future internet

MDPI

Article

Assessing Browser Security: A Detailed Study Based on CVE Metrics

Oleksii Chalyi *, Kęstutis Driaunys and Vytautas Rudžionis

Institute of Social Sciences and Applied Informatics, Vilnius University, Muitines St 8, LT-44280 Kaunas, Lithuania; kestutis.driaunys@knf.vu.lt (K.D.); vytautas.rudzionis@knf.vu.lt (V.R.)
* Correspondence: oleksii.chalyi@knf.vu.lt

Abstract: This study systematically evaluates the vulnerabilities of modern web browsers using developed indices derived from the CVE database, including I_{CVE}, I_{CVSS}, I_R and I_T. These indices incorporate metrics such as vulnerability severity and risks, along with browser popularity, to enable a balanced comparison of browser security. The results highlight significant differences in browser security: while Google Chrome and Samsung Internet exhibited lower threat indices, Mozilla Firefox demonstrated consistently higher scores, indicating greater exposure to risks. These observations a slightly contradict widespread opinion. The findings emphasize the importance of timely software updates in mitigating vulnerabilities, as many incidents were linked to outdated browser versions. This study also introduces a robust methodology for assessing browser threats, providing a framework for future research. Potential applications include developing browser-based penetration testing systems to simulate phishing and data extraction scenarios, offering insights into user-specific risks and broader organizational impacts. By combining theoretical analysis with practical implications, this work contributes to advancing browser security and lays the foundation for future applied research in cybersecurity.

Keywords: CVE; browser threats; information security; CVSS; EPSS; risk; vulnerabilities

Academic Editors: Weizhi Meng and Christian D. Jensen

Received: 5 February 2025
Revised: 17 February 2025
Accepted: 22 February 2025
Published: 25 February 2025

Citation: Chalyi, O.; Driaunys, K.; Rudžionis, V. Assessing Browser Security: A Detailed Study Based on CVE Metrics. *Future Internet* **2025**, *17*, 104. https://doi.org/10.3390/fi17030104

1. Introduction

Web browsers, used by more than 5.35 billion internet users [1], are essential tools for accessing information through websites and facilitating communication on social media platforms [2]. They play a crucial role in modern digital life, enabling activities such as online banking, shopping, and accessing cloud-based services [3]. However, their extensive use and the sensitive information they handle make them prime targets for cybercriminals [4,5].

Over the past decade, browser-related security incidents have shown a concerning upward trend in both frequency and severity [6]. For instance, in 2021, Google Chrome faced CVE-2021-4102 [7], a critical zero-day vulnerability actively exploited in the wild, prompting emergency updates. In 2024, the type confusion vulnerability CVE-2024-1238 [8] once again forced Google to patch billions of devices within days to prevent widespread exploitation. Similar threats have impacted other major browsers. For example, Mozilla Firefox experienced CVE-2022-26485 [9], a use-after-free vulnerability enabling remote code execution, which was also exploited before its discovery. In 2024, Mozilla patched critical zero-day vulnerability CVE-2024-9680 [10] in its Firefox browser, which was actively exploited in the wild. These incidents underscore how rapidly evolving attack techniques, particularly zero-day exploits and targeted malware injections, continue to compromise sensitive data [11].

Understanding these risks is crucial, as web browsers are integral to both personal and professional online activities [12]. With billions of users relying on these browsers daily, even minor vulnerabilities can result in widespread consequences [13]. This study contributes to the field by providing a comprehensive analysis of browser-related threats using the security indices CVE, CVSS and EPSS [14] and propose four new, developed for this comparison, I_{CVE}, I_{CVSS}, I_R and I_T. Unlike previous research, which often focuses on isolated vulnerabilities or specific browsers, this work offers a holistic view of the interconnected nature of browser vulnerabilities across multiple platforms. By investigating these threats in depth, this research lays the groundwork for future development of browser-based penetration testing systems. Such systems could assess user-specific risks [15] by identifying exploitable vulnerabilities, ultimately enabling broader organizational risk analysis and proactive mitigation strategies. This study bridges the gap between theoretical vulnerability assessments and practical, user-centric security solutions, emphasizing the need for scalable frameworks to address evolving cyber threats. This work stands out by providing actionable insights for both individual users and organizations, while also setting the stage for future innovations in browser security.

2. Related Works

Tewari and Datt, in their article, investigated the security vulnerabilities of popular web browsers [16]. Firstly, they defined the common vulnerabilities in the most-used web browsers, such as SQL injection, cross-site scripting (XSS), sensitive file disclosure, cross-site request forgery (CSRF), and inadequate transport layer protection. They chose six web browsers for testing, including Google Chrome, Safari, Mozilla Firefox, Microsoft Edge, Opera, and Internet Explorer. On these six web browsers, they performed three different tests: the ACID3 test, which checks how well an internet browser adheres to certain web standards, particularly those related to the Document Object Model and JavaScript; a browser speed test; and a browser security test using an online tool and performing around 400 tests. In conclusion, they stated that Mozilla Firefox is the safest of all, passing a total of 370 tests. Safari and Internet Explorer showed no CSS or JavaScript vulnerabilities as identified by the ACID3 test. According to the speed test, Google Chrome is the fastest browser of all.

Petkova, in her article, conducted a security analysis of the most-used browsers that provide access to cyberspace [17]. She analyzed three web browsers—Google Chrome, Safari, and Mozilla Firefox—as the most popular. As a source for analysis, she used the CVE and CVSS databases for the period 1989–2021. According to her results, Google Chrome has the highest percentage of moderate 6–7 and 4–5 CVSS vulnerabilities, Firefox has the highest percentage of critical 9+ CVSS vulnerabilities, and Safari has the highest percentage of moderate CVSS vulnerabilities. In conclusion, she found that all browsers are not fully secure, and due to constant technological innovations, the better browser is the one that can consistently provide updates to enhance security.

Krolo et al. investigated the architecture of modern web browsers to determine their security vulnerabilities [18]. Going deeper into browser architectures, they divided them into monolithic and modular browser architectures. In comparison to monolithic architecture, modular architecture is superior in terms of user experience, fault tolerance, accountability, security, memory management, and performance. In modular architecture, the performance of each program can be easily monitored, making it superior in accountability. In their study, they investigated the architecture of four browsers—Google Chrome, OP, Tahoma, and Gazelle—which are based on modular browser architecture. The analysis showed that the modular architecture of Chrome mitigates the most serious threats related to system compromise and data theft. However, Chrome's architecture does not provide

full protection. Threats related to cross-site attacks, session hijacking, and user interface compromise are not fully mitigated. They also reviewed similar architectures implemented in the OP, Tahoma, and Gazelle web browsers. These architectures sacrifice compatibility with the current web architecture to provide a higher level of security than Chrome. The research was conducted in 2010, and since that time, only Chrome still exists, despite its lower security compared to the other three browsers in their study.

Woo et al., in their paper, presented a quantitative characterization of browser threats that can be used to project the number of vulnerabilities and plan, test, and develop resources more efficiently [19]. For their experiments, they chose three major browsers: Internet Explorer, Firefox, and Mozilla. They classified the vulnerabilities into eight categories, including Input Validation Error, Access Validation Error, Exceptional Condition Error, Environmental Error, Configuration Error, Race Condition Error, Design Error, and Others. They determined the percentage of each vulnerability category relative to the total browser threats. For each category, they calculated the chi-square goodness of fit. Additionally, they investigated vulnerability severity levels (high, medium, and low) and also calculated the chi-square goodness of fit for each level. In conclusion, they state that the fit is significant when aggregate vulnerabilities are divided into classes, provided there are sufficient vulnerabilities in a class. According to their results, Mozilla was the most secure browser with the fewest vulnerabilities, although it was relatively unpopular during the research period.

Fajar and Yazid, in their research, provided a descriptive analysis of the weaknesses and vulnerabilities of the Chrome browser compared to other popular browsers such as Safari and Firefox using CVE data [20]. They found that Chrome has the most reported vulnerabilities but responds the fastest to updates. Firefox showed the highest average severity scores, while Chrome's sandboxing architecture proved more secure. Key issues across browsers include input validation and code injection vulnerabilities, with Chrome managing memory and resources more effectively. The study highlighted Chrome's technical superiority, frequent updates, and market dominance, suggesting that competitors adopt strategies like Chromium-based technologies to stay relevant. Despite frequent reports of vulnerabilities, Chrome's robust design ensures its continued leadership.

Cyber threats are evolving at a rapid pace, with new challenges emerging as older ones diminish in significance. Similarly, browsers are continuously adapting and improving to keep up with these changes. Although conclusions from five or more years ago may still provide useful insights, they often require reassessment in light of current developments. This study aims to evaluate the modern resilience of browsers against potential attacks while introducing additional parameters to enhance the robustness of the evaluation.

3. Materials and Methods

The methodology used in this study focuses on analyzing browser threats through developed in this article indices such as I_{CVE}, I_{CVSS}, I_R and I_T, which incorporate metrics like browser popularity and vulnerability severity. While these indices are primarily used to evaluate browser security, they also provide a foundation for future research into browser-based penetration testing systems.

In particular, the calculated indices could inform the design of scenarios targeting user-specific vulnerabilities, such as phishing simulations or data extraction tests. This opens pathways for further exploration of how user-level browser risks can translate into broader organizational vulnerabilities, a direction that may be pursued in subsequent studies.

3.1. Datasets

In this research, the browsers were selected based on their worldwide popularity as of November 2024, using data from Statcounter GlobalStats [21]. According to this database, the six most popular browsers are as follows:

1. Google Chrome—67.48%;
2. Apple Safari—18.22%;
3. Microsoft Edge—4.84%;
4. Mozilla Firefox—2.6%;
5. Samsung Internet—2.18%;
6. Opera—2%.

In this research, only the first five browsers were analyzed. The CVE Details database was chosen as the source for browser threats [22]. According to this database, the latest records for Opera browser threats end in 2019, which is why it was excluded from the analysis. Additionally, since Samsung Internet was released in 2021, the analysis was conducted starting from that year. For the CVSS score analysis, data from 1 January 2021, to 1 December 2024, were used. The number of vulnerabilities for each browser was determined through a manual calculation based on vulnerabilities categorized by type and impact per year. This approach was necessary because the total number of vulnerabilities could differ from the manual sum by type and impact. The Python Pyplot Matplotlib 3.10.0 API was used for building graphics and diagrams.

3.2. Estimating the Browser Threat Indices

To assess browser security based on CVE metrics, it is essential to define the key parameters used in this evaluation. The following parameters were selected:

1. Browser Popularity–Represents the percentage of users relying on a specific browser. More widely used browsers may attract more attackers due to their larger potential attack surface.
2. Number of CVE Vulnerabilities—Indicates the total number of documented security vulnerabilities associated with the browser, reflecting its historical security posture.
3. CVSS Score—Measures the severity of vulnerabilities, providing insight into how critical and exploitable these weaknesses are.
4. EPSS Score—Estimates the likelihood of a vulnerability being exploited in real-world attacks, offering a predictive risk assessment.

To compare browsers according to their vulnerabilities and CVSS data, a direct comparison may not be precise due to the differing popularity of web browsers. If a browser is more popular, then more users utilize it, and as a result, more users can fall victim to cyberattacks [23]. Looking at the data without an accurate comparison may lead to a false conclusion that Google Chrome is the most vulnerable browser because it has the largest number of vulnerabilities. To determine the correct conclusion about browser security, it is important not only to include the number of vulnerabilities and CVSS scores but also to factor in the popularity of browsers in the comparison. The idea is based on estimating a vulnerability index—the ratio between the number of vulnerabilities and the browser's population, which could be expressed as Formula (1):

$$I_{CVE} = \sum_{t=f}^{l} \frac{V_t}{P_t} \quad (1)$$

where:

- I_{CVE}—CVE browser threat index for certain year;

- t—year;
- f—first year;
- l—last year;
- V_t—the amount of browser vulnerabilities per year;
- P_t—average browser's popularity per year in percentage.

By summing up the ratio between the number of vulnerabilities and the average popularity for each year, the vulnerability index could be calculated for each year.

The CVE Details database can already present the average CVSS score for a selected period. However, to compare browsers via CVSS more precisely, the score should be divided by the average browser popularity by year, as indicated in developed Formula (2):

$$I_{CVSS} = \frac{C_{avg}}{\left(\sum_{t=f}^{l} P_t\right)/(l-t)} \tag{2}$$

where:

- I_{CVSS}—CVSS browser threat index;
- t—year;
- f—first year;
- l—last year;
- C_{avg}—average browser's CVSS score for the selected period;
- P_t—average browser's popularity per year in percentage.

Unlike the previous index, I_{CVE}, the average popularity per year was chosen to maintain scalability due to the available average CVSS score, which is represented as an average score.

The browser threats, which are categorized by type and impact, differ and have varying CVSS and EPSS scores. Since CVSS indicates the severity of a vulnerability and EPSS reflects the probability of its exploitation, the multiplication of CVSS and EPSS can be used to identify risk [24], as represented in developed Formula (3):

$$R_t = \sum_{t=f}^{l} \sum_{i=1}^{n_t} (C_{t,i} \times E_{t,i}) \tag{3}$$

where:

- R_t—risk of all browser threats for a selected year based on CVSS and EPSS scores;
- t—year;
- f—first year;
- l—last year;
- $E_{t,i}$—the amount of browser's vulnerabilities per year and type, impact;
- $C_{t,i}$—CVSS score for selected vulnerability per year according to it type and impact.

To estimate browser threats more accurately, the combination of the number of vulnerabilities, EPSS, and the CVSS score for each should be considered in relation to the browser's popularity, as was previously carried out for the I_{CVE} and I_{CVSS} indices. Proposed Formula (4) presents the calculation steps required for estimating the risk browser threat index:

$$I_R = \sum_{t=f}^{l} \frac{R_t}{P_t} \tag{4}$$

where:

- I_R—risk browser threat index;
- t—year;

- f—first year;
- l—last year;
- R_t—risk of all browser threats for a selected year based on CVSS and EPSS scores;
- P_t—average popularity per year in percentage.

Knowing the scores of the described browser indices, the sum of all indices, which represents the total browser threat index, can be calculated. Proposed Formula (5) illustrates this calculation:

$$I_T = I_{CVE} + I_{CVSS} + I_R \quad (5)$$

where:

- I_T—total browser threat index;
- I_{CVE}—CVE browser threat index for a certain year;
- I_{CVSS}—CVSS browser threat index;
- I_R—risk browser threat index.

In this methodology, the threat index, risk, CVE, and CVSS indices are introduced to accurately assess browser security by considering both vulnerabilities and popularity. The CVE index is calculated by dividing the number of vulnerabilities by the browser's average popularity, while the CVSS index adjusts the CVSS score accordingly. By combining these indices, a more precise comparison of browser security is achieved, taking into account both the number of vulnerabilities and their severity relative to browser usage. That is why, in this research, Formulas (1)–(5) were created to estimate and compare different browsers according to their key parameters to make the comparison more detailed and mathematically justified. This approach ensures that comparisons are not influenced solely by a browser's popularity.

4. Results

4.1. Comparison of Browser Threat Indices

Five browsers were chosen for comparison: Google Chrome, Apple Safari, Microsoft Edge, Mozilla Firefox, and Samsung Internet. Because Samsung Internet was released in 2021, the period for comparison starts from 2021. According to the formulas, the browser popularity value should be calculated. For each year in the period 2021–2024, the average popularity of each browser in the comparison was determined. Figure 1 demonstrates the visualization of the results.

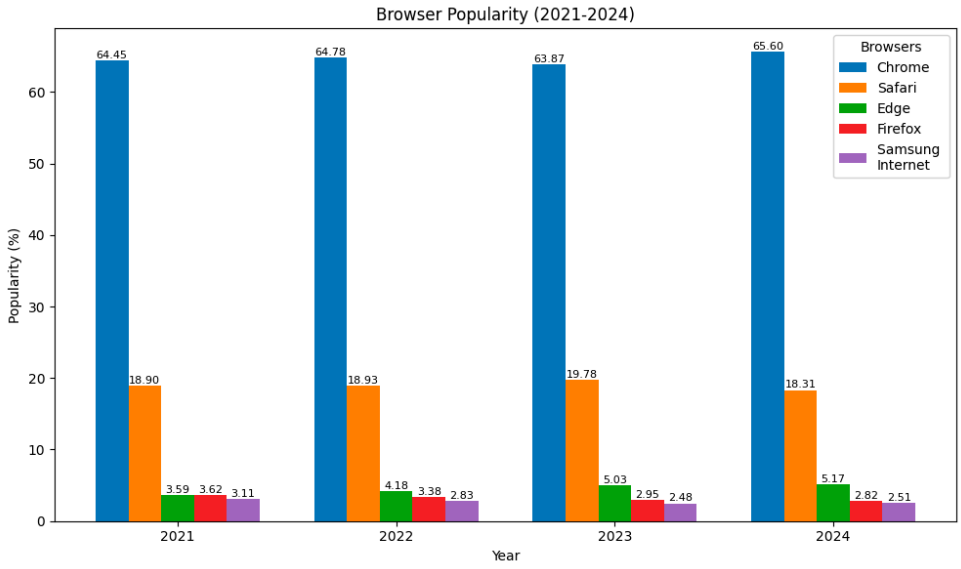

Figure 1. Average browser popularity in the 2021–2024 period.

Instead of finding the average popularity for the whole period 2021–2024, the average popularity for each month was used in calculating all indexes except I_{CVSS} to achieve higher precision. Figure 2 demonstrates the number of vulnerabilities for each browser in the selected period.

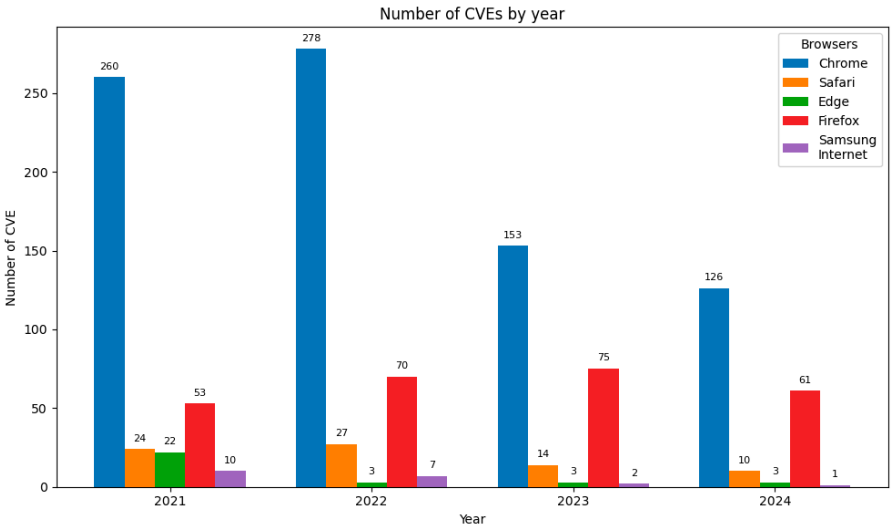

Figure 2. Number of browser threats in the 2021–2024 period.

Knowing the number of vulnerabilities for each year and the browser's average popularity for the selected period, Formula (1) can be utilized to find the I_{CVE} score. Figure 3 demonstrates the visualization of the browser's I_{CVE} score for the selected period and the year 2024.

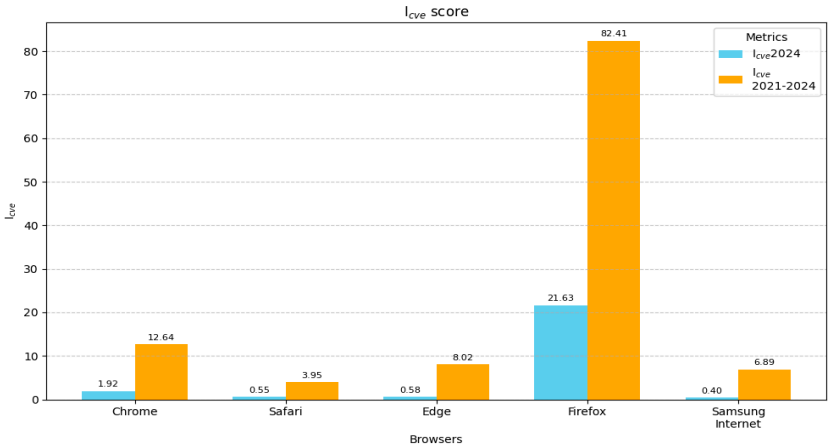

Figure 3. I_{CVE} score for the year 2024 and the 2021–2024 period.

The average CVSS score for 2024 and the period 2021–2024 was calculated. This score is required for calculating the I_{CVSS} score, as indicated in Formula (2). Figure 4 shows the visualization of the average CVSS score for different browsers for the selected period.

The next step is to find the I_{CVSS} browser score. Since the average CVSS score can be found in CVE Details for the selected period, Formula (2) uses the average browser popularity over the 4 years for the calculation. The I_{CVSS} score for the year 2024 was also calculated using the average browser popularity for 2024. Figure 5 demonstrates the visualization of the browsers' calculated I_{CVSS} scores.

The EPSS value indicates the probability that a certain vulnerability could be exploited. To estimate the final threat index score, the multiplication of the CVSS and EPSS scores,

which shows the risk, should be calculated using Formula (3). Figure 6 demonstrates the visualization of the risk value for each year.

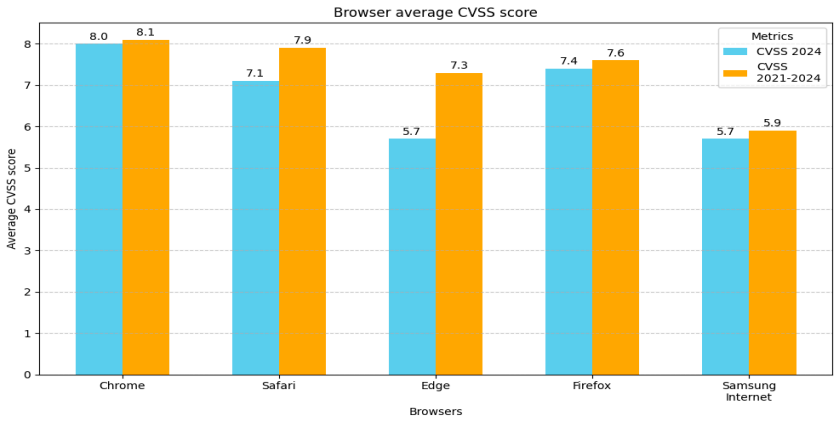

Figure 4. Browser average CVSS score.

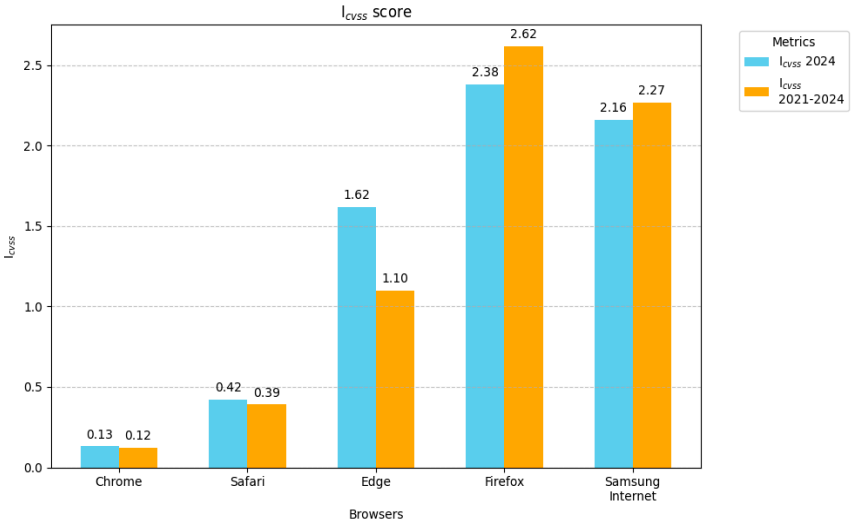

Figure 5. I_{CVSS} score for the 2021–2024 period.

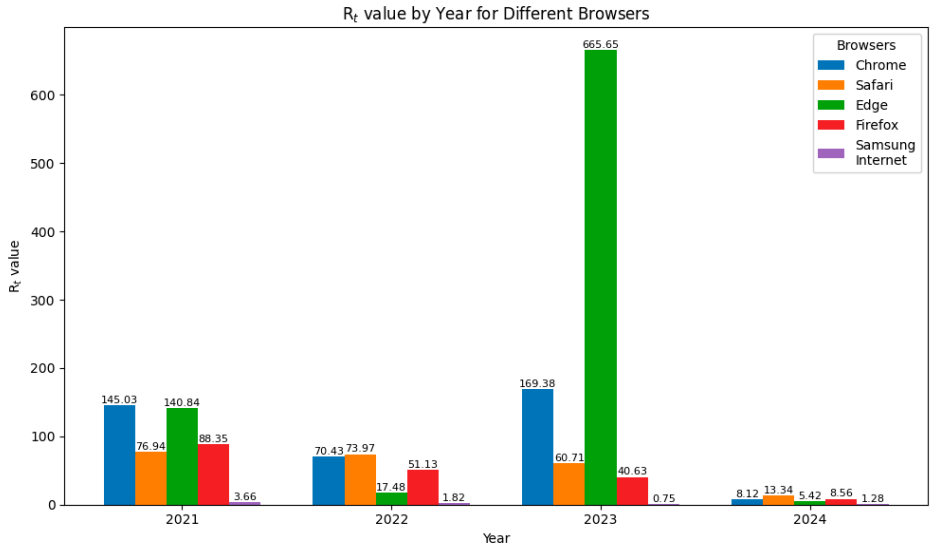

Figure 6. R_t value for the 2021–2024 period.

The final step is to estimate the I_R index using Formula (4). This index includes all the most important criteria for browser threat comparison, including their number of CVEs, average yearly popularity, CVSS, and EPSS scores. The index value for 2024 was also calculated and visualized in Figure 7. Table 1 summarizes all the index scores for 2024 and the 2021–2024 period.

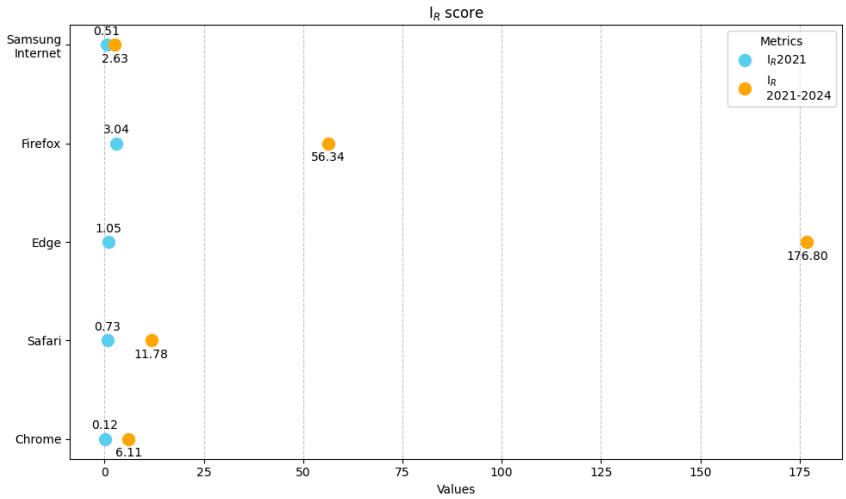

Figure 7. I_R score for the 2021–2024 period.

Table 1. Browser threat index scores for the selected period.

I_R 2024	I_{CVSS} 2024	I_{CVE} 2024	I_R 2021–2024	I_{CVSS} 2021–2024	I_{CVE} 2021–2024	Browser
0.12	0.12	1.92	6.11	0.13	12.64	Chrome
0.73	0.39	0.55	11.78	0.42	3.95	Safari
1.05	1.1	0.58	176.8	1.62	8.02	Edge
3.04	2.62	21.63	56.34	2.38	82.41	Firefox
0.51	2.27	0.4	2.63	2.16	6.89	Samsung

Figure 8 visualizes results from Table 1 as a heatmap. It also highlights the minimum and maximum values for each metric. The highest value is highlighted in a red square, while the lowest is highlighted in green.

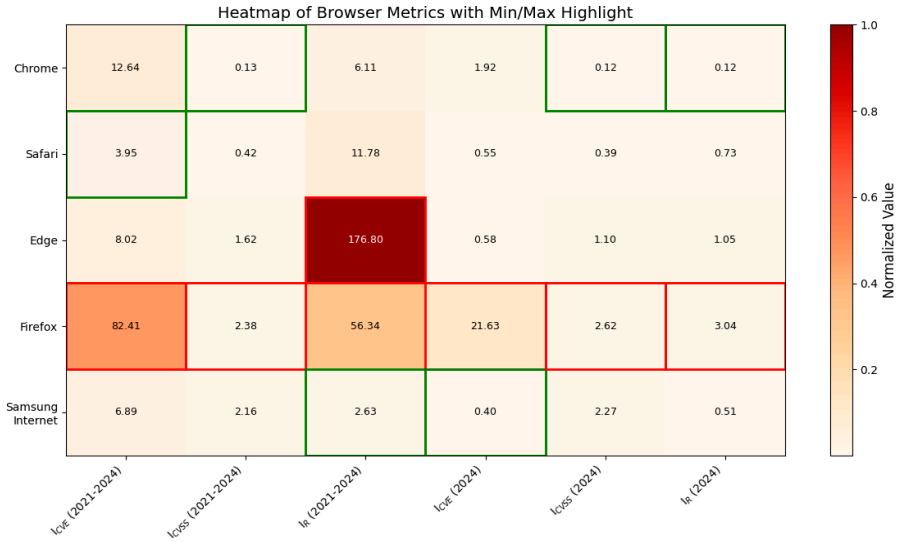

Figure 8. Browser threat index scores for the 2024 and 2021–2024 period.

Knowing all the indices, the sum of them, I_T, which represents the total browser threat index, can be calculated using Formula (5). Figure 9 shows the visualization of the I_T index for the selected period and the year 2024.

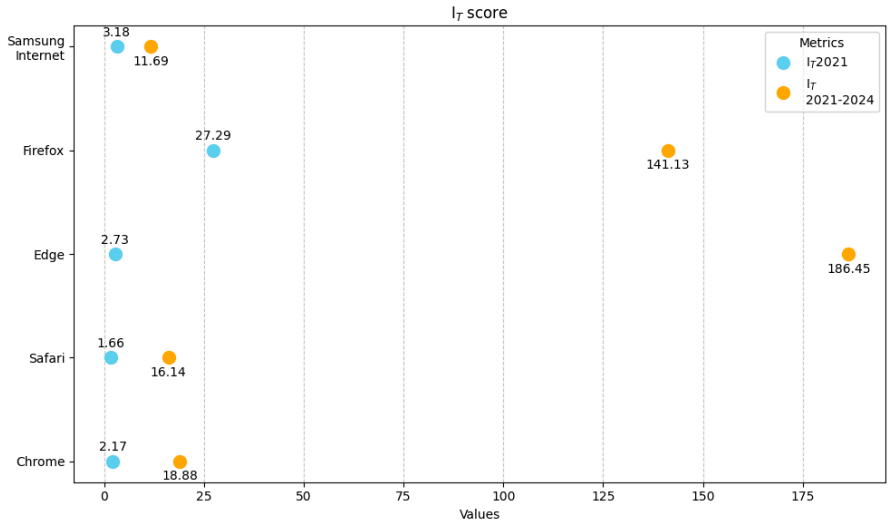

Figure 9. Total browser threat index scores for the 2024 and 2021–2024 period.

4.2. Main Findings

According to the results for the period 2021–2024, Google Chrome is the most popular browser in this comparison, while Samsung Internet is the least popular. Analyzing the CVE Details database showed that over four years, Google Chrome also had the highest number of vulnerabilities (817), followed by Mozilla Firefox with 259 vulnerabilities. Samsung Internet had the lowest number of vulnerabilities during this period, with only 20. This could lead to an early conclusion that Google Chrome is the most vulnerable browser, while Samsung Internet is the safest due to its lower number of vulnerabilities. However, browser popularity should be noted, as it influences the number of vulnerabilities.

The I_{CVE} index score shows the ratio between the number of vulnerabilities and browser popularity. Figure 3 indicates that during the 2021–2024 period, the I_{CVE} index scores for these browsers are as follows: Firefox—82.41, Chrome—12.64, Edge—8.02, Samsung Internet—6.89, and Safari—3.95.

The average CVSS score for the selected period was calculated, and all browsers have a high average CVSS score except Samsung Internet, which has a medium CVSS score. However, the I_{CVSS} index score was calculated to account for browser popularity and determine the ratio between the CVSS score and browser popularity over four years. Figure 5 shows that during the 2021–2024 period, the I_{CVSS} index scores for these browsers are as follows: Firefox—2.62, Samsung Internet—2.27, Edge—1.1, Safari—0.39, and Chrome—0.12.

Each vulnerability has a probability of being exploited, which is represented by the EPSS score. The multiplication of the CVSS score and EPSS score called Risk was calculated, and over four years, Edge had the highest R_t score, while Samsung Internet had the lowest due to its lower number of vulnerabilities. To refine the results and determine the ratio between the R_t score and browser popularity, the I_R index was calculated. Figure 7 indicates that during the 2021–2024 period, the I_R index scores for these browsers are as follows: Edge—176.8, Firefox—56.43, Safari—11.78, Chrome—6.11, and Samsung Internet—2.63.

According to the results for the year 2024, Google Chrome is still the most popular browser in this comparison, while Samsung Internet is the least popular. Analyzing the CVE Details database showed that in 2024, Google Chrome also had the highest number of

vulnerabilities (126), followed by Mozilla Firefox with 61 vulnerabilities. Samsung Internet had the lowest number of vulnerabilities for 2024, with only 1.

Figure 3 shows that for 2024, the I_{CVE} index scores for these browsers are as follows: Firefox—21.63, Chrome—1.91, Edge—0.58, Safari—0.55, and Samsung Internet—0.4.

The average CVSS score for 2024 was calculated, and all browsers have a high average CVSS score except for Samsung Internet and Edge, which have medium CVSS scores. Figure 5 shows that during 2024, the I_{CVSS} index scores for these browsers are as follows: Firefox—2.38, Samsung Internet—2.16, Edge—1.62, Safari—0.42, and Chrome—0.13.

The multiplication of the CVSS score and EPSS score was calculated for 2024, with Safari having the highest R_t score and Samsung Internet the lowest. Figure 7 shows that in 2024, the I_R index scores for these browsers are as follows: Firefox—3.04, Edge—1.05, Safari—0.73, Samsung Internet—0.51, and Chrome—0.12.

Figure 8 demonstrates the overall comparison of browser threats. The higher the index, the less secure the browser is. Mozilla Firefox has five out of six possible highest vulnerability index scores for 2024 and the 2021–2024 period. Microsoft Edge has the highest index score for the 2021–2024 period in the I_T index. Google Chrome has the three lowest index scores out of six, while Samsung Internet has two of them, and Apple Safari has one.

Figure 9 demonstrates the total sum of all three indices for each browser in 2024 and for the period 2021–2024. For 2024, Firefox has the highest total score of 27.29, followed by Samsung Internet with a score of 3.18. Chrome and Safari have the lowest scores for 2024, with 2.17 and 1.66, respectively. For the period 2021–2024, Edge and Firefox have the highest total scores of 186.45 and 141.13, while Safari and Samsung Internet have the lowest, with scores of 16.14 and 11.69. Detailed calculations of all indices are provided in Table S1: Calculations in the Supplementary Materials section.

The findings of this study highlight the potential for leveraging browser threat data to facilitate cyberattacks. This underscores the need for systems capable of collecting such information, providing users with actionable recommendations, such as updating their browsers. Such systems would not only mitigate user risks but also serve as a tool for broader cybersecurity efforts. For a basic user, it is recommended to keep the browser version up to date and regularly check for available updates. These actions do not require any specific knowledge and only require the user to click on “Settings”, find the “About browser_name” page, and click on it. The user will see the browser version and a “Check for updates” button, which can be clicked to update the browser version [25].

5. Discussion

The findings of this study highlight the potential for leveraging browser threat data to facilitate cyberattacks. This underscores the need for systems capable of collecting such information, providing users with actionable recommendations, such as updating their browsers. Such systems would not only mitigate user risks but also contribute to broader cybersecurity efforts.

One notable aspect is the high threat index of Mozilla Firefox, which contradicts some prior research. This discrepancy may stem from its open-source nature [26], which, while promoting transparency, also allows attackers to analyze the code for vulnerabilities. Additionally, Firefox may not have the same level of financial and engineering resources dedicated to security as Chrome or Edge, which benefit from corporate backing. Future research could further explore how funding and development models impact browser security.

Another critical observation is the significant risk associated with Microsoft Edge despite its relatively lower number of reported vulnerabilities. This suggests that not only

the quantity but also the exploitability of vulnerabilities plays a crucial role in overall security. The higher EPSS scores of Edge vulnerabilities indicate that they may be more attractive to attackers, possibly due to the browser's integration with Windows and enterprise environments. This aligns with the need for a more comprehensive approach to risk assessment beyond just counting CVEs.

5.1. Comparison with Previous Studies

According to the results of this research, Mozilla Firefox is not the safest browser, contrary to previous studies [16,19,20]. The difference in results could be due, firstly, to the different periods used for estimating browser threats, as their results are up to 2021. Secondly, during that period, Mozilla Firefox might have been significantly more popular than it is now. When accounting for the popularity ratio, it could have achieved a lower vulnerability index score in comparison.

In this research, it was found that over the last four years and in 2024, Google Chrome, Mozilla Firefox, and Apple Safari had high average CVSS scores, which differs from a previous study [17]. This difference could be attributed to technological advancements and the emergence of new cyberattacks. Additionally, Petkova's study analyzed a period starting from 1989 to 2021—a much longer timeframe during which the popularity of each browser could have varied, leading to differences in the number of vulnerabilities. Furthermore, Safari and Firefox were developed in 2002 and 2003, respectively, while Chrome was introduced in 2008. This is likely why the earlier study reported more critical vulnerabilities in Safari and Firefox.

In a previous study, the architecture of Chrome and other browsers was analyzed [18,20]. In their conclusions, they stated that the architecture of Google Chrome is better than that of other browsers. In this research, the Rt score was calculated, which can reflect the security of a browser's architecture. If a vulnerability has a high EPSS score, the probability of exploitation is high, and even a moderate user could execute such a cyberattack. According to the results, Google Chrome had a high Rt score during the 2021–2024 period, higher than Safari and Firefox, indicating the presence of several vulnerabilities in its architecture. However, in 2024, Google Chrome's Rt score was lower compared to Firefox and Safari, suggesting that developers are actively working on updates and striving to make their browser more secure.

5.2. Limitations

One of the most significant limitations that could impact the results is the reliance on data from the CVE Details database. There is no guarantee that browser vendors disclose all vulnerabilities publicly [27], meaning some vulnerabilities might be excluded from the CVE Details database, potentially influencing the final results. However, it could also genuinely indicate that no vulnerabilities occurred during the selected period.

5.3. Further Work

The findings of this study provide a foundation for developing a browser-based penetration testing system, which could be a central focus of future research. The primary objective of such a system would be to assess the information and resources that could be extracted from a user during a phishing attack. By simulating scenarios where users are directed to spoofed websites, this system could identify vulnerabilities specific to the user's browser, shedding light on the potential risks at an individual level rather than an organizational one.

Building on this, the second objective would involve exploring how the data gathered from these simulations could be leveraged to assess the user's interactions with external systems, such as their organization or other platforms they access. This approach could

help determine the extent to which vulnerabilities in a user's browser environment might facilitate broader system compromises.

Finally, integrating these insights could pave the way for developing an automated system with using AI technologies [28] to assess and mitigate user-specific cyber risks [29]. Such a system would provide real-time evaluations of vulnerabilities, offering practical tools for both individual users and organizations to enhance their security posture. Machine learning models could be employed to predict emerging threats [30] based on historical CVE data and exploit patterns, while AI-driven anomaly detection [31] could help identify suspicious browser behavior, further strengthening proactive defense mechanisms.

6. Conclusions

This study provides a systematic evaluation of browser vulnerabilities according to CVE database, utilizing developed indices such as I_{CVE}, I_{CVSS}, I_R and I_T to offer a comprehensive comparison across popular platforms. The results reveal critical insights into the security landscape of modern web browsers, emphasizing the impact of both vulnerability severity and platform popularity.

The analysis revealed that among the studied browsers, Google Chrome and Samsung Internet demonstrated the lowest vulnerability indices, indicating superior security performance during the analyzed period. Conversely, Mozilla Firefox showed the highest indices, suggesting greater exposure to potential threats. These findings underline the variability in security across platforms and highlight areas requiring further investigation.

One of the critical insights from this study is the importance of timely software updates in enhancing security. The CVE analysis indicated that many vulnerabilities were linked to outdated browser versions. Regular updates not only address known vulnerabilities but also introduce architectural improvements and new security features, significantly reducing the likelihood of exploitation.

By establishing a robust methodology for vulnerability assessment, this study lays the groundwork for future research aimed at practical applications. Specifically, the findings highlight opportunities for developing browser-based penetration testing systems that simulate real-world phishing and data extraction scenarios. Such systems could provide valuable insights into user-specific risks, paving the way for broader organizational security assessments and the automation of cyber risk mitigation strategies.

In conclusion, while this study advances our understanding of browser vulnerabilities, it also underscores the need for continued exploration of how these risks translate into actionable security measures. The results suggest that browser vulnerabilities, as documented in the CVE database, can provide valuable insights for identifying user-specific cyber risks. Developing systems that utilize this information to guide users in updating or removing risky components could significantly enhance security. This work lays the groundwork for such practical applications, forming a bridge to future research in the area of browser-based penetration testing and cyber risk mitigation. The work not only contributes to the theoretical framework of browser security but also serves as a stepping stone for future applied research, including the potential development of tools to enhance both individual and organizational cybersecurity.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/fi17030104/s1: Table S1: Calculations.

Author Contributions: Conceptualization, K.D. and O.C.; methodology, O.C.; validation, V.R.; investigation, O.C.; data curation, O.C.; writing—original draft preparation, O.C.; writing—review and editing, K.D. and O.C.; visualization, O.C.; supervision, K.D.; project administration, K.D. and

O.C.; funding acquisition, K.D. and V.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The popularity of web browsers across the years is available at https://gs.statcounter.com/ (accessed on 16 December 2024). The CVE data, including the number of vulnerabilities, CVSS, and EPSS scores for the analyzed web browsers, are available at https://www.cvedetails.com/ (accessed on 16 December 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
EPSS	Exploit Prediction Scoring System
SQL	Structured Query Language
XSS	Cross-Site Scripting
CSRF	Cross-Site Request Forgery
API	Application Programming Interface
AI	Artificial Intelligence

References

1. Xavier, H.S. The Web unpacked: A quantitative analysis of global Web usage. *arXiv* **2024**, arXiv:2404.17095. [CrossRef]
2. Petrosyan, A. Number of Internet and Social Media Users Worldwide as of November 2024. *Statista*. 2024. Available online: https://www.statista.com/statistics/617136/digital-population-worldwide/ (accessed on 16 December 2024).
3. Allen, J.W. *The Internet for Surgeons*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2007.
4. Rasool, A.; Jalil, Z. A Review of Web Browser Forensic Analysis Tools and Techniques. *Res. J. Comput.* **2020**, *1*, 15–21.
5. Chalyi, O.; Kolomytsev, M. Comparison of Tools for Web-Application Brute Forcing. *Theor. Appl. Cybersecur.* **2023**, *4*, 32–38. [CrossRef]
6. Harry, C.; Sivan-Sevilla, I.; McDermott, M. Measuring the size and severity of the integrated cyber attack surface across US county governments. *J. Cybersecur.* **2025**, *11*, tyae032. [CrossRef]
7. Constantinescu, V. Google Patches Zero-Day Vulnerability with Emergency Chrome Update. *Bitdefender*. 2024. Available online: https://www.bitdefender.com/en-us/blog/hotforsecurity/google-patches-zero-day-vulnerability-with-emergency-chrome-update (accessed on 16 December 2024).
8. Winger, D. Update Chrome Now—Google Warns Of 2 New High-Risk Vulnerabilities. *Forbes*. 2024. Available online: https://www.forbes.com/sites/daveywinder/2024/12/11/update-chrome-now-google-warns-of-2-new-high-risk-vulnerabilities/ (accessed on 16 December 2024).
9. Lakshmanan, R. 2 New Mozilla Firefox 0-Day Bugs Under Active Attack—Patch Your Browser ASAP! *The Hacker News*. 2022. Available online: https://thehackernews.com/2022/03/2-new-mozilla-firefox-0-day-bugs-under.html (accessed on 16 December 2024).
10. Reading, D. Critical Mozilla Firefox Zero-Day Allows Code Execution. *Dark Reading*. 2024. Available online: https://www.darkreading.com/cyberattacks-data-breaches/critical-mozilla-firefox-zero-day-code-execution (accessed on 16 December 2024).
11. Markkandeyan, S.; Ananth, A.D.; Rajakumaran, M.; Gokila, R.G.; Venkatesan, R.; Lakshmi, B. Novel hybrid deep learning based cyber security threat detection model with optimization algorithm. *Cyber Secur. Appl.* **2025**, *3*, 100075. [CrossRef]
12. Clark, D.; Testart, C.; Luckie, M. A path forward: Improving Internet routing security by enabling zones of trust. *J. Cybersecur.* **2024**, *10*, tyae023. [CrossRef]
13. Admass, W.S.; Munaye, Y.Y.; Diro, A.A. Cyber security: State of the art, Challenges and Future Directions. *Cyber Secur. Appl.* **2024**, *2*, 100031. [CrossRef]
14. Parla, R. Efficacy of EPSS in High Severity CVEs found in KEV. *arXiv* **2024**, arXiv:2411.02618. [CrossRef]
15. Simpson, A. Into the unknown: The need to reframe risk analysis. *J. Cybersecur.* **2024**, *10*, tyae022. [CrossRef]
16. Tewari, N.; Datt, G. A Study On The Systematic Review Of Security Vulnerabilities Of Popular Web Browsers. In Proceedings of the 2021 International Conference on Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan, 10–12 November 2021. [CrossRef]

17. Petkova, L. Security Analysis on Browsers. *Knowl.—Int. J.* **2024**, *49*, 469–474.
18. Šilić, M.; Krolo, J.; Delač, G. Security vulnerabilities in modern web browser architecture. In Proceedings of the 33rd International Convention MIPRO, Opatija, Croatia, 24–28 May 2010; pp. 1240–1245.
19. Woo, S.W.; Alhazmi, O.H.; Malaiya, Y.K. An analysis of the vulnerability discovery process in webbrowsers. In Proceedings of the 10th International Association of Science and Technology for Development: Software Engineering and Applications (IASTED SEA), Dallas, TX, USA, 13–15 November 2006; pp. 13–15.
20. Fajar, A.; Yazid, S. Web Browser threats and Weakness Descriptive Analysis: Is it Chrome Keep Dominant? *Int. J. Eng. Technol.* **2018**, *7*, 242. [CrossRef]
21. StatCounter. Browser Market Share Worldwide. Available online: https://gs.statcounter.com/ (accessed on 16 December 2024).
22. CVE Details. CVE Security Vulnerability Database. Security Vulnerabilities, Exploits, References and More. Available online: https://www.cvedetails.com/ (accessed on 16 December 2024).
23. Chalyi, O.; Stopochkina, I. Information retrieval and deanonymization in the tasks of early detection of potential attacks on critical infrastructure. *Cybersecur. Educ. Sci. Tech.* **2024**, *2*, 305–322. [CrossRef]
24. GitHub. Understanding Risk—Risk Based Prioritization. Available online: https://riskbasedprioritization.github.io/risk/Understanding_Risk/ (accessed on 16 December 2024).
25. Christiansen, P. How to Update Your Web Browser. Available online: https://www.highspeedinternet.com/resources/how-to-update-web-browser (accessed on 14 February 2024).
26. Oshri, I.; de Vries, H.J.; de Vries, H. The rise of Firefox in the web browser industry: The role of open source in setting standards. *Bus. Hist.* **2010**, *52*, 834–856. [CrossRef]
27. Johnson, P.; Gorton, D.; Lagerström, R.; Ekstedt, M. Time between vulnerability disclosures: A measure of software product vulnerability. *Comput. Secur.* **2016**, *62*, 278–295. [CrossRef]
28. Chalyi, O. An Evaluation of General-Purpose AI Chatbots: A Comprehensive Comparative Analysis. *InfoScience Trends* **2024**, *1*, 52–66. [CrossRef]
29. Woods, D.W.; Wolff, J. A history of cyber risk transfer. *J. Cybersecur.* **2025**, *11*, tyae028. [CrossRef]
30. Edkrantz, M.; Said, A. Predicting Cyber Vulnerability Exploits with Machine Learning. In *Frontiers in Artificial Intelligence and Applications*; IOS Press: Amsterdam, The Netherlands, 2015; Volume 278. [CrossRef]
31. Omobolaji, O.; Samuel, U.O.; Udochukwu, I.; Abidemi, A.S.; Tunboson, O.O.; Oluwaseun, O.O. Combating the Challenges of False Positives in AI-Driven Anomaly Detection Systems and Enhancing Data Security in the Cloud. *Soc. Sci. Res. Netw.* **2024**, *17*, 264–292. [CrossRef]