**Open Access Library Journal**
2025, Volume 12, e12756
ISSN Online: 2333-9721
ISSN Print: 2333-9705

# Vulnerability and Accessibility Analysis of Bangladesh Ministry of Land's Government Websites

**Noor-E-Sefat Ahmed**

ALAMS, D.K.M.P, Ministry of Land, Dhaka, Bangladesh
Email: sifatahmedabc7@gmail.com

**How to cite this paper:** Ahmed, N.-E-S. (2025) Vulnerability and Accessibility Analysis of Bangladesh Ministry of Land's Government Websites. *Open Access Library Journal*, **12**: e12756.
https://doi.org/10.4236/oalib.1112756

**Received:** December 2, 2025
**Accepted:** January 27, 2025
**Published:** January 30, 2025



## Abstract

Since Bangladesh recently announced the Smart Bangladesh concept, the Government has decided to move its national services online. To that end, they have built websites for each sector, including the Land Ministry, to serve the nation. The initial goal of this step is to ensure that the service is equal and hassle-free in both urban and rural areas of the country. With this modern technological support, almost one hundred percent of the Land Ministry's office work has shifted to online services. However, with these advancements, some drawbacks, such as security concerns related to data safety risks, accessibility, and vulnerabilities, have emerged, threatening the nation's billions of sensitive data. Common vulnerabilities found on these sites, such as SQLi and XSS, could expose the nation to significant threats. This paper aims to identify various Common Vulnerabilities and Exposures (CVE), Common Weakness Enumerations (CWE), potential XSS vulnerabilities, and SQLi possibilities on the websites of the Land Ministry. To do so, the study employs penetration testing and scans six types of risk alerts (high, medium, low) on the Land Ministry's websites using OWASP ZAP and Vega tools. Surprisingly, security concerns were not properly addressed during the development phase of these websites in Bangladesh. Based on the collected data and its analysis, this study concludes with an assessment of the current accessibility issues and vulnerabilities on the Land Ministry's websites.

## Subject Areas

Computer Engineering, Computer Vision,
Information and Communication: Security, Privacy, and Trust

## Keywords

SQLi, Bangladeshi Government Websites, Accessibility, Website Vulnerabilities

## 1. Introduction

Services through websites are almost fully established and provide efficient service access across Bangladesh, as well as enabling cybercrime. In the current time, the quality of services is not limited physically because of websites. Anybody can access services while sitting at home, so hackers are targeting websites to exploit security loopholes. In recent times, we have witnessed a significant amount of cybercriminal activity. According to a study from a security magazine, cybercrime occurs every 39 seconds [1]. A major cyberattacks recently took place in Bangladesh, resulting in damages of roughly USD $101 million. Hackers successfully carried out five damaging instructions out of thirty-five attempts, targeting the Bangladesh Central Bank's account in the Federal Reserve Bank of New York. The goal was to transfer USD $1 billion [2] [3]. Websites used for national services are now commonplace in Bangladesh. After the government announced the "Smart Bangladesh" initiative, nearly every service in the country became accessible through websites. This shift has placed more importance on the maintenance and security of these websites. However, in Bangladesh, many website developers do not implement secure coding practices like those outlined in the CWE Premises 7, leaving websites vulnerable to cyberattacks. A recent report by security company Kaspersky ranked Bangladesh 8th for infection rates online and 7th locally, with 54.74% of users facing infections [4], compared to other countries. These security vulnerabilities can expose government websites to a range of cyberattacks, spread fake news, and lead to financial losses for civilians. Since the impact of COVID-19, cyberattacks have increased by nearly 600% [5]. Given these developments, it is clear that Bangladesh is not yet secure from cyberattacks, particularly as developers do not always use secure coding practices. Recent incidents, such as NID information theft through Land Ministry websites, reinforce this concern. Therefore, there is an urgent need for a secure environment in which people can access national services risk-free. The objective of this study is to analyze the security and vulnerabilities of the Land Ministry's national websites, helping the government address these issues and reduce the risk of cyberattacks. This study will also assist the ministry in ensuring minimal data loss or service disruption if an attack occurs in the future. To achieve this, I have analyzed the collected data using the latest penetration testing tools such as OWASP ZAP and VEGA [6]. I identified security loopholes such as potential SQL Injection (SQLi) or Cross-Site Scripting (XSS) vulnerabilities, as well as Common Vulnerabilities and Exposures (CVE) and Common Weakness Enumerations (CWE) risks. Given that the security landscape is constantly changing, I used up-to-date data collected until 15/11/2024 for this study.

The paper is organized as follows: Section 2 presents the literature review, Section 3 outlines the methodology, Section 4 discusses the analysis and results, Section 5 provides the discussion, and Section 6 concludes the study.

## 2. Literature Review

The most critical concern in recent years is the concern of cybersecurity more preciously about Government service websites. Government organizations are

always top to face the most security challenges worldwide. So, this section of the study elaborates on the review of existing related work on accessibility and vulnerability analysis based on the Bangladesh government and land service websites.

With concerns like this topic Touhid Bhuiyan *et al.* Performed a study [7] [8] on types of SQLi present in Government and non-government websites of Bangladesh and they found that a hundred web applications five hundred ten are at risk of SQLi and another ninety are risky for GET POST base. Additionally, they ran a test on another 359 web apps and found 86% of them are at risk and were 19% high 18% medium and 63% low risk. Statistical evaluation is the objective of the study.

And have research in this [9] field by MD. Asaduzzaman Masum, Md. Rishad Istiak Sachcha, and Abu Nayem, Their focus is on finding vulnerabilities like SQLi XSS on government sites and also working to find out the risk level of those sites from high to low and additionally the part of Info, they use tools for penetration testing and find top five Vulnerability type those are Clickjacking, Misconfiguration, Cross-Site Request Forgery, Information, Cross-Site Scripting. The objective of this study is to find top security holes in Bangladesh.

We have a recent study by Totul *et al.* The objective of the study is to [10] find the 'total block time' and 'load time' parameters in various Bangladeshi e-commerce websites.

Vulnerability measurements are an effective way to prevent and reduce cyberattacks; because of that reason, some concerned citizens of Bangladesh have done some appreciable work mentioned above those studies focused on identifying the risk factors mostly their work goes through SQLi and XSS to identify the top five vulnerability and one is to identify the 'total block time' and 'load time' parameters. **Figure 1** provides an insightful illustration of how these authors have effectively conducted their research, highlighting key aspects and focus areas of their study.

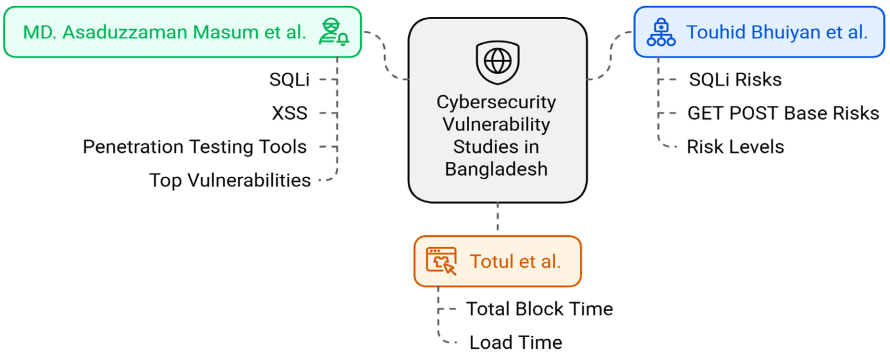


**Figure 1.** Literature review.

To understand why my work is different, we have to understand that vulnerabilities of any website are not permanent. With every update on a website, the vendor solves previously found risk factors and also if they didn't practice secure

coding unknowingly they add some new risk factors, and websites are subject to update on a regular base so previous study's are not valuable on this ever-changing topic because their data is backdated, but in my study, I am using most recent data until 15/11/2024. Also, no work specifically focuses on the Land Ministry of Bangladesh which I took as a chance to work on. Other than that, I have some more highlighted risk factors CVE, CWE, SSL Certification, and the Website Encryption status, which were not given proper importance in previous studies, so this study is going to fill those gaps.

## 3. Methodology

Vulnerabilities in websites can lead to severe consequences, especially when the affected website belongs to the government, as the threat becomes nationwide. In such cases, the potential cost includes unpredictable losses of sensitive data and financial resources. This part of my paper belongs to the methodology section, where discussions focus on how I set up my environment, selected tools for data collection, and efficiently executed the process of collecting and processing data. Most importantly, this section highlights the techniques used for data analysis, as shown in Figure 2. To achieve this, I identified the target websites and conducted penetration tests on the selected sites using my established environment and chosen tools. The test results were saved for further analysis, and through preprocessing and analyzing the collected data, I obtained the desired outcomes.

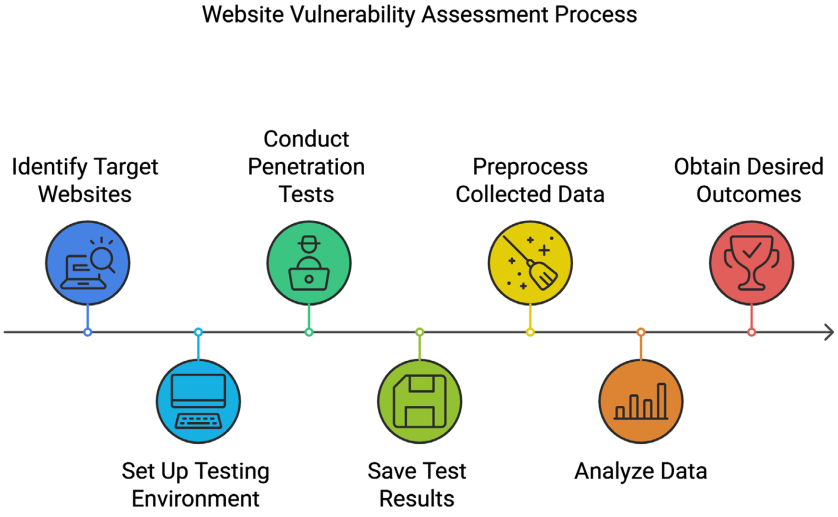


Figure 2. Website vulnerability assessment process.

### 3.1. Environment Setup

To conduct the necessary tests for this study, I utilized my personal laptop running on the Windows 10 operating system. Notably, the tools selected for data collection recently introduced a Windows GUI version, eliminating the need for a Linux OS setup and creating a hassle-free testing environment. The hardware specifications of my laptop include an Intel Core i7 11th generation CPU, 16GB

of memory, a 500GB SSD, and a 25 Mbps internet connection. These specifications were sufficient to meet the requirements for running all tests without significant performance limitations.

### 3.2. Tools Selection

A successful cyberattack on the Land Ministry's websites could have catastrophic consequences, including the loss of confidential national data and significant financial damage to the government. Additionally, the use of the tested URLs in this study could inadvertently expose them to cybercriminals, increasing the risk of exploitation. For security reasons, public URLs are not disclosed in this research to ensure the safety and integrity of the websites involved. To conduct successful penetration tests on my selected list of websites, I chose two advanced tools known for their versatility and efficiency. These tools provide comprehensive functionality to achieve the desired results while offering user-friendly graphical interfaces (GUI), making them accessible even for users without extensive technical expertise. Moreover, both tools are free to use and compatible with Windows operating systems, ensuring a hassle-free setup. The tools selected for this research are OWSAP ZAP 2.12.0 and VEGA.

### 3.3. OWASP ZAP and Vega: Selection Rationale and Advantages

The selection of OWASP ZAP and Vega was based on their alignment with the study's objectives, their proven capabilities in identifying web vulnerabilities, and their practical advantages. Both tools were chosen due to their open-source nature, comprehensive functionality, and ease of use. Below is a detailed justification.

1) OWASP ZAP

- Comprehensive Vulnerability Detection: ZAP features an extensive database of vulnerability fingerprints, making it highly effective in detecting SQL Injection (SQLi), Cross-Site Scripting (XSS), Common Weakness Enumerations (CWEs), and categorizing risks by severity (High, Medium, Low).
- Community Support: As an OWASP-supported tool, ZAP benefits from regular updates and a large user community that ensures reliability and accuracy.
- Flexibility: The tool supports manual and automated testing, making it versatile for in-depth penetration testing.
- Ease of Use: Its graphical user interface (GUI) lowers the technical barrier, enabling efficient usage even by non-experts.

2) Vega

- Lightweight and Targeted Scanning: Vega's streamlined interface focuses on detecting high-risk vulnerabilities, particularly SQLi and XSS, with minimal configuration.
- Standardized Procedures: It uses a well-recognized data collection methodology, ensuring consistent and reproducible results.
- Efficiency for Simple Tasks: While not as comprehensive as ZAP, Vega excels

in quickly identifying critical vulnerabilities, complementing ZAP's capabilities.

3) Comparative Advantages

- The combination of ZAP and Vega provides a robust testing framework, leveraging ZAP's depth and Vega's speed.
- Both tools are compatible with Windows OS, reducing setup complexities and ensuring consistent performance in the testing environment.

### 3.4. Collection of Data and Preprocessing

To collect and preprocess data, I began by preparing my environment, which operated on a Windows OS. I then identified the Ministry of Land service websites. Finding these websites was straightforward and did not require specialized tools. After gathering the URLs of the desired websites, I used VEGA to identify potential SQL Injection (SQLi) and Cross-Site Scripting (XSS) vulnerabilities. If VEGA failed to produce results within a specific timeframe, I switched to ZAP, as it is generally faster than VEGA. This method allowed me to efficiently test for potential XSS and SQLi vulnerabilities. Once the vulnerabilities were identified, I recorded the results in a dataset using Microsoft Excel in CSV format. For each website, if an XSS or SQLi vulnerability was detected, it was marked as "YES," and if none were found, it was marked as "NO." Figure 3 provides an overview of the collected SQLi and XSS data.

| XSS Possibility | SQL Injection Possibility |
|---|---|
| YES | YES |
| NO | NO |
| YES | NO |
| YES | No |
| YES | NO |
| NO | YES |
| YES | NO |
| YES | NO |
| YES | NO |
| YES | NO |
| YES | NO |
| YES | NO |
| YES | NO |
| YES | NO |

Figure 3. Overview of SQLi and XSS data collection and preprocessing.

To identify Common Weakness Enumerations (CWEs), I utilized ZAP scanning. After completing the scan, ZAP generated a detailed report, categorizing security issues into High, Medium, and Low severity levels. If multiple issues were detected, corresponding risk messages appeared in the alert section. Each alert included the CWE ID, allowing me to classify and organize the findings based on their severity. For the dataset, I recorded five attributes, including the "Found Count," which represents the number of identified risks, and further divided the

findings into three categories: High, Medium, and Low. Figure 4 illustrates an overview of the collected CWE data.

| CWE Found | CWE Count | High CWE Risk Rating Count | Medium CWE Risk Rating Count | Low CWE Risk Rating Count |
|---|---|---|---|---|
| YES | Multiple | 1 | 4 | 5 |
| YES | Multiple | 1 | 7 | 9 |
| YES | Multiple | 2 | 4 | 6 |
| YES | Multiple | 1 | 4 | 5 |
| YES | Multiple | 1 | 4 | 4 |
| YES | Multiple | 1 | 4 | 6 |
| YES | Multiple | 2 | 5 | 7 |
| YES | Multiple | 1 | 5 | 4 |
| YES | Multiple | 1 | 4 | 5 |
| YES | Multiple | 1 | 5 | 6 |
| YES | Multiple | 1 | 4 | 8 |
| YES | Multiple | 2 | 6 | 7 |
| YES | Multiple | 1 | 4 | 9 |
| YES | Multiple | 2 | 4 | 6 |
| YES | Multiple | 2 | 7 | 5 |
| YES | Multiple | 1 | 6 | 6 |
| YES | Multiple | 1 | 5 | 7 |
| YES | Multiple | 1 | 7 | 6 |
| YES | Multiple | 1 | 4 | 5 |
| YES | Multiple | 1 | 5 | 7 |
| YES | Multiple | 1 | 5 | 4 |
| YES | Multiple | 1 | 4 | 5 |

Figure 4. Overview of the CVE and CWE data collection and preprocessing.

Using a similar technique, I organized the Common Vulnerabilities and Exposures (CVEs) into three categories: High, Medium, and Low CVSS (Common Vulnerability Scoring System) scores. Figure 5 illustrates the CVE and CVSS data organization. The method for collecting CVEs relied solely on VEGA. After a successful scan, VEGA provides a well-organized interface displaying all potential CVE states, making it efficient to identify and categorize them.

| Potential CVE Found | Potential CVE Count | Potential CVSS High Count | Potential CVSS Medium Count | Potential CVSS Low Count |
|---|---|---|---|---|
| YES | Multiple | 7 | 2 | 1 |
| YES | Multiple | 0 | 2 | 1 |
| YES | Multiple | 0 | 0 | 2 |
| YES | Multiple | 0 | 0 | 2 |
| YES | Single | 0 | 0 | 1 |
| YES | Multiple | 3 | 2 | 0 |
| YES | Single | 0 | 0 | 1 |
| YES | Multiple | 0 | 0 | 2 |
| YES | Single | 0 | 0 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Multiple | 2 | 0 | 6 |
| YES | Multiple | 0 | 1 | 1 |
| YES | Multiple | 0 | 1 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Multiple | 0 | 1 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Single | 0 | 0 | 1 |
| YES | Multiple | 4 | 3 | 1 |

Figure 5. Overview of potential CVE and CVSS data collection and preprocessing.

After collecting the initial data, I decided to gather information that was straightforward and easy to understand, requiring no specialized tools. Specifically, I focused on the SSL certification and encryption status of the websites. To do this, I visited each website using its URL. If the URL contained "HTTPS," it indicated that the site was encrypted and SSL-certified. Conversely, if the URL appeared as "HTTP" without the "S," it meant the site was neither SSL-certified nor encrypted.

This information was collected through direct observation and manually recorded. The data was then set and preprocessed into my dataset. Figure 6 provides

an overview of the data collection and preprocessing process, while Figure 7 illustrates the method used to verify SSL certification.

| SSL Certified | Unencrypted |
|---|---|
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |
| NO | YES |

Figure 6. Overview of SSL certification and encrypted data collection and preprocessing.


Figure 7. Identifying unencrypted and not SSL-certified sites with the naked eye.

### 3.5. Data Analysis Techniques

To analyze the data for this study, I employed Python-based machine learning (ML) techniques, which provided meaningful insights into the dataset. Using these methods, I explored and assessed the accessibility and vulnerability aspects of the Bangladesh Land Ministry's websites. The analysis was conducted using two primary Python ML techniques: I. Descriptive Statistics, II. Data Visualization Methods.

Descriptive Statistics: Descriptive statistics allowed me to summarize the dataset and understand the distribution and characteristics of variables effectively. These measures included counts, percentages, frequencies, and central tendencies such as the mean, median, and averages. I also identified trends in the dataset and quantified attributes such as different types of risks. Data Visualization Methods: For data visualization, I utilized various plotting techniques, including bar charts, pie charts, and histograms. These methods enabled a visual representation of the dataset's distribution. Additionally, I used counterplots to visualize categorical features like SSL certification status and histograms to depict the distribution of numerical variables effectively.

By applying these Python-based ML data analysis techniques, I identified key patterns and trends in the dataset, as shown in Figure 8. These methods not only enhanced my understanding of the dataset but also allowed me to present the results in a visually effective and communicative manner. Ultimately, these techniques facilitated a deeper understanding of the vulnerabilities and accessibility risks associated with the Bangladesh Land Ministry's websites.

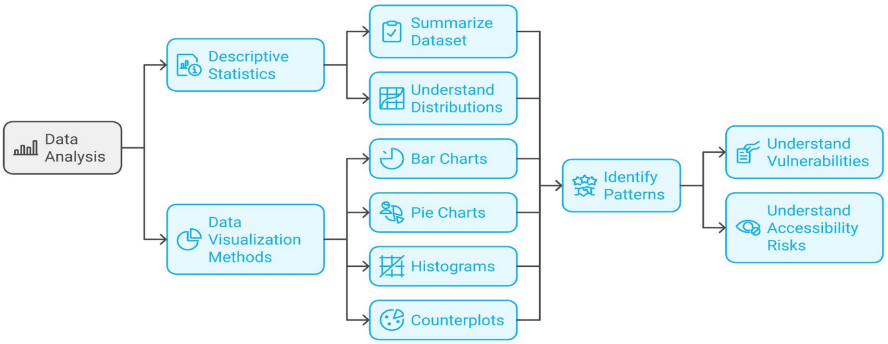


Figure 8. Data analysis flow chart.

## 3.6. Detailed Steps of Penetration Testing and Risk Alert Scanning

This section outlines the specific steps undertaken during penetration testing and risk alert scanning to ensure reproducibility and validation of the findings. The methodology is designed to be comprehensive, systematic, and adaptable for similar research contexts.

Step 1: Setting Up the Testing Environment

The testing environment was configured on a Windows 10 machine with the following specifications.

- Processor: Intel Core i7 11th generation.
- RAM: 16 GB.
- Tools: OWASP ZAP 2.12.0 and Vega, with Java (minimum version 8 for Vega and 11 for ZAP).
- Network: A stable internet connection with 25 Mbps bandwidth.

This setup ensured optimal performance for automated scanning and manual verification tasks.

Step 2: Identifying and Preparing Target Websites

The URLs of the Ministry of Land's websites were identified through government directories. The criteria for selection included:

- Websites directly associated with public services under the Ministry of Land.
- Active URLs verified for accessibility and relevance to the study.

Each URL was documented, and access permissions were ensured to maintain ethical standards.

Step 3: Initial Automated Scanning with OWASP ZAP

- Configuration: OWASP ZAP was set to "Active Scan" mode to ensure a comprehensive vulnerability check. The following parameters were configured.

- Target URLs were input into the "Sites" tab.
- Scanning filters were applied to focus on vulnerabilities such as SQL Injection (SQLi), Cross-Site Scripting (XSS), and Common Weakness Enumerations (CWE) and (CVE).

- Execution: The scan was initiated, and ZAP analyzed the structure of the websites, injecting test cases to identify potential vulnerabilities.
- Results: ZAP generated a detailed report, categorizing risks into High, Medium, and Low severity levels, along with CWE IDs and descriptions for each identified issue.

Step 4: Secondary Validation with Vega

- Configuration: Vega was deployed as a secondary tool to validate ZAP's findings and identify additional vulnerabilities. The following setup was used.
- URLs were input into Vega's target scanner.
- Predefined vulnerability profiles focusing on SQLi and XSS were activated.
- Findings of Encryption level.

- Execution: Vega performed automated scans on the same URLs, identifying vulnerabilities not flagged by ZAP and validating overlapping results.
- Comparison: The results from Vega were cross-referenced with ZAP reports to ensure consistency and accuracy.

Step 5: Risk Alert Scanning

- Severity Classification: Alerts generated by ZAP and Vega were reviewed to classify vulnerabilities into High, Medium, and Low-risk categories.
- CWE and CVE Mapping: Each identified vulnerability was mapped to its respective CWE and CVE identifiers for standardization and reference.
- Manual Cross-Validation: Alerts with significant discrepancies or ambiguous severity levels were manually verified to confirm accuracy.

Step 6: Documentation and Reproducibility

- Data Recording: All findings were systematically recorded in a structured format, including attributes such as vulnerability type, risk level, and corresponding CWE/CVE identifiers.
- Report Generation: Detailed reports were exported from both tools to ensure transparency and facilitate reproducibility.
- Ethical Considerations: URLs and vulnerability details were anonymized to prevent misuse, ensuring the security of the tested websites.

Step 7: Limitations and Scope

While the methodology was designed for comprehensiveness, certain limitations were acknowledged.

False positives inherent to automated tools were mitigated through manual validation.

Ethical constraints limited the scope of tests to non-destructive scans.

This detailed procedure ensures that other researchers can replicate the process, validate findings, and adapt the methodology for similar studies.

## 4. Result Analysis

This section presents the results of the analysis conducted on 50 selected Land Ministry websites. The findings are visualized using histograms, pie charts, and bar charts, reflecting the structure of the dataset for clarity and coherence.

CVE Analysis Results: The analysis revealed that all the national service websites in the dataset contain at least one CVE (Common Vulnerabilities and Exposures). Figure 9 illustrates this finding, highlighting the widespread presence of CVEs across the analyzed websites. Further examination showed that the distribution of CVEs is as follows: 70% of the websites have multiple CVEs. 30% of the websites have a single CVE. This distribution is visually represented in Figure 10. Additionally, the CVEs were categorized into three severity levels: High, Medium, and Low. For each category, I analyzed the count of websites with zero risks and the number of websites with varying levels of risk. The average number of risks within each category was calculated and presented for better understanding in subsequent figures.

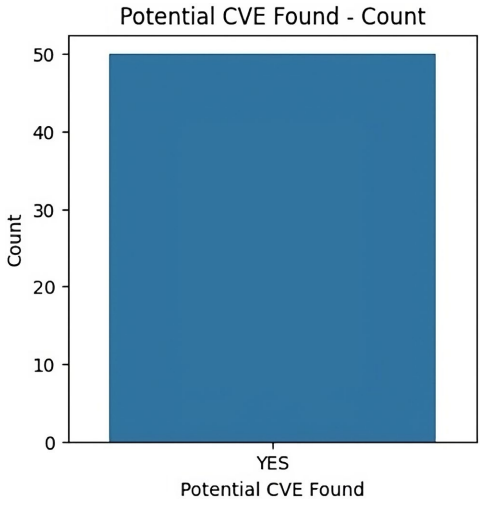


Figure 9. Result of potential CVE Found on websites of bangladesh government.

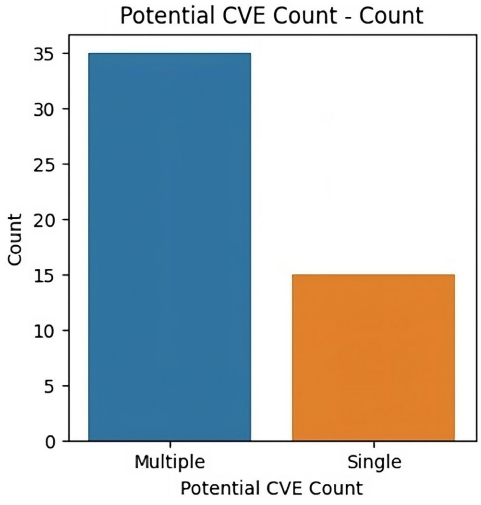


Figure 10. Percentage of CVE found on bangladesh government websites.

High-Risk CVE Analysis: In the "High" category of CVE, the results indicate an average risk of 3.03% across the analyzed websites. The total number of high-risk vulnerabilities identified in all the researched websites is 79.0.

The distribution of websites based on the count of high-risk vulnerabilities is as follows.

24 websites have 0 high-risk vulnerabilities.

10 websites have 4 high-risk vulnerabilities each.

7 websites have 1 high-risk vulnerability each.

3 websites have 3 high-risk vulnerabilities each.

3 websites have 2 high-risk vulnerabilities each.

2 websites have 5 high-risk vulnerabilities each.

1 website has 7 high-risk vulnerabilities.

This is visually represented in Figure 11.

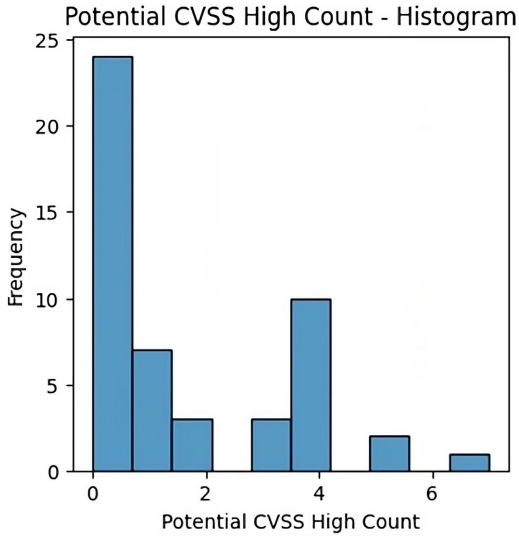


Figure 11. Number of high CVE vulnerabilities present on websites.

Medium-Risk CVE Analysis: In the "Medium" category of CVE, the results indicate an average vulnerability of 2.72%. The total number of identified medium-risk threats across all the researched websites is 68.0.

The distribution of medium-risk vulnerabilities across the websites is as follows.

25 websites have 0 medium-risk vulnerabilities.

8 websites have 4 medium-risk vulnerabilities each.

7 websites have 1 medium-risk vulnerability each.

5 websites have 2 medium-risk vulnerabilities each.

3 websites have 3 medium-risk vulnerabilities each.

2 websites have 5 medium-risk vulnerabilities each.

This info is visually represented in Figure 12.

Low-Risk CVE Analysis: In the "Low" category of CVE, the results reveal an average vulnerability of 1.49%. The total number of identified low-risk threats across all researched websites is 73.0, which is represented in Figure 13.

The distribution of low-risk vulnerabilities among the websites is as follows.
35 websites have 1 low-risk vulnerability each.
10 websites have 2 low-risk vulnerabilities each.
1 website has 0 low-risk vulnerabilities.
1 website has 6 low-risk vulnerabilities.
1 website has 3 low-risk vulnerabilities.
1 website has 4 low-risk vulnerabilities.
1 website has 5 low-risk vulnerabilities.

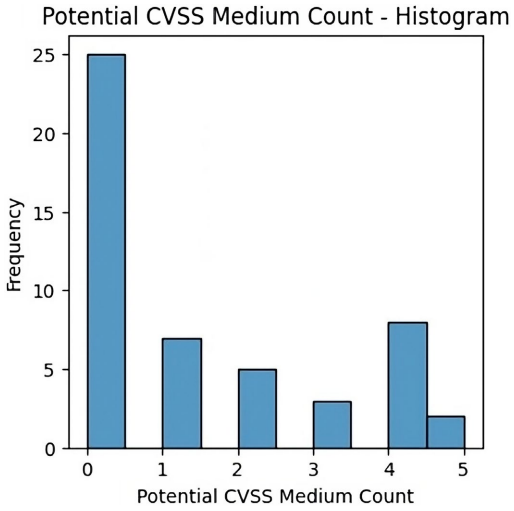


Figure 12. Number of medium CVE vulnerabilities present in websites of bangladesh.

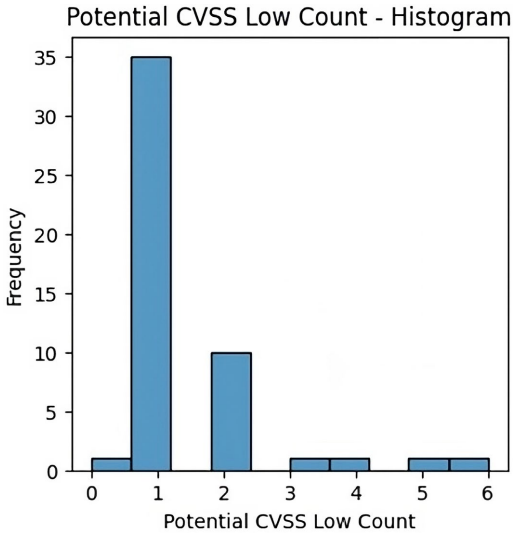


Figure 13. The number of Low CVE vulnerabilities present in websites of bangladesh.

CWE Analysis Results: The analysis of CWE (Common Weakness Enumeration) reveals that every website in the study contains multiple CWEs. This finding underscores the widespread presence of weaknesses across the selected websites. The visual representation of these results is provided in Figure 14.

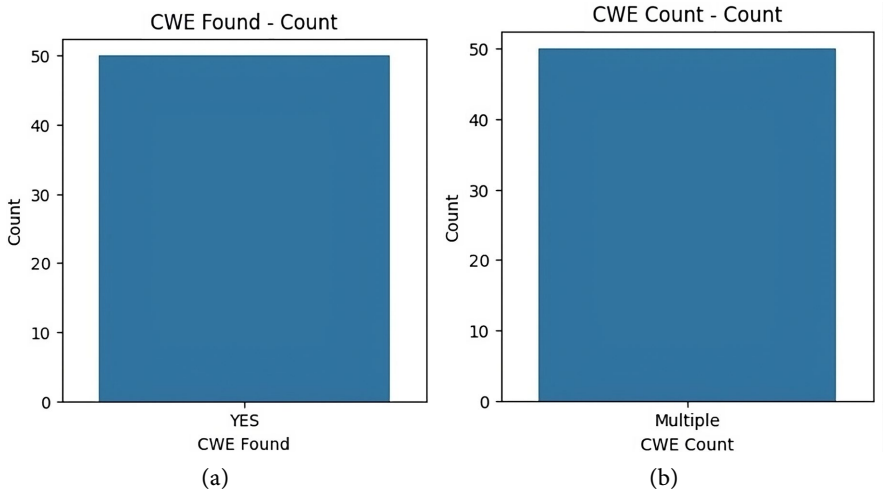


Figure 14. CWE and its numbers are found on Bangladesh Government Websites.

CWE Analysis: Similar to CVE, CWE is also categorized into three levels.

High-Risk CWE Analysis: This part of CWE, the analysis shows an average risk of 1.26%, with a total of 63 risks across all the researched websites. The distribution of these risks is as follows.

1 risk on 37 websites and 2 risks on 13 websites. This information is visually represented in Figure 15.

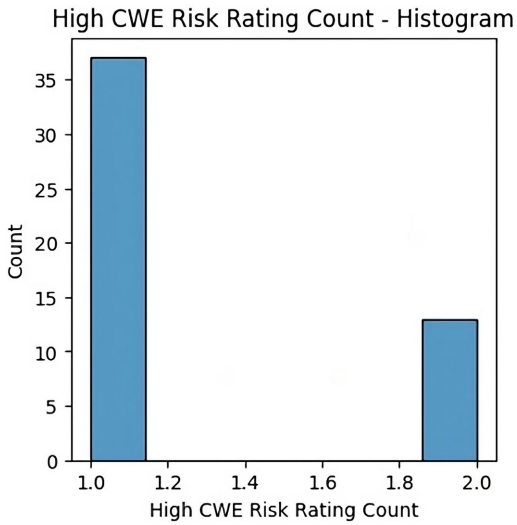


Figure 15. Number of high CWE vulnerabilities present in websites of bangladesh.

Medium-Risk CWE Analysis: For the Medium-Risk part of CWE, the analysis reveals an average vulnerability of 5.14%, with a total of 257 threats across all the researched websites. The distribution of these risks is as follows: 4 risks on 21 websites, 5 risks on 12 websites, 7 risks on 9 websites, 6 risks on 7 websites, and 8 risks on 1 website. This information is visually represented in Figure 16.

Low-Risk CWE Analysis: In the Low-Risk part of CWE, the analysis shows an average vulnerability of 6.26%, with a total of 313 threats across all the researched

websites. The distribution of these risks is as follows: 6 risks on 15 websites, 5 risks on 12 websites, 7 risks on 10 websites, 9 risks on 5 websites, 4 risks on 4 websites, and 8 risks on 4 websites. This information is visually represented in Figure 17.

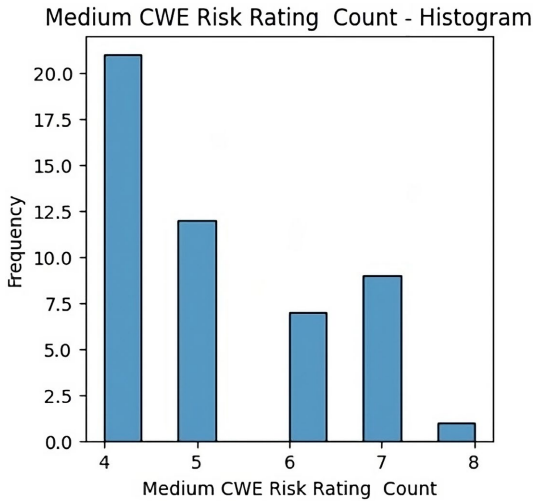


Figure 16. Number of medium CWE vulnerabilities present in websites of bangladesh.

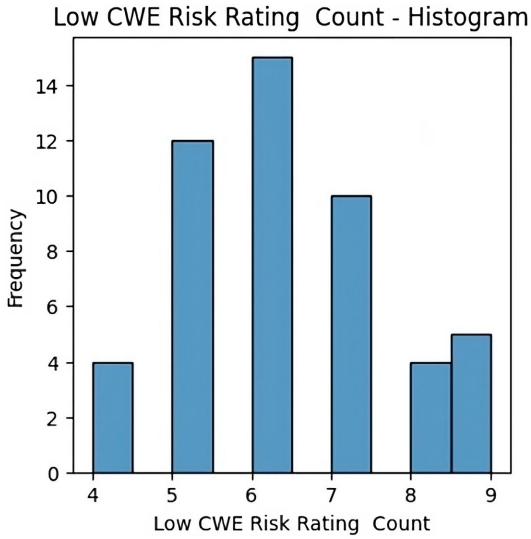


Figure 17. Number of medium CWE vulnerabilities present in websites of bangladesh.

SSL Certification and Website Encryption: Now, let's discuss the SSL Certification and Website Encryption aspect. As observed in the results, almost none of the studied Land Ministries websites are using SSL Certification, and as a result, their websites are not encrypted. This information is visually presented in Figure 18 and Figure 19.

The possibility of Cross-Site Scripting (XSS) vulnerabilities in the studied Bangladeshi Land Ministries websites shows a 90% positive result and a 10% negative result. A visual representation of this is provided in Figure 20 for better understanding.


Figure 18. Result of using SSL certification on bangladesh government websites.

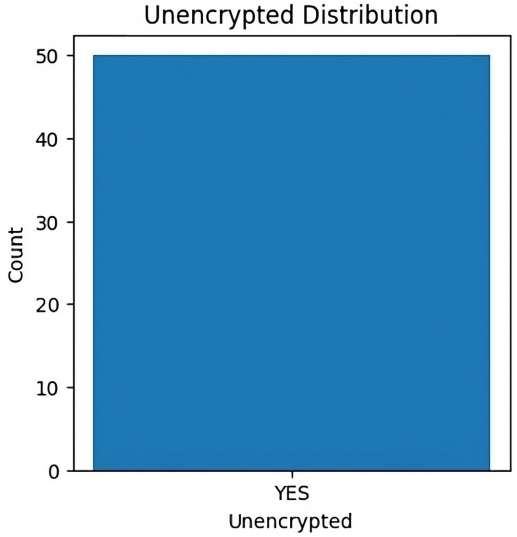


Figure 19. Result of encryption method used on bangladesh government websites.

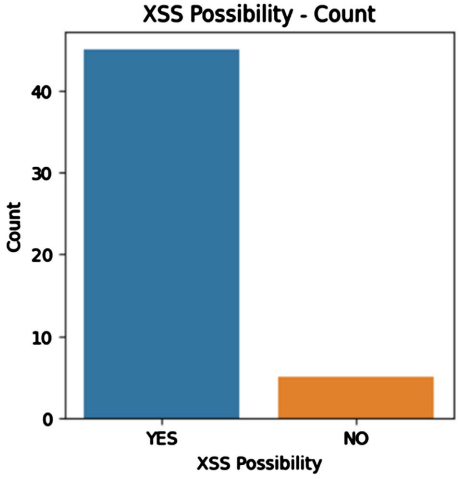


Figure 20. XSS possibility on bangladeshi government websites.

Here, I present the final result regarding the SQL Injection (SQLi) vulnerability in the studied websites. I found that 34% of the websites are vulnerable to SQLi attacks, while 66% are not vulnerable. To visually demonstrate this result, refer to Figure 21. Note that the figure includes two parts labeled "NO," as the spelling of "no" in the dataset appears in two different styles.

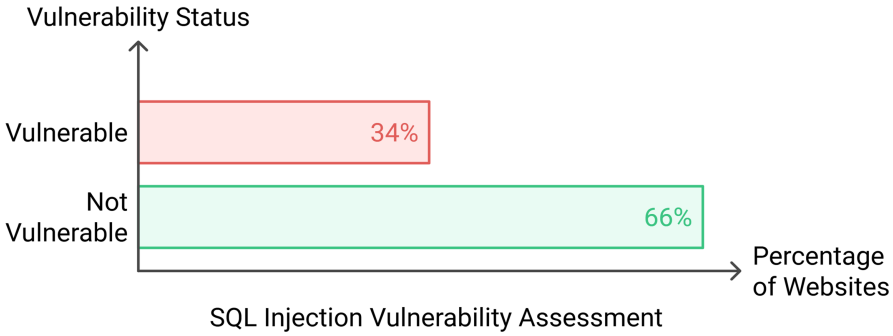


Figure 21. SQLi possibility on bangladeshi government websites.

## 5. Discussion

In previous studies, similar results were found for vulnerabilities such as XSS and SQLi, though these studies focused on different sectors of websites. While their findings were relevant at the time, the field of web security evolved with each update made to websites. My results, being the most current, show that the websites of Bangladesh's Land Ministries are at significant security risk. Specifically, the presence of a large number of XSS and SQLi vulnerabilities makes these sites highly susceptible to attacks. A skilled cybercriminal can easily exploit these vulnerabilities to gain unauthorized access and cause damage. Moreover, none of the websites I studied are SSL certified or encrypted, which further exposes them to risk. Without encryption, these websites are vulnerable to various forms of attack, including session hijacking and Man-in-the-Middle attacks. The absence of SSL certification means that users of these websites are at high risk of falling victim to such attacks. Additionally, with the CVE and CWE vulnerabilities identified, a cybercriminal could easily craft targeted attacks to gain unlawful access to these sites.

### 5.2. Shortcomings in Development

A critical observation during the study was the apparent lack of attention to security during the development phase of government websites. Many developers in Bangladesh do not adhere to secure coding practices outlined in standards like CWE Premises 7. This oversight leads to common vulnerabilities such as SQLi and XSS. Additionally, the absence of encryption and SSL certification on several websites highlights the neglect of foundational security measures. These shortcomings stem from inadequate training of developers, limited budget allocation for cybersecurity, and a lack of standardized guidelines for secure website development. (Figure 22)

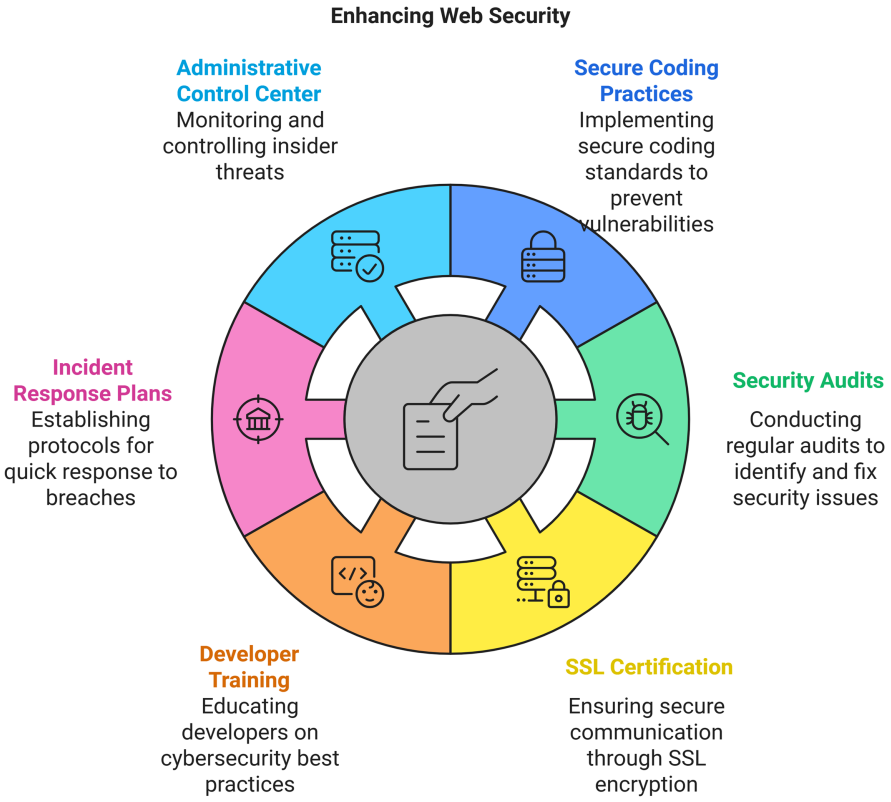


**Figure 22.** Enhancing land ministries websites.

## 5.3. Strategies to Improve Security

To address these vulnerabilities, several strategies must be implemented.

- Adoption of Secure Coding Practices: Enforce the use of secure development frameworks and conduct regular code audits.
- Regular Security Audits: Implement periodic penetration testing and vulnerability scanning using tools like OWASP ZAP and Vega.
- SSL Certification and Encryption: Mandate SSL certificates for all government websites to ensure secure communication.
- Developer Training: Provide ongoing cybersecurity training to web developers to familiarize them with secure coding standards and emerging threats.
- Incident Response Plans: Establish incident response protocols to quickly mitigate the impact of potential breaches.
- Administrative Operation Control Center (AOCC): Establish an Insider Monitoring and operating control center with a cybersecurity specialist Team.

## 5.4. Future Research Directions

This study highlights the vulnerabilities and accessibility issues in government websites, but there is scope for further research.

- Emerging Threats: Explore the impact of modern threats like ransomware and supply chain attacks on government websites.
- Sectoral Analysis: Extend the study to include other government sectors

beyond the Land Ministry.

- Automation in Security: Investigate the use of machine learning and AI for automated vulnerability detection and response.
- Policy Frameworks: Study the effectiveness of existing cybersecurity policies and propose enhancements for better implementation.

## 6. Conclusions

Websites are always a prime and accessible target for cybercriminals seeking unauthorized access. In this study, I have demonstrated that the majority of service websites of the Land Ministries in Bangladesh are in a vulnerable state. To highlight these vulnerabilities, I followed a strategic approach: first, I collected the URLs of the target websites and then used advanced tools such as OWASP ZAP and VEGA to identify security issues present on the sites. Through my findings, I have successfully presented the current security scenario of the Land Ministries' websites in Bangladesh, though regrettably, none of these sites are risk-free.

Among the vulnerabilities discovered, I identified critical risks such as SQLi, XSS, CVE, and CWE—issues that should never be present on any website, particularly government websites. Such vulnerabilities can expose sensitive information, and if exploited by malicious actors, could lead to serious consequences for the nation. The purpose of this study is to highlight these pressing vulnerabilities, so that the Ministry of Land in Bangladesh can take prompt and effective measures to mitigate these risks and prevent potential security breaches.

## Conflicts of Interest

The author declares no conflicts of interest.

## References

[1] Hackers Attack Every 39 Seconds. https://www.securitymagazine.com/articles/87787-hackers-attack-every-39-seconds

[2] The Great Bangladesh Cyber Heist Shows the Truth Is Stranger than Fiction. https://www.dhakatribune.com/opinion/op-ed/122939/the-great-bangladesh-cyber-heist-shows-truth-is

[3] Congresswoman Wants Probe of "Brazen" $81 M Theft from New York Fed. https://nypost.com/2016/03/22/congresswoman-wants-probe-of-brazen-81m-theft-from-new-york-fed/

[4] Kaspersky Security Bulletin 2020. Overall Statistics for 2015. https://go.kaspersky.com/rs/802-IJN240/images/KSB_statistics_2020_en.pdf

[5] U.N. Official Warns Cybercrime up 600% during COVID-19 Pandemic. https://www.newsy.com/stories/u-nwarns-cybercrime-up-600-during-covid-19-pandemic/

[6] Farah, T., Alam, D., Kabir, M.A. and Bhuiyan, T. (2015) SQLi Penetration Testing of Financial Web Applications: Investigation of Bangladesh Region. 2015 *World Congress on Internet Security* (*WorldCIS*), Dublin, 19-21 October 2015, 146-151. https://doi.org/10.1109/worldcis.2015.7359432

[7] Alam, D., Bhuiyan, T., Kabir, M.A. and Farah, T. (2015) SQLi Vulnerability in Education Sector Websites of Bangladesh. 2015 2*nd International Conference on Information Security and Cyber Forensics* (*InfoSec*), Cape Town, 15-17 November 2015, 152-157. https://doi.org/10.1109/infosec.2015.7435521

[8] Alam, D., Kabir, M.A., Bhuiyan, T. and Farah, T. (2015) A Case Study of SQL Injection Vulnerabilities Assessment of .bd Domain Web Applications. 2015 4*th International Conference on Cyber Security*, *Cyber Warfare*, *and Digital Forensic* (*CyberSec*), Jakarta, 29-31 October 2015, 73-77. https://doi.org/10.1109/cybersec.2015.23

[9] Masum, Md.A., Istiak Sachcha, Md.R. and Nayem, A. (2022) Security Analysis of Government & Financial Websites of Bangladesh. *International Journal of Education and Management Engineering* (*IJEME*), **12**, 21-29.

[10] Hossain, M., Hassan, R., Amjad, M. and Rahman, M. (2021) Web Performance Analysis: An Empirical Analysis of E-Commerce Sites in Bangladesh. *International Journal of In-formation Engineering & Electronic Business*, **13**, 47-54.