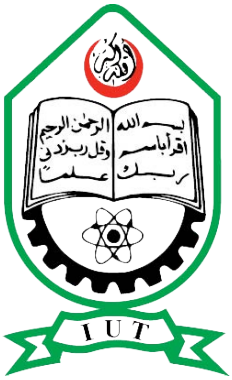
بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ



Analyzing Web Application Vulnerability of Educational Institutions in Bangladesh

Authors
Mahbubul Karim - 170042022
Afia Muntakim - 170042029
Hridita Nur Zaman Tiasha - 170042041

Supervisor
Ashraful Alam Khan
Assistant professor, Department of CSE,
Islamic University of Technology (IUT)
Co-Supervisor
S.M. Sabit Bananee
Lecturer, Department of CSE,
Islamic University of Technology (IUT)
Imtiaj Ahmed Chowdhury
Lecturer, Department of CSE,
Islamic University of Technology (IUT)

A thesis submitted in partial fulfilment of the requirements for the degree of B. Sc. Engineering in Computer Science and Engineering (CSE)

Academic Year: 2020-2021

Department of Computer Science and Engineering (CSE)
Islamic University of Technology (IUT),
A Subsidiary Organ of the Organization of Islamic Cooperation (OIC)
Gazipur-1704, Dhaka, Bangladesh

Declaration of Authorship

This is to certify that the work presented in this thesis is the outcome of the analysis and experiments carried out by *Mahbubul Karim*, *Afia Muntakim* and *Hridita Nur Zaman Tiasha* under the supervision of *Ashraful Alam Khan*, Professor of the Department of Computer Science and Engineering (CSE), *S.M. Sabit Bananee* , Lecturer of the Department of Computer Science and Engineering (CSE), *Imtiaj Ahmed Chowdhury* , Lecturer of the Department of Computer Science and Engineering (CSE), Islamic University of Technology (IUT), Dhaka, Bangladesh.

We are very grateful to our supervisors *Prof. Ashraful Alam Khan*, Department of Computer Science and Engineering, Islamic University of Technology (IUT), *S.M. Sabit Bananee*, Department of Computer Science and Engineering, Islamic University of Technology (IUT), *Imtiaj Ahmed Chowdhury* , Department of Computer Science and Engineering, Islamic University of Technology (IUT), for their supervision, knowledge and support, which has been invaluable for us.

It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Authors:

Mahbubul Karim — Student ID - 170042022

Afia Muntakim — Student ID - 170042029

Hridita Nur Zaman Tiasha — Student ID - 170042041

Approved By:

Supervisor:

Ashraful Alam Khan
Assistant professor, Department of CSE

Co-Supervisor:

S.M. Sabit Bananee
Lecturer, Department of CSE

Imtiaj Ahmed Chowdhury
Lecturer, Department of CSE

Abstract

The biggest challenge we face today is web security. It is the fundamental framework for the global data society. People's daily activities mostly depend on internet-based applications. No web application is free from threats and security issues. Clients and users make mistakes when interacting with web applications, which can lead to security issues. Besides, there are coding flaws and server misconfiguration issues which gradually lead to service failure or attacks on vulnerable information. Strong security in the web application is a vital need for online presence nowadays. Dealing with web security issues requires deep insight as there are a lot of tools available to detect vulnerabilities. Proper understanding and deep analysis are required to find the proper tool for this application. This study aims to detect vulnerabilities of the educational websites in Bangladesh and analyze which scanning tool provides more accurate results. For our analysis, we have used the two most prominent web application security scanners, Acunetix and Nikto. After scanning, many security issues and vulnerabilities were found. However, the most common vulnerability issue among all the websites was SQL injection, XSS, and Clickjacking.

Keywords: *SQL injection, XSS, Clickjacking, Acunetix, Nikto*

Contents

List of Figures

1 Introduction

1.1 Overview

We all must agree on the fact that web applications are the most effective medium for providing services in recent times. Governments and other communities rely on ICT-based architectures for collecting information, storing valuable data, connecting with people, and sharing information with trusted authorities. Education, e-commerce, banking, business, and many other sites highly depend on secured websites and web activities. We also enjoy these services through thousands of web applications. The use of websites is at every step of our life. So any interruption to these websites highly impacts our daily life.

Websites in the real world are complicated systems that share and integrate data with other systems, as well as store and process data in a variety of locations. Web application security is concerned with various software vulnerabilities that seek to cause the program to do a malicious action.For online applications, web security is a critical consideration. Today, cyber security is a major worry in the online world. It is regarded as the fundamental basis for the global data society. Through a web page, web apps provide a better interface for a client. Attackers or cybercriminals are always trying to figure out a way to exploit the security holes. That's why server-side applications release new versions with the latest updates. But, unfortunately, still many websites use defective libraries for various dependency issues. Therefore approaching cyber security policies and procedures is a must for the security of the websites. But some website authorities neglect these cyber security issues which encourage cybercriminals to perform more attacks. The three common security vulnerabilities according to our study are SQL injection, XSS, and Clickjacking. A large number of web applications in Bangladesh are exposed to security issues. A study from the Web Application Security Consortium shows that about 49 percent of the web applications contain vulnerabilities of high level. Dhaka University's website was hacked in 2017 due to some vat-related issues by the team "Cyber 71". Again, the National University website was hacked once where near about 15000 important credentials were leaked. Different news portals in our country published this news. There was another incident where Bangladesh Bank had to face a major financial loss of about 101 million dollars. All these incidents point to the lack of security measures on different websites in Bangladesh.

This research analyzed vulnerability detection of educational websites in Bangladesh using Acunetix and Nikto. Depending on the severity degree of the findings, they are classified as high, medium, or low. The data from both programs were then evaluated and compared, with Acunetix providing greater information and discovering the majority of the vulnerabilities.

1.2 Research Challenges

We faced huge challenges while performing the research activities. Compatibility with devices was the first challenge. To properly run Acunetix Nikto, we had to manage devices with higher RAM. Setup with 16GB RAM is recommended for better performance. Here we worked with some selected school, college, and university websites. We faced another challenge here. Not all the websites are properly responsive. So we had to search for a long time to select the websites where we will perform security vulnerability testing. Besides, some of the websites took about 16 hours and more to complete scanning. Any small interruption could hamper scanning. During this time if there were any issues with the internet connection or power supply, we had to start the scanning process again from the beginning. Again, the scanning reports generated from the tools were not specified properly in most of the cases. We had to manually check and figure out the proper security attacks. Lastly, there is a huge lack of open-source scanning tools. Most of the tools provide paid service. So we had to face a lot of difficulties to use the proper tools for scanning.

1.3 Contribution

We specifically worked with educational institutions in Bangladesh. We found many research papers on e-commerce, banking, health, etc. In Bangladesh, no research activities were performed with educational institutions. That's why we choose this sector.
Our contributions are:

1. We detected which security attacks mostly occur and which attacks are less frequent.
2. A comparison between two popular tools.
3. We selected 3 types of educational sectors(Schools, Colleges, and Universities among which , We figured out which educational sector is at high risk of security vulnerabilities.

2 Background Study

2.1 Attacks:

Cross-Site Scripting(XSS)
One of the most dangerous online security flaws is cross-site scripting. It allows attackers to inject malicious scripts into a web application. Attackers send malicious links to the users and manipulate them to click on the link. The malicious link executes the selected operation on the user device. Thus the attacker steals the user's session cookie. (1) Through a cross-site scripting attack, an attacker:

1. **Redirects a user to some malicious or unauthorized website.**
2. **Crashes browser activities.**
3. **Steals the cookie information of a user**
4. **Manipulates a user to perform illegal steps.**

In the worst cases, an XSS attack can destroy the victim's whole account. Thus many important credentials can be leaked into unsafe hands. There are basically 3 types of XSS attacks: Stored XSS, Reflected XSS, and DOM-based XSS.

2.2 Buffer Overflow

Buffer overflow is a software programming vulnerability. It might be used by attackers to gain unauthorized access to company systems. Buffer overflow is one of the most vulnerable security attacks. Software faults generally affect buffers, which provide temporary storage for data as it travels between locations. In general, buffer overflow is caused when too much data is being stored. Those extra data overflow into the nearest memory locations and overwrite the data. A buffer overflow attack occurs when:

- **The code relies on external data to execute its operations.**
- **The code is so hard that even the programmers do not get its behavior correctly**

Due to buffer overflow attacks, a system crashes, the authority may lose access to a program, and important data may be lost. Stack-based, heap-based, and format string buffer overflow attacks are the most prevalent forms of buffer overflow attacks.
Each strategy includes a variety of tactics that have been seen in the field being utilized in hacks

by malware or threat actor organizations. Techniques are used to describe how attackers escalate privileges or how adversaries exfiltrate data, for example.

Each tactic contains an array of techniques that have been observed being used in the wild by malware or threat actor groups in compromises. Techniques are thought of as how attackers are escalating privileges or how adversaries are exfiltrating data, etc. The number of techniques are too many to list but can be visualized using the MITRE ATT&CK Navigator which is a web based application.(2)

2.3 Cross-site request forgery)

Cross-site request forgery is a security vulnerability by which the attackers manipulate the users to perform something malicious that they do not want to do. The same-origin policy is designed to prevent websites from interacting with each other, but it can be partially bypassed. CSRF attacks can occur if there is any kind of relevant action, any sort of cookie-based session handling. Most of the CSRF vulnerability occurs due to faults in the validation process of CSRF tokens. The validation process of the CSRF tokens depends on the request method.

To carry out a CSRF attack, two things must happen: An attacker uses social engineering, such as phishing, to fool an authenticated or logged-in user into clicking a link or loading a website. Your web application stores CSRF token values in cookies, which isn't a good idea for web applications because revealing cookies might also disclose CSRF tokens. Authentication tokens should be maintained separate from cookies and used solely for account change actions.

2.4 SSL Attacks

SSL stripping attacks are a sort of cyber attack in which hackers lower the security of an online connection from HTTPS to HTTP. HTTP is less safe since it sends data in plaintext, but HTTPS is more secure because all data is encrypted. HTTPS encrypts data using SSL/TLS, which is a digital certificate that can both verify identities and encrypt data. When a hacker interferes with the connection between a user and a website, this is known as SSL stripping. The hacker sits amid the connection, connecting to the HTTPS version of the site while the user connects to the HTTP version. This gives them unencrypted access to anything the user says. Usually, there are 3 possible ways an attacker can gain access to perform SSL stripping attacks: Proxy servers, ARP spoofing, and Network access. The possible risks regarding SSL attack can be:

1. **Stolen information.**
2. **Improper communication.**
3. **Fraud transactions.**

2.5 Structured Query Language Injection

SQLI stands for Structured Query Language Injection. Attackers are able to use SQL injection to spoof identity, tamper with existing data, cause repudiation issues such as voiding transactions or changing balances, and allow complete access to all system data. The following are some examples of SQL injection: Changing a query to retrieve hidden data will give you more results. Subverting application logic entails modifying a query in such a way that it disrupts the program's logic. Data from several database tables may be obtained via UNION attacks. By using SQL injection, a hacker will try to enter specially prepared SQL instructions into a form field instead of the intended data. The purpose is to acquire a response from the database that will allow the hacker to deduce how the database is constructed, such as table names.

2.6 Clickjacking

Clickjacking is a process of fooling a user. A user is manipulated into thinking that they are clicking on one thing while they are clicking on another. A user thinks they're using a standard UI on a webpage, but in reality, a concealed UI is in control. When consumers click on something they believe to be safe, the hidden UI performs a different action.

The purpose of Clickjacking isn't to make consumers think they're doing something safe; rather, it's to make them think they're doing something safe. The real assault can be anything that can be done through web pages. This can range from dangerous operations like installing malware or stealing credentials to more benign activities like raising click counts on unrelated sites, increasing ad income on sites, increasing Facebook likes, or increasing YouTube video views.

2.7 Cookie Vulnerability

Cookies that are not produced or sent securely can be hijacked or poisoned. Cross-site scripting (XSS) is a frequent method of stealing cookies, although other techniques such as packet sniffing and brute force can also be used to get unwanted access to cookies. Cookie Theft, commonly known as "pass-the-cookie assault," is a session hijacking technique that uses session cookies saved in the browser to gain access to user accounts. Tracking cookies compromise your online privacy by collecting information about you without your knowledge. Tracking cookies, like third-party cookies, follow your activities across several websites rather than improving your experience.

2.8 Directory Traversal

Path traversal, also known as directory traversal, is an HTTP attack that allows attackers to access restricted files and perform commands that are not located in the web server's root directory. An Access Control List is used in the permission process. A directory traversal vulnerability is caused by insufficient filtering/validation of browser input from users. Directory traversal vulnerabilities can be identified in web server software/files and server-side application code. An attacker can use the system's directory traversal vulnerability to get access to other parts of the file system, allowing them to study restricted files and gather more information to further compromise the system.

2.9 Denial-of-Service (DoS) attack

A DoS attack seeks to put a system or network to a standstill, leaving it unavailable to its intended users. DoS attacks force the target to crash by flooding it with traffic or sending information. A distributed denial-of-service (DDoS) attack occurs when several machines collaborate to attack a single target. To mount large-scale DDoS assaults, DDoS attackers usually use a botnet, which is a group of hijacked internet-connected devices. Denials of service, for example, are prevalent during Black Friday sales, when thousands of people are begging for a bargain. However, they can be hazardous. In this picture, an attacker tries to deliberately drain the site's resources, preventing legitimate users access.

2.10 Broken Files

Computer files that have been corrupted become dysfunctional or useless. A file can get corrupted for a variety of reasons. In certain circumstances, the damaged file can be recovered and fixed, while in others, it may be necessary to delete the file and replace it with a previously saved version. Symptoms that appear to be virus-related might be caused by common software issues

like program execution difficulties and corrupted files, therefore it's crucial to distinguish between viral symptoms and those caused by corrupted system files.

3 Tools

Computer files that have been corrupted become dysfunctional or useless. A file can get corrupted for a variety of reasons.(3) In certain circumstances, the damaged file can be recovered and fixed, while in others, it may be necessary to delete the file and replace it with a previously saved version. Symptoms that appear to be virus-related might be caused by common software issues like program execution difficulties and corrupted files, therefore it's crucial to distinguish between viral symptoms and those caused by corrupted system files.(4)

1. **Acunetix: (5)**Using Acunetix, you can scan and identify vulnerabilities such as SQL injection, cross-site scripting, and other exploitable issues in your online applications. Acunetix was designed initially as a web application security scanner, with network infrastructure scanning added afterward. Its scanning engine is OpenVAS, a renowned open-source vulnerability scanning project. Acunetix is one of the most essential online application penetration testing tools and resources for hackers and security experts, according to GBHackers on Security.

2. **Nikto:(6)** Nikto is a pluggable web server and CGI scanner written in Perl that leverages RFP's LibWhisker for rapid security and informative testing. Database of CSV-format tests that may be quickly updated. Nikto is a free and open-source web server scanner that checks a website for vulnerabilities that might be exploited or hacked. It's also one of the most widely used website vulnerability tools in the industry, and in many ways, it's the industry standard.

4 Literature Review

Previously some studies were done on the E-Commerce sectors in Bangladesh. They conducted their analysis by Acunetix and Nikto (7). Some vulnerabilities were figured out that are potential and have high chances to attack the e-commerce web applications.(7) Bangladesh has a great prospect for e-commerce as the Government has declared IT as a thrust sector and that a computer training center will be set up in each divisional and district headquarters of Bangladesh. (2) The government has made computer gear and software imports duty-free, deregulated VSAT (Very Small Aperture Terminal), and launched high-speed DDN (Digital Data Network). The bulk of early e-commerce enterprises in Bangladesh were C2C (Consumer to Consumer) and B2B (Business to Business). Malicious attacks intended to impair web applications can be bifurcated into disrupting the service it provides to end-users and another to leak valuable information about users from web application servers (8).
A single vulnerability in a web application may cost us money, let others to mimic us on social media sites, expose our personal information, and even jeopardize national security by stealing key military data or causing a power outage. The vulnerabilities found on the investigated e-commerce web apps were scanned, and the results were used as the starting point for additional analysis. According to their severity in terms of the potential harm they might cause to the system or users, the collected findings are classified as high, medium, low, and informational level security defects.

Some research activities took place with the educational institutions that had exposed public IPs. These institutions were not from Bangladesh. Here 4 tools were used for vulnerability scanning- Advanced IP Scanner, Shodan, Nessus, and Acunetix. (8) This study was based on the evaluation of the overall security infrastructure of the higher educational institutions and their status regarding cyber threats. In this process, Nessus was used at a wide range to discover the vulnerabilities of the network. Moreover, some exposed IP addresses of educational institutions were selected for scanning where an Advanced IP scanner generated a report with all the vulnerable IP addresses. Then performing Shodan analysis, the previous vulnerabilities were analyzed. It generated a report of how many times those vulnerable attacks occurred in a network. Finally, they generated a report with the top 5 most security threats or vulnerabilities. So at a glance, the previous studies or research activities did not focus on the educational sectors in Bangladesh. Besides, a proper comparison between the vulnerability scanners used for research purposes did not happen. These were our focus while performing the study.

5 Problem Formulation

The goal of our study consists of 3 parts. Firstly, we want to figure out to which security attacks the educational websites in Bangladesh are mostly exposed. Secondly, we want to perform our analysis in 3 parts- School websites, College websites, and university websites. Our target is to find out which educational sector is most under threat. And finally, we want to make a comparison between the two tools- **Acunetix** and **Nikto** to see which scanning tool provides better predictions (9).
So in one sentence, our problem statement is, "Security vulnerability and risk level analysis of the educational institutions in Bangladesh". The purpose of this study is to create awareness among the website developers about the security threats and attacks as they are mostly overlooked while building a website. It will improve the security posture, and protect personal devices and important credentials from attackers.

6 Methodology

After analyzing all the possible threats and security measures against them, first, we selected some of the top responsive school, college, and university websites. The first issue we faced here was that many of those websites were not that responsive. Then those responsive educational websites were scanned one by one using the vulnerability scanning tools Acunetix and Nikto. After generating reports from all the websites, the most vulnerable security threats were analyzed. Then we separately analyzed the status of the 3 sectors. From this process, we found the sector which is at a high-security risk. Finally, we made a comparison of the two vulnerability scanners we used.

6.1 Experimental Workflow

The following 1 diagram shows the basic workflow of our experiment. Here, 1stly we listed the websites we are going to scan. Then we selected the tools we are going to work with which were Acunetix and Nikto. Then we scanned the selected websites using the scanner and collected the reports. Then we analyzed the reports and generated the results. After that, we categorized the vulnerabilities and compared the performance of the tools.

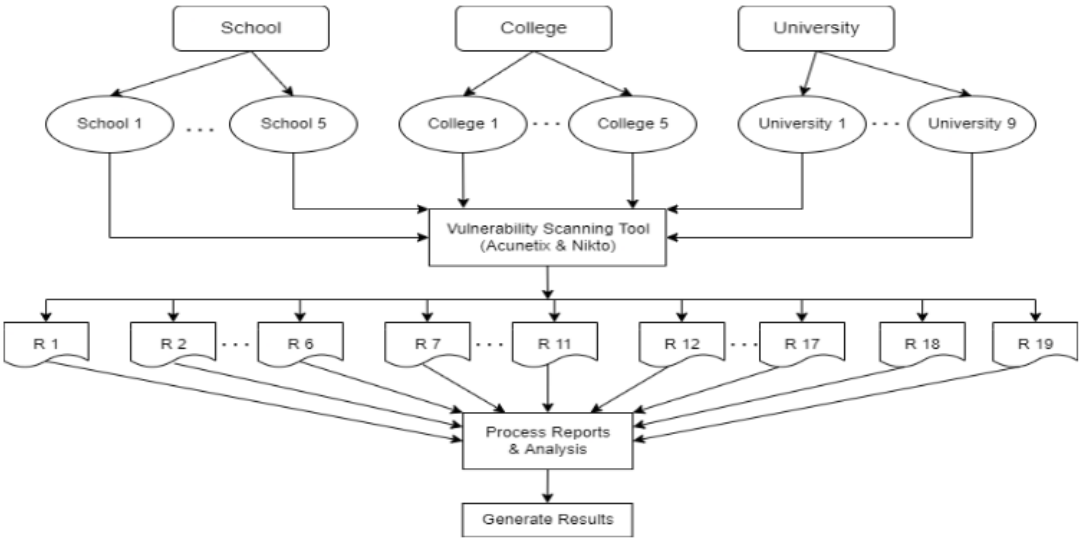


Figure 1: Experimental Workflow

7 Implementation

A total of 19 websites was listed up for scanning, including the top 5 websites of schools and college and the top 9 websites of the university. Then we listed down their websites and scanned them according to their category.

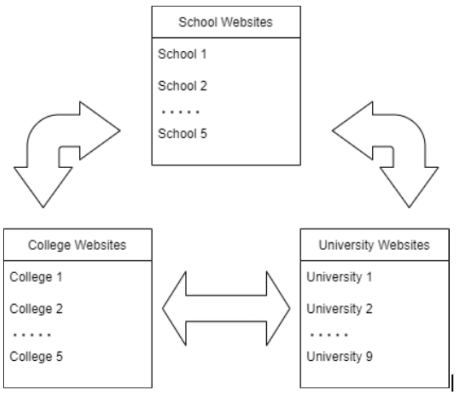


Figure 2: Website selection

Then we selected the scanning tools. We selected Acunetix and Nikto. We set Nikto because it is an open-source scanning tool. On the other hand, Acunetix provides a trial version which was convenient for us. We used Acunetix trial version 13. We then set a virtual machine on our pc and Scanned the listed websites.

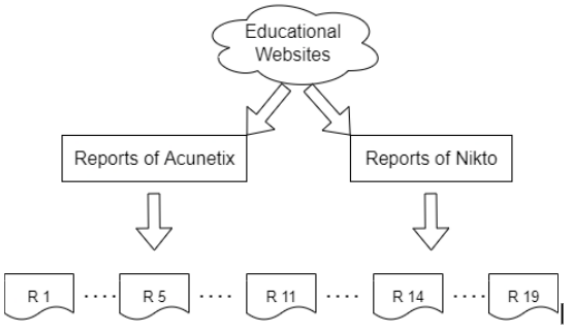


Figure 3: Scanning the websites

Then we collected the report that the scanners generated and analyzed them. We listed the vulnerabilities that we found and also the number of times we found them. Then we categorized the vulnerabilities according to their severity level. We identified the most vulnerable security attack and listed them accordingly. Lastly, we compared the performance of Acunetix and Nikto That is how many attacks they actually can detect.

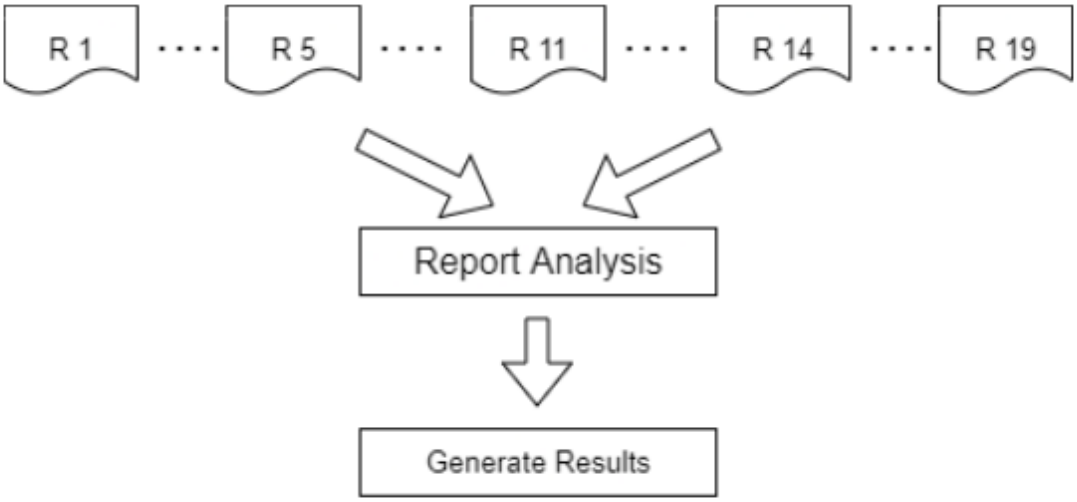


Figure 4: Report generation

7.1 Vulnerabilities and risk level analysis

The vulnerability risks are classified as High-Risk Alert level 3, Medium Risk Alert Level 2, and Low-Risk Alert Level 1, which are given and discussed below:

1. **High-Risk Alert Level 3.** Vulnerabilities are categorized as the most dangerous, which put the scan target at maximum risk for hacking and data theft

2. **Medium Risk Alert Level 2** Vulnerabilities caused by server miss configuration and site-coding flaws, which facilitate server disruption and intrusion.

3. **Low-Risk Alert Level 1** Vulnerabilities derived from lack of encryption of data traffic or directory path disclosures.

4. **Informational** Vulnerabilities that are for informational purposes

A risk-based vulnerability management methodology helps identify the organization's main risks, avoiding guessing and wasted cycles spent chasing vulnerabilities that don't move the risk needle. This allows us to make actual, significant progress in decreasing the risk profile of the websites, which means we'll spend less time attempting to uncover the types of vulnerabilities that exist in the websites and more time resolving them and keeping those in our mind for future development.

8 Experimental Findings

The following tables show the list of vulnerabilities detected by Acunetix. At the same time, we classified vulnerabilities as high, medium, low and informative. From the table, we can see that the number of vulnerabilities detected by Acunetix is larger than Nikto. It detected most of the vulnerabilities.

Website	High	Medium	Low	Informative	Total
1	1	3	8	7	19
2	0	4	2	0	6
3	0	0	1	10	11
4	0	16	14	13	43
5	0	1	1	8	10
6	33	95	2	90	220
7	8	3	8	3	22
8	0	0	3	0	3
9	0	2	1	8	11
10	0	1	2	2	5
11	0	26	1	9	36
12	0	20	10	6	36
13	17	41	16	26	100
14	0	0	2	0	2
15	25	33	4	447	509
16	0	9	14	144	167
17	0	20	3	11	34
18	1	3	1	84	89
19	0	3	1	1	5

Figure 5: List of vulnerabilities detected by Acunetix

The following tables show the list of vulnerabilities detected by Nikto. At the same time, we classified vulnerabilities as high, medium, low and informative. From the table, we can see the number of vulnerabilities detected by the Nikto is smaller. Nikto mostly detected high-level vulnerabilities.

Website	High	Medium	Low	Informative	Total
1	0	1	2	2	5
2	2	2	3	1	8
3	1	2	0	1	4
4	0	5	0	7	12
5	1	2	0	1	4
6	15	31	1	1	47
7	3	2	0	2	7
8	0	0	1	0	1
09	4	2	1	3	10
10	0	0	1	1	2
11	15	3	1	0	19
12	10	10	6	0	26
13	5	20	10	0	35
14	0	0	1	0	1
15	9	0	1	120	130
16	1	2	4	15	22
17	0	4	3	0	7
18	6	0	4	0	10
19	4	2	10	16	32

Figure 6: List of vulnerabilities detected by Nikto

The following table shows the list of vulnerabilities found on the school websites. From the results, we saw that there are fewer vulnerabilities because there are not many pages and functionality on the school websites. So they need less information and redirection. Though the websites are not maintained properly but for this reason, they have fewer vulnerabilities.

School Website	Site 1	Site 2	Site 3	Site 4	Site 5
XSS	0	1	1	0	0
Buffer Overflow	0	1	0	0	0
Csrf	1	1	1	1	1
SSL	0	10	0	0	0
SQLI	0	0	0	0	0
Clickjacking	1	1	1	2	1
Cookie vulnerability	1	1	0	0	0
Backup Files	0	2	0	0	0
Dos	0	0	0	4	0
SSN	0	0	0	0	0
Directory traversal	0	0	0	0	0
Broken Files/Link	1	0	10	4	8
Possible sensitive directories	1	1	0	2	0

Figure 7: List of vulnerabilities found in school websites

The following table shows the list of vulnerabilities found on the college websites. College website vulnerabilities are higher in number because they don't maintain their websites regularly and that can result in some sensitive information leakage which can result in major defamation of the college.

CollegeWebsite	Site 1	Site 2	Site 3	Site 4	Site 5
XSS	3	0	0	1	0
Buffer Overflow	0	0	0	1	0
Csrf	3	0	0	0	0
SSL	0	1	0	0	0
SQLI	30	1	0	1	0
Clickjacking	1	1	1	1	1
Cookie vulnerability	0	0	2	0	0
Backup Files	0	0	0	0	0
Dos	0	1	0	1	0
SSN	0	0	0	0	0
Directory traversal	0	0	0	1	0
Broken Files	28	0	0	4	1
Possible sensitive directories	0	1	0	0	2
Application error message	97	0	0	2	0

Figure 8: List of vulnerabilities found in college websites

From the table, we can see the list of vulnerabilities on the university websites. From the findings, we saw that engineering universities have a lower number of vulnerabilities. From the table, site 6-9 belongs to engineering universities . They have fewer vulnerabilities as they are technical universities so they have the resources to maintain their website regularly. On the other hand, other universities have a lot of vulnerabilities as they don't have the resources to maintain their websites regularly which makes their sensitive information really vulnerable. This can result in major information leakage and defamation.

University Website	Site 1	Site 2	Site 3	Site 4	Site 5	**Site 6**	**Site 7**	**Site 8**	**Site 9**
XSS	15	17	18	0	1	**0**	**1**	**0**	**0**
Buffer Overflow	0	0	0	0	1	**0**	**1**	**0**	**0**
Csrf	0	7	0	18	20	**1**	**1**	**0**	**4**
SSL	0	0	1	0	0	**0**	**0**	**0**	**0**
SQLI	0	0	2	0	1	**0**	**0**	**0**	**0**
Clickjacking	1	1	1	1	1	**1**	**1**	**1**	**1**
Cookie vulnerability	0	2	0	0	0	**2**	**2**	**1**	**2**
Backup Files	0	12	1	0	0	**9**	**0**	**0**	**0**
Dos	3	1	0	0	1	**0**	**1**	**0**	**1**
SSN	2	9	0	0	0	**6**	**0**	**0**	**1**
Directory traversal	0	0	0	0	1	**0**	**0**	**0**	**0**
Broken Files/Links	0	14	129	6	10	**1**	**0**	**0**	**139**
Possible sensitive directories	1	10	0	0	0	**5**	**0**	**0**	**9**
Application error message	0	4	17	0	0	**0**	**0**	**0**	**0**

Figure 9: List of vulnerabilities found in university websites

From the combined result, we saw that the most observed vulnerability is broken Files/Link. It has been detected 355 times. For this computer files that have been corrupted become dysfunctional or useless. A file can get corrupted for a variety of reasons. In certain circumstances, the damaged file can be recovered and fixed, while in others, it may be necessary to delete the file and replace it with a previously saved version. And the least observed vulnerability is a directory traversal which has been detected only 2 times. It allows attackers to enter forbidden folders and run commands outside of the web server's root directory. In the authorization procedure.

websites/ vulnerabilities	S 1	S 2	S 3	S 4	S 5	S 6	S 7	S 8	S 9	S 10	S 11	S 12	S 13	S 14	S 15	S 16	S 17	S 18	S 19	Total
XSS	0	1	1	0	0	3	0	0	1	0	15	0	17	0	18	0	0	1	1	58
Buffer Overflow	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	4
CSRF	1	1	1	1	1	3	0	0	0	0	0	1	7	0	0	4	18	20	1	59
SSL	0	10	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	12
SQLI	0	0	0	0	0	30	1	0	1	0	0	0	0	0	2	0	0	1	0	35
Clickjacking	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20
Cookie vulnerability	1	1	0	0	0	0	0	2	0	0	0	2	2	1	0	2	0	0	2	13
Backup Files	0	2	0	0	0	0	0	0	0	0	0	9	12	0	1	0	0	0	0	24
Dos	0	0	0	4	0	0	1	0	1	0	3	0	1	0	0	1	0	1	1	13
SSN	0	0	0	0	0	0	0	0	0	0	2	6	9	0	0	1	0	0	0	18
Directory traversal	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	***2***
Broken Files/Link	1	0	10	4	8	28	0	0	4	1	0	1	14	0	129	139	6	10	0	***355***
Possible sensitive directories	1	1	0	2	0	0	1	0	0	2	1	5	10	0	0	9	0	0	0	32
Application error message	0	0	0	0	0	97	0	0	2	0	0	0	4	0	17	0	0	0	0	120

Figure 10: Combined result

9 Conclusion

After performing all the scanning processes, we found that the college websites are most vulnerable to threats. School and university websites are comparatively safer. The reason for finding the college websites more vulnerable is that these websites are not maintained properly. The software developers do not properly maintain the security loopholes. The college websites involve too many important credentials. So the college websites are suitable targets for the hackers. The reason for being school and university websites less vulnerable is that school websites usually do not contain that much information. Moreover, the number of web pages on school websites is too limited. So these web applications are comparatively less vulnerable. And the university websites are maintained with proper security checks. The university website developers are pretty advanced. So the university websites are less hamper than the college websites. And after the analysis of the two security tools, Acunetix almost covers all vulnerabilities whereas Nikto mostly detects high-level vulnerabilities.

10 Future Work

In this study, we used two security scanning tools. Acunetix Nikto. Due to cost issues, we could not perform scanning with the paid tools. Moreover, there were some device-related issues. Due to low configuration, we could not implement the proper platform for scanning. But in the future, we want to scan with some paid tools as they provide a more detailed report with the security

vulnerabilities. This time our dataset was limited. We selected 5 school websites, 5 college websites, and 9 university websites. But in the future, we want to increase our dataset to get an even better and more accurate scanning report. Still many sectors in Bangladesh are under huge security vulnerability threats which are not tested at all. So our next plan is to work on the other sectors in Bangladesh.

References

[1] H. Shahriar, "Web security vulnerabilities: Challenges and solutions," *A Tutorial Proposal for ACM*, pp. 1–5, 2018.

[2] H.-C. Huang, Z.-K. Zhang, H.-W. Cheng, and S. W. Shieh, "Web application security: threats, countermeasures, and pitfalls," *Computer*, vol. 50, no. 6, pp. 81–85, 2017.

[3] A. Masood and J. Java, "Static analysis for web service security-tools & techniques for a secure development life cycle," in *2015 IEEE International Symposium on Technologies for Homeland Security (HST)*, pp. 1–6, IEEE, 2015.

[4] A. Alzahrani, A. Alqazzaz, Y. Zhu, H. Fu, and N. Almashfi, "Web application security tools analysis," in *2017 ieee 3rd international conference on big data security on cloud (bigdatasecurity), ieee international conference on high performance and smart computing (hpsc), and ieee international conference on intelligent data and security (ids)*, pp. 237–242, IEEE, 2017.

[5] "Acunetix." https://www.acunetix.com/plp/web-vulnerability-scanner/?ab=v2&gclid=Cj0KCQjw1ZeUBhDyARIsAOzAqQJpgmvQu68aECYo34vBCeFACSHR6zwtTP_qd9gpeZzz_xhuoAy-raAaAtH5EALw_wcB&utm_medium=cpc&utm_source=Adwords&utm_content=69244266647&utm_campaign=1683924377&utm_term=acunetix.

[6] "Nikto." https://cirt.net/Nikto2.

[7] M. A. Rahman, M. Amjad, B. Ahmed, and M. S. Siddik, "Analyzing web application vulnerabilities: an empirical study on e-commerce sector in bangladesh," in *Proceedings of the international conference on computing advancements*, pp. 1–6, 2020.

[8] A. Chancusi, P. Diestra, and D. Nicolalde, "Vulnerability analysis of the exposed public ips in a higher education institution," in *2020 the 10th International Conference on Communication and Network Security*, pp. 83–90, 2020.

[9] M. Akour and I. Alsmadi, "Vulnerability assessments: a case study of jordanian universities," in *2015 International Conference on Open Source Software Computing (OSSCOM)*, pp. 1–7, IEEE, 2015.