026 was found to contain a buffer overflow vulnerability in downloadFile.cgi. **Published:** March 27, 2025; 12:15:30 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2849	A vulnerability, which was classified as problematic, was found in UPX up to 5.0.0. Affected is the function PackLinuxElf64::un_DT_INIT of the file src/p_lx_elf.cpp. The manipulation leads to heap-based buffer overflow. It is possible to launch the attack on the local host. The exploit has been disclosed to the public and may be used. The patch is identified as e0b6ff192412f5bb5364c1948f4f6b27a0cd5ea2. It is recommended to apply a patch to fix this issue. **Published:** March 27, 2025; 10:15:55 am -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)
CVE-2025-2837	Silicon Labs Gecko OS HTTP Request Handling Stack-based Buffer Overflow Remote Code Execution Vulnerability. This vulnerability allows network-adjacent attackers to execute arbitrary code on affected installations of Silicon Labs Gecko OS. Authentication is not required to exploit this vulnerability. The specific flaw exists within the handling of HTTP requests. The issue results from the lack of proper validation of the length of user-supplied data prior to copying it to a stack-based buffer. An attacker can leverage this vulnerability to execute code in the context of the device. Was ZDI-CAN-23245. **Published:** March 26, 2025; 6:15:15 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.18: 18th 9 of Buffer Overflow

CVE-2025-26004	Telesquare TLR-2005KSH 1.1.4 is vulnerable to unauthorized stack buffer overflow vulnerability when requesting admin.cgi parameter with setDdns. **Published:** March 26, 2025; 3:15:27 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27836	An issue was discovered in Artifex Ghostscript before 10.05.0. The BJ10V device has a Print buffer overflow in contrib/japanese/gdev10v.c. **Published:** March 25, 2025; 5:15:43 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27835	An issue was discovered in Artifex Ghostscript before 10.05.0. A buffer overflow occurs when converting glyphs to Unicode in psi/zbfont.c. **Published:** March 25, 2025; 5:15:43 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27834	An issue was discovered in Artifex Ghostscript before 10.05.0. A buffer overflow occurs via an oversized Type 4 function in a PDF document to pdf/pdf_func.c. **Published:** March 25, 2025; 5:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27833	An issue was discovered in Artifex Ghostscript before 10.05.0. A buffer overflow occurs for a long TTF font name to pdf/pdf_fmap.c. **Published:** March 25, 2025; 5:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27832	An issue was discovered in Artifex Ghostscript before 10.05.0. The NPDL device has a Compression buffer overflow for contrib/japanese/gdevnpdl.c. **Published:** March 25, 2025; 5:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27831	An issue was discovered in Artifex Ghostscript before 10.05.0. The DOCXWRITE TXTWRITE device has a text buffer overflow via long characters to devices/vector/doc_common.c. **Published:** March 25, 2025; 5:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-27830	An issue was discovered in Artifex Ghostscript before 10.05.0. A buffer overflow occurs during serialization of DollarBlend in a font, for base/write_t1.c and psi/zfapi.c. **Published:** March 25, 2025; 5:15:42 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-30216	CryptoLib provides a software-only solution using the CCSDS Space Data Link Security Protocol - Extended Procedures (SDLS-EP) to secure communications between a spacecraft running the core Flight System (cFS) and a ground station. In versions 1.3.3 and prior, a Heap Overflow vulnerability occurs in the `Crypto_TM_ProcessSecurity` function (`crypto_tm.c:1735:8`). When processing the Secondary Header Length of a TM protocol packet, if the Secondary Header Length exceeds the packet's total length, a heap overflow is triggered during the memcpy operation that copies packet data into the dynamically allocated buffer `p_new_dec_frame`. This allows an attacker to overwrite adjacent heap memory, potentially leading to arbitrary code execution or system instability. A patch is available at commit 810fd66d592c883125272fef123c3240db2f170f. **Published:** March 25, 2025; 4:15:22 pm -0400	*V4.0:*(not available) *V3.1:* 9.1 CRITICAL *V2.0:*(not available)
CVE-2025-2531	Luxion KeyShot DAE File Parsing Heap-based Buffer Overflow Remote Code Execution Vulnerability. This vulnerability allows remote attackers to execute arbitrary code on affected installations of Luxion KeyShot. User interaction is required to exploit this vulnerability in that the target must visit a malicious page or open a malicious file. The specific flaw exists within the parsing of dae files. The issue results from the lack of proper validation of the length of user-supplied data prior to copying it to a heap-based buffer. An attacker can leverage this vulnerability to execute code in the context of the current process. Was ZDI-CAN-23704. **Published:** March 25, 2025; 11:15:25 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2757	A vulnerability classified as critical was found in Open Asset Import Library Assimp 5.4.3. This vulnerability affects the function AI_MD5_PARSE_STRING_IN_QUOTATION of the file code/AssetLib/MD5/MD5Parser.cpp of the component MD5 File Handler. The manipulation of the argument data leads to heap-based buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. **Published:** March 25, 2025; 6:15:16 am -0400	*V4.0:*(not available) *V3.1:* 6.3 MEDIUM *V2.0:*(not available)

Figure E.19: 19th 11 of Buffer Overflow

CVE-2025-2756	A vulnerability classified as critical has been found in Open Asset Import Library Assimp 5.4.3. This affects the function Assimp::AC3DImporter::ConvertObjectSection of the file code/AssetLib/AC/ACLoader.cpp of the component AC3D File Handler. The manipulation of the argument tmp leads to heap-based buffer overflow. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. **Published:** March 25, 2025; 6:15:16 am -0400	*V4.0:*(not available) *V3.1:* 6.3 MEDIUM *V2.0:*(not available)
CVE-2025-2754	A vulnerability was found in Open Asset Import Library Assimp 5.4.3. It has been declared as critical. Affected by this vulnerability is the function Assimp::AC3DImporter::ConvertObjectSection of the file code/AssetLib/AC/ACLoader.cpp of the component AC3D File Handler. The manipulation of the argument it leads to heap-based buffer overflow. The attack can be launched remotely. The exploit has been disclosed to the public and may be used. **Published:** March 25, 2025; 5:15:17 am -0400	*V4.0:*(not available) *V3.1:* 6.3 MEDIUM *V2.0:*(not available)
CVE-2025-29135	A stack-based buffer overflow vulnerability in Tenda AC7 V15.03.06.44 allows a remote attacker to execute arbitrary code through a stack overflow attack using the security parameter of the formWifiBasicSet function. **Published:** March 24, 2025; 5:15:17 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29100	Tenda AC8 V16.03.34.06 is vulnerable to Buffer Overflow in the fromSetRouteStatic function via the parameter list. **Published:** March 24, 2025; 5:15:17 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2021-26105	A stack-based buffer overflow vulnerability (CWE-121) in the profile parser of FortiSandbox version 3.2.2 and below, version 3.1.4 and below may allow an authenticated attacker to potentially execute unauthorized code or commands via specifically crafted HTTP requests. **Published:** March 24, 2025; 12:15:16 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2621	A vulnerability was found in D-Link DAP-1620 1.03 and classified as critical. This issue affects the function check_dws_cookie of the file /storage. The manipulation of the argument uid leads to stack-based buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. This vulnerability only affects products that are no longer supported by the maintainer. **Published:** March 22, 2025; 1:15:34 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-2620	A vulnerability has been found in D-Link DAP-1620 1.03 and classified as critical. This vulnerability affects the function mod_graph_auth_uri_handler of the file /storage of the component Authentication Handler. The manipulation leads to stack-based buffer overflow. The attack can be initiated remotely. The exploit has been disclosed to the public and may be used. This vulnerability only affects products that are no longer supported by the maintainer. **Published:** March 22, 2025; 11:15:38 am -0400	*V4.0:* 9.3 CRITICAL *V3.1:* 9.8 CRITICAL *V2.0:* 10.0 HIGH
CVE-2025-2619	A vulnerability, which was classified as critical, was found in D-Link DAP-1620 1.03. This affects the function check_dws_cookie of the file /storage of the component Cookie Handler. The manipulation leads to stack-based buffer overflow. It is possible to initiate the attack remotely. The exploit has been disclosed to the public and may be used. This vulnerability only affects products that are no longer supported by the maintainer. **Published:** March 22, 2025; 10:15:16 am -0400	*V4.0:* 9.3 CRITICAL *V3.1:* 9.8 CRITICAL *V2.0:* 10.0 HIGH
CVE-2025-2618	A vulnerability, which was classified as critical, has been found in D-Link DAP-1620 1.03. Affected by this issue is the function set_ws_action of the file /dws/api/ of the component Path Handler. The manipulation leads to heap-based buffer overflow. The attack may be launched remotely. The exploit has been disclosed to the public and may be used. This vulnerability only affects products that are no longer supported by the maintainer. **Published:** March 22, 2025; 10:15:16 am -0400	*V4.0:* 9.3 CRITICAL *V3.1:* 9.8 CRITICAL *V2.0:* 10.0 HIGH

Figure E.20: 20th 9 of Buffer Overflow

CVE-2025-30472	Corosync through 3.1.9, if encryption is disabled or the attacker knows the encryption key, has a stack-based buffer overflow in orf_token_endian_convert in exec/totemsrp.c via a large UDP packet. **Published:** March 21, 2025; 10:15:16 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-2592	A vulnerability, which was classified as critical, has been found in Open Asset Import Library Assimp 5.4.3. This issue affects the function CSMImporter::InternReadFile of the file code/AssetLib/CSM/CSMLoader.cpp. The manipulation leads to heap-based buffer overflow. The attack may be initiated remotely. The exploit has been disclosed to the public and may be used. The patch is named 2690e354da0c681db000cfd892a55226788f2743. It is recommended to apply a patch to fix this issue. **Published:** March 21, 2025; 10:15:17 am -0400	*V4.0:*(not available) *V3.1:* 6.3 MEDIUM *V2.0:*(not available)
CVE-2025-2584	A vulnerability was found in WebAssembly wabt 1.0.36. It has been declared as critical. This vulnerability affects the function BinaryReaderInterp::GetReturnCallDropKeepCount of the file wabt/src/interp/binary-reader-interp.cc. The manipulation leads to heap-based buffer overflow. The attack can be initiated remotely. The complexity of an attack is rather high. The exploitation appears to be difficult. The exploit has been disclosed to the public and may be used. **Published:** March 21, 2025; 4:15:11 am -0400	*V4.0:*(not available) *V3.1:* 7.4 HIGH *V2.0:*(not available)
CVE-2024-13903	A vulnerability was found in quickjs-ng QuickJS up to 0.8.0. It has been declared as problematic. Affected by this vulnerability is the function JS_GetRuntime of the file quickjs.c of the component qjs. The manipulation leads to stack-based buffer overflow. The attack can be launched remotely. Upgrading to version 0.9.0 is able to address this issue. The patch is named 99c02eb45170775a9a679c32b45dd4000ea67aff. It is recommended to upgrade the affected component. **Published:** March 21, 2025; 3:15:34 am -0400	*V4.0:*(not available) *V3.1:* 7.5 HIGH *V2.0:*(not available)
CVE-2025-26336	Dell Chassis Management Controller Firmware for Dell PowerEdge FX2, version(s) prior to 2.40.200.202101130302, and Dell Chassis Management Controller Firmware for Dell PowerEdge VRTX version(s) prior to 3.41.200.202209300499, contain(s) a Stack-based Buffer Overflow vulnerability. An unauthenticated attacker with remote access could potentially exploit this vulnerability, leading to Remote execution. **Published:** March 20, 2025; 11:15:12 pm -0400	*V4.0:*(not available) *V3.1:* 9.8 CRITICAL *V2.0:*(not available)
CVE-2025-29149	Tenda i12 V1.0.0.10(3805) was discovered to contain a buffer overflow via the ping1 parameter in the formSetAutoPing function. **Published:** March 20, 2025; 1:15:38 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-29121	A vulnerability was found in Tenda AC6 V15.03.05.16. The vulnerability affects the functionality of the /goform/fast_setting_wifi_set file form_fast_setting_wifi_set. Using the timeZone parameter causes a stack-based buffer overflow. **Published:** March 20, 2025; 1:15:38 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-57440	D-Link DSL-3788 revA1 1.01R1B036_EU_EN is vulnerable to Buffer Overflow via the COMM_MAKECustomMsg function of the webproc cgi **Published:** March 20, 2025; 1:15:37 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure E.21: 21th 8 of Buffer Overflow

F

Appendix 6 Appearance of Double Free in NVD

Total 5.(20th March 2025 to 20th April 2025)

CVE-2025-22097	In the Linux kernel, the following vulnerability has been resolved: drm/vkms: Fix use after free and double free on init error If the driver initialization fails, the vkms_exit() function might access an uninitialized or freed default_config pointer and it might double free it. Fix both possible errors by initializing default_config only when the driver initialization succeeded. **Published:** April 16, 2025; 11:16:04 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2025-2925	A vulnerability has been found in HDF5 up to 1.14.6 and classified as problematic. This vulnerability affects the function H5MM_realloc of the file src/H5MM.c. The manipulation of the argument mem leads to double free. The attack needs to be approached locally. The exploit has been disclosed to the public and may be used. **Published:** March 28, 2025; 4:15:26 pm -0400	*V4.0:*(not available) *V3.1:* 5.5 MEDIUM *V2.0:*(not available)
CVE-2025-2027	A double free vulnerability has been identified in the ASUS System Analysis service. This vulnerability can be triggered by sending specially crafted local RPC requests, leading to the service crash and potentially memory manipulation in some rare circumstances. Refer to the 'Security Update for MyASUS' section on the ASUS Security Advisory for more information. **Published:** March 28, 2025; 2:15:33 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2023-52930	In the Linux kernel, the following vulnerability has been resolved: drm/i915: Fix potential bit_17 double-free A userspace with multiple threads racing I915_GEM_SET_TILING to set the tiling to I915_TILING_NONE could trigger a double free of the bit_17 bitmask. (Or conversely leak memory on the transition to tiled.) Move allocation/free'ing of the bitmask within the section protected by the obj lock. [tursulin: Correct fixes tag and added cc stable.] (cherry picked from commit 10e0cbaaf1104f449d695c80bcacf930dcd3c42e) **Published:** March 27, 2025; 1:15:42 pm -0400	*V4.0:*(not available) *V3.1:* 7.8 HIGH *V2.0:*(not available)
CVE-2022-49753	In the Linux kernel, the following vulnerability has been resolved: dmaengine: Fix double increment of client_count in dma_chan_get() The first time dma_chan_get() is called for a channel the channel client_count is incorrectly incremented twice for public channels, first in balance_ref_count(), and again prior to returning. This results in an incorrect client count which will lead to the channel resources not being freed when they should be. A simple test of repeated module load and unload of async_tx on a Dell Power Edge R7425 also shows this resulting in a kref underflow warning. [124.329662] async_tx: api initialized (async) [129.000627] async_tx: api initialized (async) [130.047839] ------------[cut here]------------ [130.052472] refcount_t: underflow; use-after-free. [130.057279] WARNING: CPU: 3 PID: 19364 at lib/refcount.c:28 refcount_warn_saturate+0xba/0x110 [130.065811] Modules linked in: async_tx(-) rfkill intel_rapl_msr intel_rapl_common amd64_edac edac_mce_amd ipmi_ssif kvm_amd dcdbas kvm mgag200 drm_shmem_helper acpi_ipmi irqbypass drm_kms_helper ipmi_si syscopyarea sysfillrect rapl pcspkr ipmi_devintf sysimgblt fb_sys_fops k10temp i2c_piix4 ipmi_msghandler acpi_power_meter acpi_cpufreq vfat fat drm fuse xfs libcrc32c sd_mod t10_pi sg ahci crct10dif_pclmul libahci crc32_pclmul crc32c_intel ghash_clmulni_intel igb megaraid_sas i40e libata i2c_algo_bit ccp sp5100_tco dca dm_mirror dm_region_hash dm_log dm_mod [last unloaded: async_tx] [130.117361] CPU: 3 PID: 19364 Comm: modprobe Kdump: loaded Not tainted 5.14.0-185.el9.x86_64 #1 [130.126091] Hardware name: Dell Inc. PowerEdge R7425/02MJ3T, BIOS 1.18.0 01/17/2022 [130.133806] RIP: 0010:refcount_warn_saturate+0xba/0x110 [130.139041] Code: 01 01 e8 6d bd 55 00 0f 0b e9 72 9d 8a 00 80 3d 26 18 9c 01 00 75 85 48 c7 c7 f8 a3 03 9d c6 05 16 18 9c 01 01 e8 4a bd 55 00 <0f> 0b e9 4f 9d 8a 00 80 3d 01 18 9c 01 00 0f 85 5e ff ff ff 48 c7 [130.157807] RSP: 0018:ffffbf98898afe68 EFLAGS: 00010286 [130.163036] RAX: 0000000000000000 RBX: ffff9da06028e598 RCX: 0000000000000000 [130.170172] RDX: ffff9daf9de26480 RSI: ffff9daf9de198a0 RDI: ffff9daf9de198a0 [130.177316] RBP: ffff9da7cddf3970 R08: 0000000000000000 R09: 0000000ffff7fff [130.184459] R10: ffffbf98898afd00 R11: ffffffff9d9e8c28 R12: ffff9da7cddf1970 [130.191596] R13: 0000000000000000 R14: 0000000000000000 R15: 0000000000000000 [130.198739] FS: 00007f646435c740(0000) GS:ffff9daf9de00000(0000) knlGS:0000000000000000 [130.206832] CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 [130.212586] CR2: 00007f6463b214f0 CR3: 00000008ab98c000 CR4: 00000000003506e0 [130.219729] Call Trace: [130.222192] <TASK> [130.224305] dma_chan_put+0x10d/0x110 [130.227988] dmaengine_put+0x7a/0xa0 [130.231575] __do_sys_delete_module.constprop.0+0x178/0x280 [130.237157] ? syscall_trace_enter.constprop.0+0x145/0x1d0 [130.242652] do_syscall_64+0x5c/0x90 [130.246240] ? exc_page_fault+0x62/0x150 [130.250178] entry_SYSCALL_64_after_hwframe+0x63/0xcd [130.255243] RIP: 0033:0x7f6463a3f5ab [130.258830] Code: 73 01 c3 48 8b 0d 75 a8 1b 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84 00 00 00 00 00 90 f3 0f 1e fa b8 b0 00 00 00 0f 05 <48> 3d 01 f0 ff ff 73 01 c3 48 8b 0d 45 a8 1b 00 f7 d8 64 89 01 48 [130.277591] RSP: 002b:00007fff22f972c8 EFLAGS: 00000206 ORIG_RAX: 00000000000000b0 [130.285164] RAX: ffffffffffffffda RBX: 000055b6786edd40 RCX: 00007f6463a3f5ab [130.292303] RDX: 0000000000000000 RSI: 0000000000000800 RDI: 000055b6786edda8 [130.299443] RBP: 000055b6786edd40 R08: 0000000000000000 R09: 0000000000000000 [130.306584] R10: 00007f6463b9eac0 R11: 0000000000000206 R12: 000055b6786edda8 [130.313731] R13: 0000000000000000 R14: 000055b6786edda8 R15: 00007fff22f995f8 [130.320875] </TASK> [130.323081] ---[end trace eff7156d56b5cf25]--- cat /sys/class/dma/dma0chan*/in_use would get the wrong result. 2 2 2 Test-by: Jie Hai <haijie1@huawei.com> **Published:** March 27, 2025; 1:15:40 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure F.1: First 5 of Double Free

G

Appendix 7 Appearance of HardCoded Credentials in NVD

(20th March 2025 to 20th April 2025)

Totally 3.

CVE-2025-28230	Incorrect access control in JMBroadcast JMB0150 Firmware v1.0 allows attackers to access hardcoded administrator credentials. **Published:** April 18, 2025; 11:15:58 am -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)
CVE-2024-41794	A vulnerability has been identified in SENTRON 7KT PAC1260 Data Manager (All versions). Affected devices contain hardcoded credentials for remote access to the device operating system with root privileges. This could allow unauthenticated remote attackers to gain full access to a device, if they are in possession of these credentials and if the ssh service is enabled (e.g., by exploitation of CVE-2024-41793). **Published:** April 08, 2025; 5:15:20 am -0400	*V4.0:*(not available) *V3.1:* 10.0 CRITICAL *V2.0:*(not available)
CVE-2025-3426	We observed that Intellispace Portal binaries doesn't have any protection mechanisms to prevent reverse engineering. Specifically, the app's code is not obfuscated, and no measures are in place to protect against decompilation, disassembly, or debugging. As a result, attackers can reverse-engineer the application to gain insights into its internal workings, which can potentially lead to the discovery of sensitive information, business logic flaws, and other vulnerabilities. Utilizing this flaw, the attacker was able to identify the Hardcoded credentials from PortalUsersDatabase.dll, which contains .NET remoting definition. Inside the namespace PortalUsersDatabase, the class Users contains the functions CreateAdmin and CreateService that are used to initialize accounts in the Portal service. Both CreateAdmin and CreateService functions contain a hardcoded encrypted password along with its respective salt that are set with the function SetInitialPasswordAndSalt. This issue affects IntelliSpace Portal: 12 and prior; Advanced Visualization Workspace: 15. **Published:** April 07, 2025; 1:15:40 pm -0400	*V4.0:*(not available) *V3.x:*(not available) *V2.0:*(not available)

Figure G.1: First 3 of HardCoded Credentials

H

Appendix 8 Calculation Program and source code

This simple program is write in C++, variable values should based on real metric setting.

```cpp
#include<iostream>
#include<vector>
#include<algorithm>
using namespace std;

int main(void)
{
    double con = 16.3407;
    double av = 0.88;  // Local 0.55, Online 0.88
    double pr = 0.85;  // None 0.85, Low 0.62, High 0.27
    double ui = 0.85;  // None 0.85, Required 0.62

    double s = 0.22;   // E 0.56, UE 0.22
    double c = 0.56;
    double i = 0;      // CIA: HIGH 0.56, Low 0.22, None 0
    double a = 0;

    double rl = 0.95;  // OF 0.95
    double f = 1.15;   // High 1.2, Medium 1.15
    double p = 1.15;   // High 1.2, Medium 1.15, Low 1.1, VL 1.05

    double final_score = con * (1 - (1 - c) * (1 - i) * (1 - a) *
        (1 - s)) * av * pr * ui * rl * f * p;
    cout << "final_score: " << final_score << endl;
}
```

Listing H.1: Code for Calculating Final Vulnerability Score