Hybrid Framework Architecture for Mitigating Vulnerability Attacks on E-Commerce Using Machine Learning

Martin Doe
Department of Computer Science, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana
warriorw112@gmail.com

Christian Bakaweri Baatuomu
Department of Computer Science, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana
bayellingchris@gmail.com

Michael Asante
Department of Computer Science, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana
michasst@yahoo.com

ABSTRACT

The use of e-commerce as a new trend for making a business accessible to anyone interested in starting their own company is on the rise. Its advantages are so great that people are spending a lot of money on Ecommerce platforms to improve the quality of their business, retain clients, and please them. The recent wave of cyber-attacks against e-commerce platforms and websites, on the other hand, presents new challenges. Existing e-commerce platforms have flaws in their structure, making them open to many types of assaults. As a result, this research devised and implemented a safe hybrid framework to combat persistent e-commerce platform threats. The architecture includes an SVM machine learning component for identifying and rejecting suspicious e-commerce system attacks, which solves vulnerability attacks. The suggested e-commerce architecture outperforms the existing ones in terms of the Open Web Application Security Project, according to performance evaluations.

Index Terms – Vulnerability, E-Commerce, Security, Cyber-Attack, Support Vector Machine.

1. INTRODUCTION

In recent times, transacting businesses via the web has become more popular and has gained more attention during this COVID-19 pandemic season. Business information becomes a more significant asset to many business institutions [1]. Data security needs to be seriously protected in an organization because data is presently an imperative resource to numerous organizations and as a result, internet users with bad intentions are targeting to access these data without permission [1]. Electronic businesses have become so much important means of engaging and maintaining good customer relationships. Given this, many such business institutions have become more targeted for unauthorized users with malicious intentions to force attacks on these business institutions as they engage in electronic business [2]. Threats and vulnerability evaluation of these electronic platforms more precisely websites could be one major way by which the security of these platforms can be enhanced. Assessment of Vulnerabilities on websites and e-commerce platforms is how security can be enhanced on them[3]. We seek to scan by doing security audit on vulnerabilities as a tool to enhance security challenges as threats and vulnerabilities are identified and propose many effective solutions to improve the security of online or electronic commerce platforms. As the electronic business transaction is becoming the most popular business model of the day and more users (customers, employees) are becoming frequent users of these platforms, it is significant to know how vulnerable this platform is. Hence, this study seeks to find out if there are any level of security weaknesses on electronic commerce websites? To what extent do the security vulnerabilities pose a threat to electronic commerce websites? And also find out if there are ways by which attacks are being initiated on electronic commerce websites? What will be the effect on the electronic commerce websites after using the proposed model to mitigate the identified vulnerabilities? These are the major mind blogging issues that this research study intends to find solutions to. Currently, e-commerce platform clients face a significant barrier in detecting and anticipating illegal login attempts. Again, due to covid-19 widespread, there's an advocate for more utilization of e-marketing instead of physical or face to face marketing and since many of these online marketers are going to compete with online attackers, there is justification to critically examine and propose security vulnerability safeguards for these users. Having been painted with this gap, a novel solution that examines the security weaknesses on electronic commerce websites assesses the security vulnerabilities, that could pose a threat to electronic commerce, and propose a vulnerability assessment model to mitigate those threats and vulnerabilities in electronic commerce websites is highly recommended [4]. In other to promote a healthy computing atmosphere, there is a need for computer security enhancement. A computer-based crime is on the rise all over the world and organizations and business firms need to secure their frameworks against those assaults [5]. The spite of the fact that several specialized technical measures are in place, a few organizations are likely to encounter security challenges and breaches. The cyber security breaches of the 2019 study detailed that most business organizations are exposed to cyber security dangers, as well as their utilization of cloud computing [6].

2. RELATED WORK

Internet technologies, in line with major advances over the past few decades, and advances in exploiting the dangers associates with computer systems have also dramatically been on the rise. Utilization is perpetrated by mischievous hackers who discover themselves weaknesses or vulnerabilities that are already existing faults within the computer systems that offer security [7].

The most typical examples of vulnerabilities are flaws in the construction or design of network frameworks, agreements, media, operating systems, web-based applications, and related services and databases. The threat may be a potential threat to security breaches and could cause damage and potential disruption to information or service stored or provided internally or by computer programs or communications links. Computer system threats occur when privacy (in this case preventing unapproved parties from having access), reliability (not changed without permission), and access (easily accessible at the request of approved individuals) to data on systems are compromised. Therefore, the threat of a computer program can generally involve anything intentionally, unintentionally, or caused by a natural disaster involving loss of information deception, or physical demolishing of hardware.

Similarly, the dangers to computerized systems can be categorized as well as physical and non-physical threats Physical data threats cause a system's hardware to fail or a hard disk that holds sensitive information. Nonphysical threats identify software and information in computer programs by tampering with data or taking advantage of software flaws. Exploitation that is successful results in security breaches in computer system assaults [7]. Therefore, a threat or a danger is a potential risk caused by susceptibility in a system. On the other hand, an attack is an attempt by an unapproved entity to take a harmful action or a harmful act. So basically, an attack is often the presence of danger.

Open Web Application Security Project (OWASP) has listed ten security vulnerabilities as the most serious security hazards related to web applications [8]. These threats are well-known types of risks. Aside from the fact that they are well-known to be exploitable and, when used, have a detrimental influence on websites, resulting in their ranking as the top ten (10 WASP, 2017) [9]. The following are OWASP's top ten risks:

2.1. Top Ten Security Vulnerability of OWASP

1.SQL Injection

2. Broken_authentication

3.Sensitive_Data_Exposure

4. XML_External_Entities (XXE)

5. Broken_Access_Control

6. Security_Misconfiguration

7. Cross_Site_cripting (XSS)

8. Insecure_Deserialization

9. Using_Components_with known_vulnerabilities

10. Insufficient_Logging & Monitoring

2.2 Attacks on Hardware

Day in and day out, computer users face attacks online; be it a malware attack or an attack on a hardware component. The primary difference between hardware and the severity of attacks caused by hardware tools is determined by software attacks. Unlike software attacks, which can be carried out by simply downloading an online vulnerability tool, the hardware attack threshold is raised since any authorized person who wishes to gain access or attack the hardware must have a full understanding of the hardware [10].

Denial of service attacks (DOS) is one commonest threats or attacks on the web which prevent a rightful user of a system from accessing a resource from such a system. In most cases, the hardware resources and the network are shut down or made inaccessible for a legitimate customer or user of such hardware and the network resources [11].

When a user of a system whose privileges are low (privileges with restrictions) tries to use vulnerabilities in the system to change his or her privileges to high-level privileges, it allows unauthorized users to get access to restricted data that are critical and sensitive [12].

Legitimate privileges Abuse occurs when a legitimate user like a database administrator is given a high-level privilege due to his role and the nature of his work, in a situation where such a database administrator is trying to gain more access control in an aspect of the system which does not bother him rather what has been given to him [13].

Platform vulnerabilities occur whenever there are any vulnerabilities in any of the operating systems such as Linux, Windows 2000, Windows XP, etc., it can cause the available data at rest or in motion to get corrupt. It can also increase privilege and denial of service and unauthorized data access security threat. [14].

The attackers can have full control over the database of a system. This can be done by inserting an unauthorized SQL script into the database query. This can be done in a form of modifying cookies fields in such a way that they can contain malicious strings. Headers can also be modified to contain strings that can cause harm through the variables of the service. Malicious scripts(codes) can also be injected into the user input forms on the web [14].

Weak authentication can result in data theft if how to access data from any database is so weak, unauthorized users can easily get the identity of privileged users through the use of brute force, social engineering, and direct credit theft technologies. To secure a system’s backend, two user authentication needs to be done [15].

The hackers’ motivation for attacking web-based systems in most cases, hackers aim to destroy computer securities and networks that provide privacy, integrity, and access to information or services on systems. This type of hacking activity is termed illegal as they spend a lot of their time just knowing how they can discover and break any security system over the network. Taking a glance at the taxonomic collection of computer system threats, there lies the need to differentiate between diverse forms of hackers. Any category of hacker is likely to have a motive for the various actions they initiate [16].

[17] in their study sought to use blockchain algorithms to avoid SQL injection by using IP node verification. The study sort to figure out how hackers get legitimate and privileged access to databases unlawfully. The blockchain algorithm used helped to prevent illegal access to the database of web applications but the blockchain algorithm couldn’t check the node-to-node signature verification.

[18] proposed a single VAPT with Raspberry Pi 3b+ to analyze systems and make them secure to achieve security on networks by mitigating the identifying threats. This methodology helped to enhance and strengthen online safety but the use of the single VAPT built with Raspberry Pi 3b+ alone cannot resolve the vulnerability problems entirely.

[19] their study carried out a research flow with OWASP 10 level to carry out penetration testing to evaluate and test the security level of the web together using six sub-domains to determine and evaluate a website's security level. Though the approach was able to give severity rates for the various vulnerabilities, scientifically, the study couldn’t mitigate the vulnerabilities.

[20] Propose Secure code PHP functions to show and stop XSS attacks check the validity of data from the web, and check and protect all input forms. A vulnerable PHP website was initiated to evaluate the proposed system's efficiency. The methodology couldn’t automate the plugin for the browser that will automatically identify and stop harmful code from operating in the online form. [21] Proposed an IOT-based system to protect user’s data in a health-based application system and the system is to demonstrate how the attacks are initiated. The approach was able to show how attackers get access to users’ data without authorization but the more the devices with IOTs are deployed, the greater the system becomes more vulnerable to the attackers again.

3. METHOD AND PROPOSED MODELLING

3.1 The Vulnerability Assessment Process

A vulnerability assessment was performed on three major popular electronic commerce sites in Ghana. Because of security reasons, IP addresses, URL addresses, and domain names of these popular electronic commerce sites have been excluded from this paper, rather, aliases were used.

The testing was done via the Netsparker, a Web application Scanning platform from the cloud, and the Black Box Testing scope was employed.

3.2 Information Gathering and Vulnerability Detection

The Scanner field was used to choose the location the scanners should run from, and the target field is the target eCommerce platform's hostname.

The scanner was set to crawl all URLs discovered in the Scope setting. This ensures that all possible URL endpoints are checked for vulnerabilities, except for any URL recognized to let a user log out from the e-commerce platform, which is any URL that contains the string "logout." There was also a type-based exclusion of files.

The scanners were set to audit all of the following items: cookies, headers, forms, links, parameter names and values, JSON, XML, and DOM elements, hence an Extensive Scan type was chosen. The maximum scan time was set to eight (8) hours, with a maximum number of crawled and browsed URLs of 10,000. Following the setting, the launch button was pressed to begin the scans for each e-commerce platform.

The platform has three applications that address specific security needs (Netsparker.com). They are

1. Vulnerability Management
2. Container Security
3. Web Application Scanning

3.3 Preliminary Scanned Results on Targeted E-Commerce Platforms

To secure e-commerce platforms' vulnerabilities with a proposed hybrid framework architecture using machine learning, the selected electronic platforms were gray hacked by using Netsparker.com; a vulnerability scanner that was to reveal the secure state of the various e-commerce platforms. This method was used to detect the presence and levels of vulnerabilities on the respective e-commerce platforms. The vulnerabilities found on the various e-commerce platforms were categorized into severity summary and OWASP Top 10 vulnerabilities categories.

Table 3.1 is a severity summary of the various kinds of vulnerabilities that exist in the scanned e-commerce websites, indicating the amount of LOW, MEDIUM, HIGH and CRITICAL vulnerabilities respectively. Low vulnerabilities may not have a higher impact on the e-commerce platforms, whiles MEDIUM vulnerabilities could affect the e-commerce platforms with a minimum impact, HIGH and CRITICAL vulnerabilities are a serious threat to the e-commerce system upon which the proposed mitigating architecture with machine learning was needed. One High or Critical vulnerability is enough to bring down a whole e-commerce site.

Table 3.1: Scanned Results from the Targeted Host

HOST	LOW	MEDIUM	HIGH	CRITICAL
ALL HOST	19	9	2	3
Host T	5	3	1	1
Host J	2	2	0	1
Host F	12	4	1	1

3.4 Top Ten Security Vulnerability of OWASP

Open web security application protocol (OWASP) was the standard used in measuring safe and secured web applications by checking the top 10 vulnerabilities against any web application that is seeking to be secured. These vulnerabilities often provide attackers unapproved access to some information being targeted. Rarely, do such weaknesses cause full system compromises.

However, it was detected that high levels and critical vulnerabilities were recorded on at least one of the e-commerce platforms scanned. The vulnerability assessment program netsparker.com, which was employed in the evaluation, produced the following results, along with proposed temporary solutions to assist reduce the high and critical levels of vulnerabilities discovered on each host of the e-commerce website environment. The findings were categorized using the OWASP Top 10 vulnerabilities, which gives a set of criteria for a better understanding of web application security risk.

Table 3.2 is the categorization of the summary of the vulnerabilities found in all three scanned e-commerce websites, this result has been further categorized in the OWASP top 10 vulnerability categories and grouped by each e-commerce platform scanned, this categorization is to help understand which of the vulnerability needs more attention and mitigation model to mitigate the activities of hackers and attackers. It could be seen from table 3.2 that injection and cross-site scripting which is critical vulnerabilities were present in at least one of the e-commerce sites and therefore will need a better mitigating approach.

Table 3.2: Scanned Results, OWASP Top 10 Categorized from the Targeted Host

	Vulnerability by Category	Vulnerabilities Count			
SN.	OWASP TOP 10 Vulnerabilities	Host T	Host J	Host F	TOTAL
1	Injection	2	2	3	7
2	Broken_Authentication	1	1	1	3
3	Sensitive_Data_Exposure	1	0	0	1
4	XML_External_Entities (XXE)	0	0	1	1
5	Broken_Access_Control	1	0	3	4
6	Security_Misconfiguration	3	1	6	10
7	Cross_Site_Scripting (XSS)	0	0	2	2
8	Insecure_Deserialization	0	0	0	0
9	Using_Components_with Known_Vulnerabilities	1	0	1	2
10	Insufficient_logging_and_Monitoring	1	1	1	3

3.4.1 Proposed Hybrid Framework Architecture for Preventing Ecommerce Vulnerability Attacks

The proposed system seeks to prevent vulnerabilities on e-commerce platforms. Figure 3.1 shows how an attack on the e-commerce site can be launched through an attack vector and how an attack could be detected and rejected if found malicious. From figure 3.1, the attacker initiated an attack through an attack vector. The attack vector launched an attack as SQLiA or XSS which is passed through the URL segregator and if the input variable string is SQLiA or XSS; then, it will segregate SQLiA string into ARGS, URL, BODY, and HEADER and then compare against security hooks rules. The input is compared against the security rule hook database in the security hook web database system and records legitimate logs and attack vector Logs. If the log is legitimate, then "The outcome of the classification is negative, which means that the URL string does not include any potential SQLiA or XSS code. However, If SQLiA attacks, then "The classifier SVM forwards the URL string through the precise security rules detection engine for separation into suspicious and normal code. If detection is matched, then the "Detection engine uses a pattern matching algorithm to audit the URL content. In a case where no rule matches in the detection engine, then the next attack vector is loaded until the last attack vector. Nevertheless, if the rule matches in the detection engine, then the HTTP request is rejected, and store vulnerable logs in the vulnerability DB Log. The detection engine will stop passing all requests through the detection engine of the proposed system.

3.5 Proposed Attack Trace Collection Framework

Figure 3.2 shows a proposed Attack Trace Collection Framework that collects attack logs from e-commerce sites. This proposed framework is to help collect attack logs through the vulnerable web application model through the e-commerce website. From figure 3.2 an attack vector is launched through SQLiA and XSS as a request to the webserver model, attack vector logs are collected through the weblog script model through the vulnerable web applications, and all vulnerable web DB stores the vulnerable logs. This framework is fit into the main framework in collecting attack vector logs into the dataset of the main framework.

3.6 Proposed Machine Learning approach

The supervised machine learning method known as SVM can be applied to classification and regression issues. Accuracy and efficacy of optimal classification algorithms, Support Vector Machine (SVM) was used or deployed to analyze, train, test, and classify the dataset collected through the proposed attack trace collection framework.

Figure 3.3 shows how the SVM is used in classifying the dataset into sub-categories of Class A and Class B, where Class A is considered as the normal code and Class B as the malicious code of the attack vector such as SQLiA or XSS. The SVM is used in the framework to help classify incoming attack vector codes. In figure 3.3 The margin, or the separation between marginal hyperplanes, is equal to $\frac{2}{||w||}$. In any practice cases that touch a hyperplane H_1or H_2, the sides defining the margin, as shown in Figure, are support vectors. 3.3.

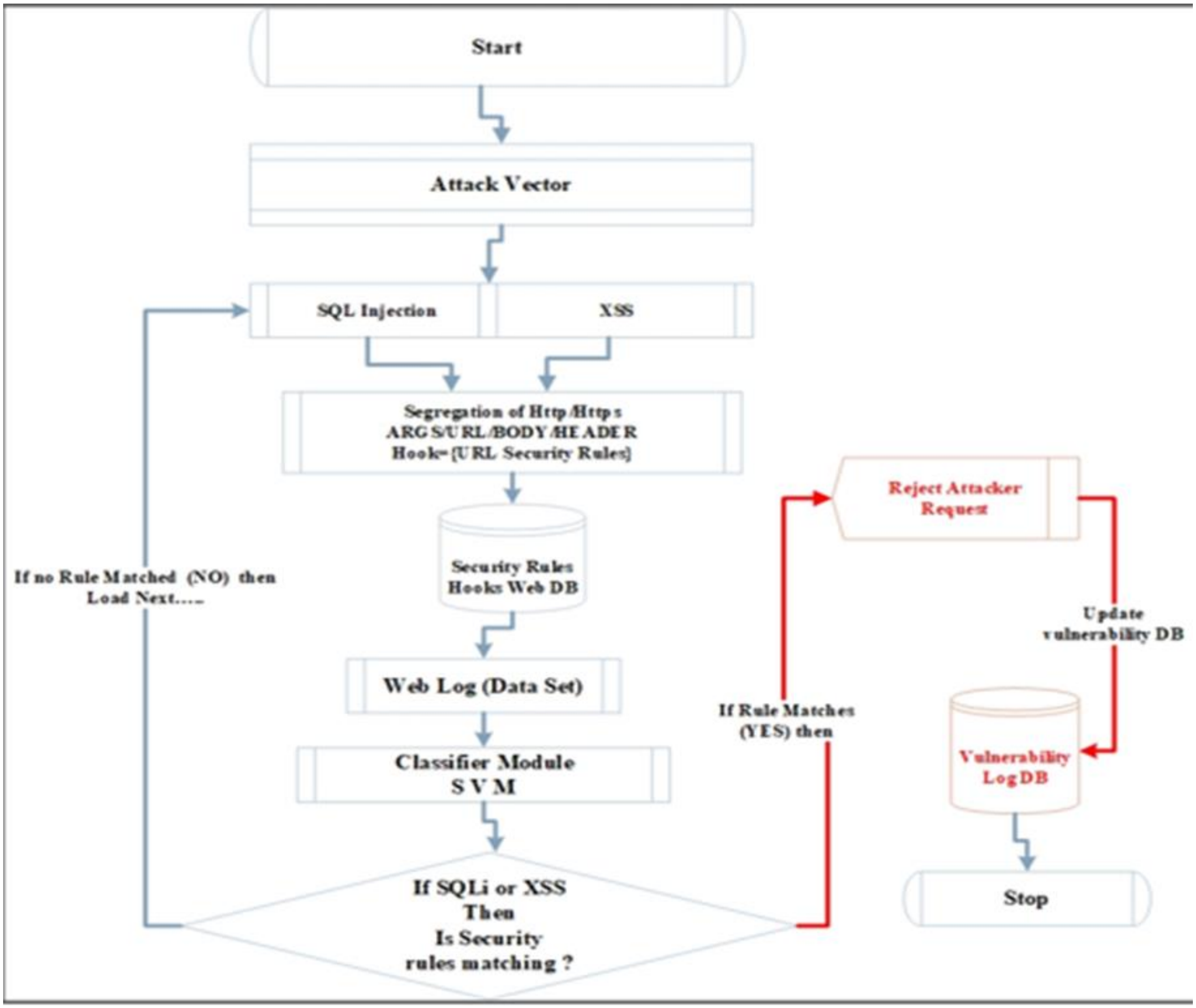


Figure 3.1 Proposed System Architecture Framework for Preventing E-Commerce Vulnerability Attacks

3.7 SVM Mathematical Machine Learning Algorithm

On both standard text classification jobs and script code, SVM has performed admirably. SVM is often used to separate linear data, but it can also be used to separate nonlinear data [22]. Because it is not always possible to distinguish between classes of dangerous code and benign code using linear separation, this work uses SVM for nonlinear separation. The classifier calculates each training data set's vectors in feature space, as well as the element λ , It establishes the hyperplane that can classify training data. The classifier uses the discriminant function *f(x)* to classify test data during the classification process [23].

$$f(x) = sign(g(x), g(x) = w\phi(x) + b \; \dots \dots \dots \dots (1)$$

x is a vector that is used to tokenize training data into x_i *(i =1,…,n)*. ϕ tokens. is the function that transforms x into another feature space. The parameters that determine $f(x)$ are the vector w and the scalar b Then we'll assume:

$$g(x_i) = w\phi(x_i) + b \begin{cases} \geq 1x_i \in c_1 \\ \leq 1x_i \in c_2 \end{cases} \dots \dots \dots .. (2)$$

$$y_i = \begin{cases} 1x_i \in c_1 \\ -1x_i \in c_2 \end{cases} \dots \dots \dots \dots .. (3)$$

code that isn't malicious or code that isn't normal on a hyperplane, we assume $g(x) = 0$, so *SV* satisfies $y_i = (w(x) + b) - 1 = 0$. To make a partial distinction between w and b. $K(i, j)$ is a kernel operation that allows you to determine the inner product without having to calculate the outer product $\phi(x)$. w and b are also given by:

$g(x) = \sum_{i=1}^{n} \lambda_i \, y_i \, K(x_i, x) + b$ .... (4)

$b = y_s - \sum_{i=1}^{n} \lambda_i \, y_i \, K(x_i, x_s)$ .... (5)

Any *SV* is represented by x_s. When i is found analytically in this study, the constraint condition determines the best j. As a result, $g(x)$ is determined, and test data is categorized by $g(x)$.

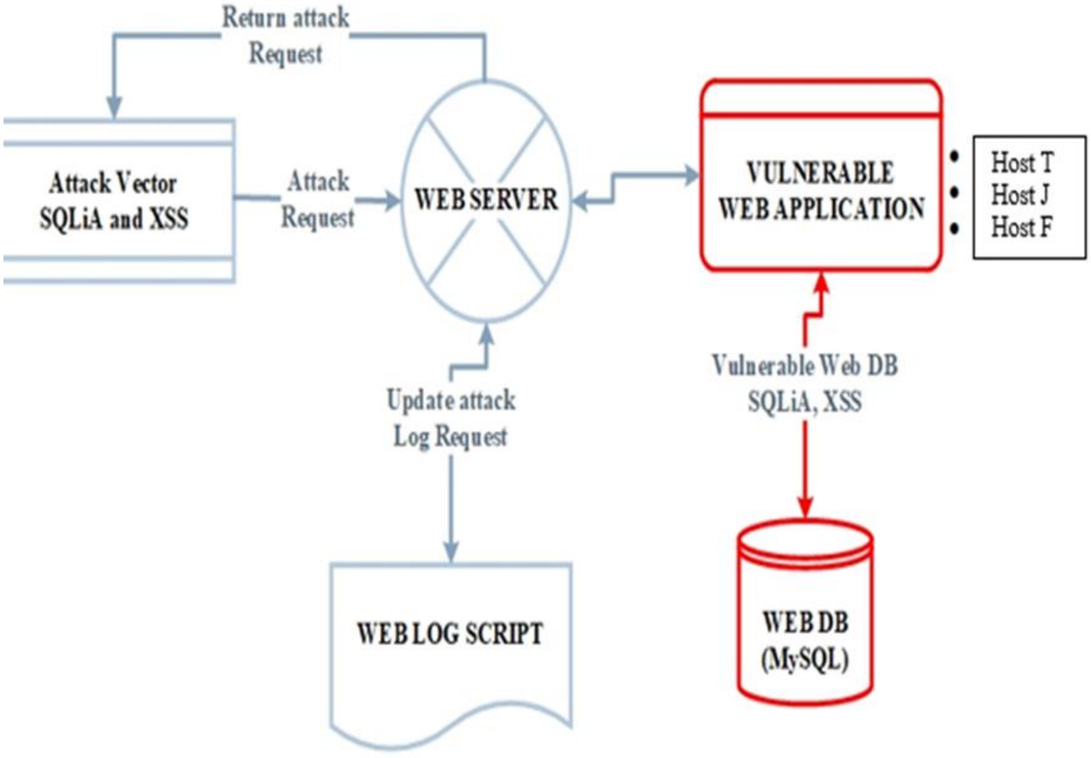

Return attack Request
Attack Vector SQLiA and XSS
Attack Request
WEB SERVER
VULNERABLE WEB APPLICATION
Host T
Host J
Host F
Update attack Log Request
Vulnerable Web DB SQLiA, XSS
WEB LOG SCRIPT
WEB DB (MySQL)

Figure 3.2: Attack Vector Trace Collection of Datasets

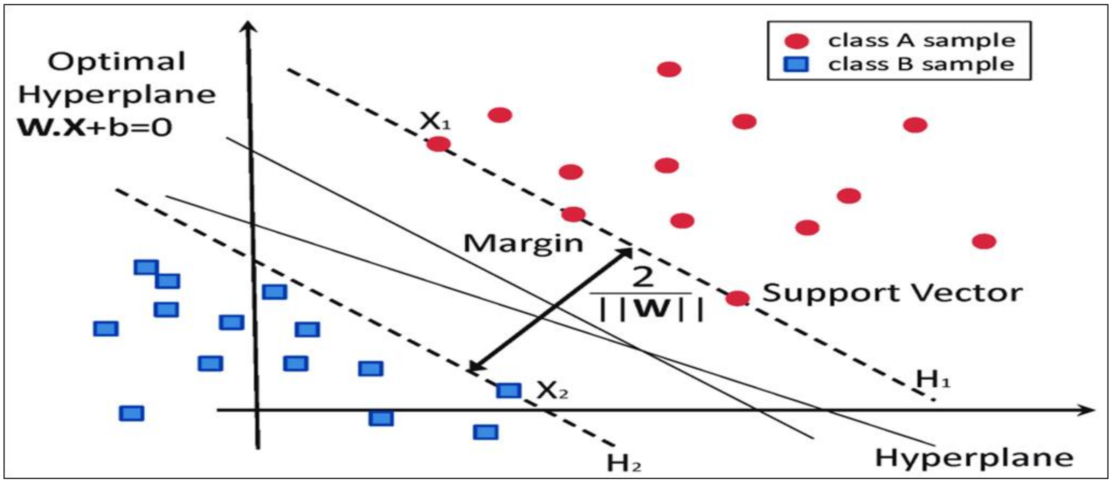


Figure 3.3: SVM Data Classification Diagram Source: [24]

3.8 The Quantum of Training and Test Data for SQLiA and XSS

For any robust machine learning model to be effective and result oriented, it needs to be trained and get tested with some amount of data. Given this, table 3.3 shows the amount of dataset used as normal code and malicious code. Table 3.3 indicated the amount of dataset used as training data and testing data and the output of this is shown and narrated in table 3.3.

Table 3.3: The Quantum of Training and Test Data for SQLiA and XSS

Items	Training Data	Test Data
Normal Code	9885	13686
Malicious Code	3801	12801

3.9 SQLiA Query String, Detection Technique, and Feature Engineering

To be able to train the proposed model to detect, feature engineer and mitigate the vulnerabilities using machine learning, a form of SQL injection attack's SQL scripts were used as illustrated in table 3.4. This implies that anytime this token: (select * from admin where uid = "1";) is passed, the system should recognize it as an original query else if a token in a form of (select * from admin where uid = " „ OR 1 = 1; --„,), it should be recognized as a suspicious query as shown in table 3.4 hence serving as a hybrid architecture for detecting and mitigating vulnerabilities on the e-commerce.

Table 3.4: SQLiA Query String, Detection Technique, and Feature Engineering

Name	Tokens
Original_Query	select * from admin where uid = "1";
Suspicious_Query	select * from admin where uid = " „ OR 1 = 1; --„
"O"	Original query
„S"	Suspicious query

3.10 SQLiA Query String, Detection Technique, and Token Rule

In this case, the original query is passed and a dangerous or malicious query is blocked. The Word list encapsulates the tokens of SQL-query strings O indicating the original SQLi query and S indicating suspicious SQLiA query.

Table 3.5 shows SQL injection attack query strings, their detection techniques, and token rules.

Table 3.5: SQLiA Query String, Detection Technique, and Token Rule

Name	Tokens
"O"	select * from admin where uid = "1";
"S"	select * from admin where uid = " „ OR 1 = 1; --„
("O")	select * from admin where uid ="1" && pwd = "abc";
("S")	select * from admin where uid = „ „ OR 1=1; --„

3.11 SQLiA Query String, Detection Technique, and Token Key Mapping

The Word list contains various tokens named *i*1… *i*12, listed in table 3.6. The string is produced as a vector, furthermore, the classifier categorizes the Orignal and suspicious SQLiA query string or XSS script.

Table 3.6: XSS String, Detection Technique, and Token Rule

Name	Tokens
“O"	<script>alert (” XSS”) </script>;
“S"	{script alert, alert XSS, XSS script, …}
(“O")	{script alert XSS, alert XSS script, …}
(“S")	{script alert XSS script, …}

4. DISCUSSION AND FINDING

4.1 Trend Of Vulnerabilities Found On All Targeted Host

The scan was conducted to ascertain the trends and levels of vulnerabilities on the targeted e-commerce hosts. It can be seen as presented in Fig 4.1 that the scanned results for the target e-commerce platforms have shown the various levels of vulnerabilities as Low, Medium, High and Critical. Even though all targeted e-commerce platforms had different sets and levels of vulnerabilities, they all have high and critical vulnerabilities which makes the platform very vulnerable to attackers. One critical vulnerability is enough to bring down a whole site through an exploit of an attacker. It could also be seen clearly in Figure 4.1 showing high levels of vulnerabilities depicting the various targeted e-commerce platform showing both low, medium, high, and critical vulnerabilities on all the scanned sites.

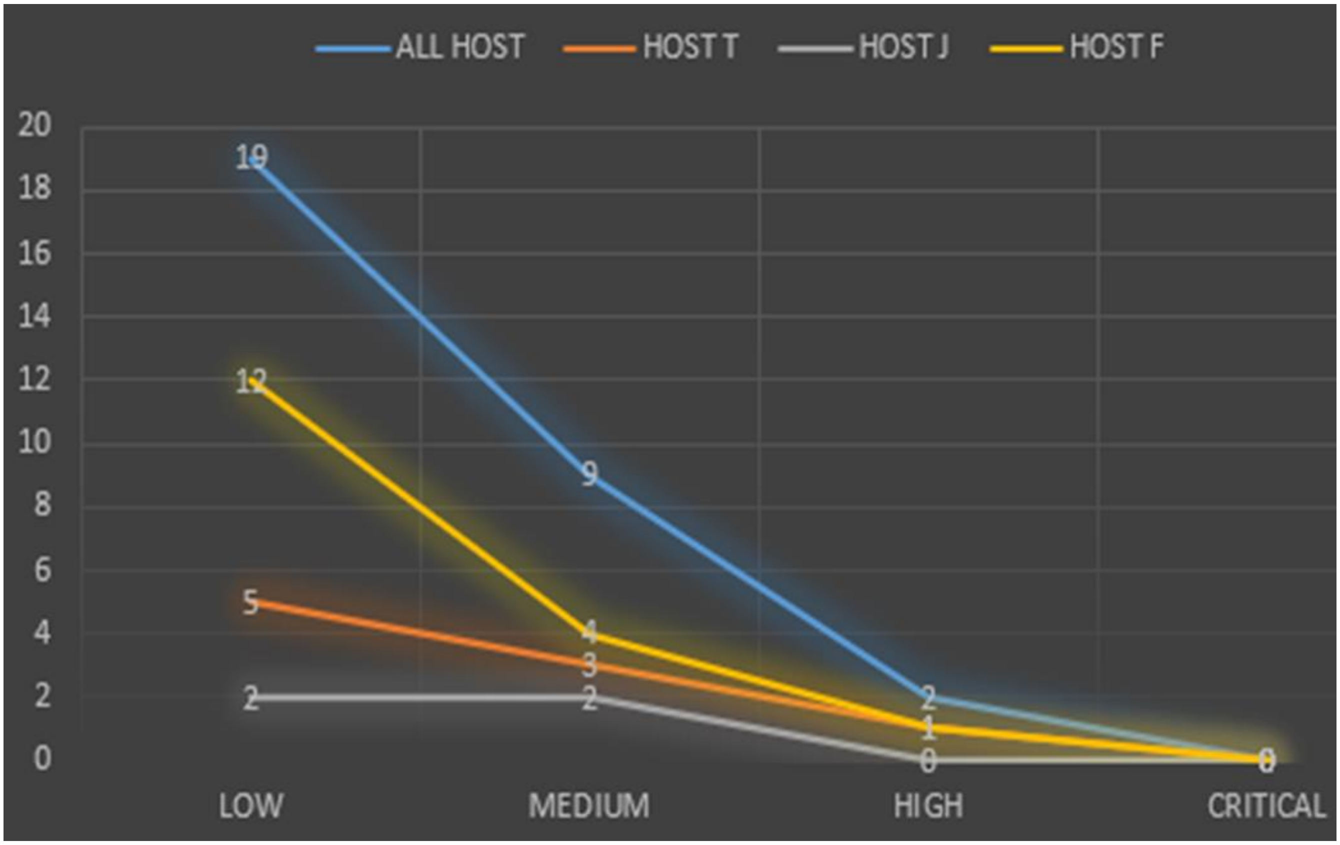


Figure 4.1: Severity Summary of Vulnerabilities Found on All Targeted Hosts

It is shown that Host T has 5 vulnerabilities to be low, 3 medium vulnerabilities identified, 1 high vulnerability identified, and 1 critical vulnerability identified. From the same Fig 4.2, Host J has been identified with 2 vulnerabilities to be low, 2 security vulnerabilities identified as a medium, and 1 critical security vulnerability identified and there wasn't any high vulnerability found on Host J. On the other hand, as shown in Fig 4.2, there were 12 security vulnerabilities found to be low on Franko, 4 identified security vulnerabilities recorded as the medium on Host F, 1 security vulnerability found and classified as high and 1 critical security vulnerability found on Host F. In the course of summarizing the totality of all the recorded vulnerabilities on all the three domains, it was identified from Fig 4.2 that all the low vulnerabilities summed up to 19, 9 medium vulnerabilities, 2 total counts of high-security vulnerability found and 3 critical counts of security vulnerability found. Since at least each of the targeted hosts or domains has a record of high and critical security vulnerabilities, there is a need to arrest that situation with a hybrid framework architecture with a machine learning algorithm.

Scanned results from targeted hosts were categorized according to OWASP's top 10 security vulnerabilities. As shown in figure 4.2, Host T has recorded 2 counts of SQL injection as one of the top 10 security flaws according to OWASP, 1 count of broken authentication security vulnerability, 1 count of sensitive data exposure security vulnerability, 1 count of broken access control security vulnerability, 3 counts of security misconfiguration as security vulnerability according to OWASP top 10 categorizations, 1 count of using elements that have a known vulnerability, 1 count of insufficient logging and monitoring. From figure 4.2, Host T has not recorded any security vulnerability count for cross-site scripting (XSS), XML external entities (XXE), and unsafe deserialization.

On the part of Jumia as clearly depicted in figure 4.2, the result showed that Host J has recorded 2 counts of SQL injection security vulnerabilities, 1 count of broken authentication security vulnerability, 1 count of security misconfiguration, and 1 count of insufficient logging and monitoring. From the record, as shown in figure 4.2, Host J has not encountered the following security vulnerabilities as categorized by OWASP top 10: Sensitive Data Exposure, XML External Entities (XXE), Defective Access Control, Cross-Site Scripting (XSS), Insecure Deserialization, and Utilizing Components with Known Vulnerabilities are all security issues.

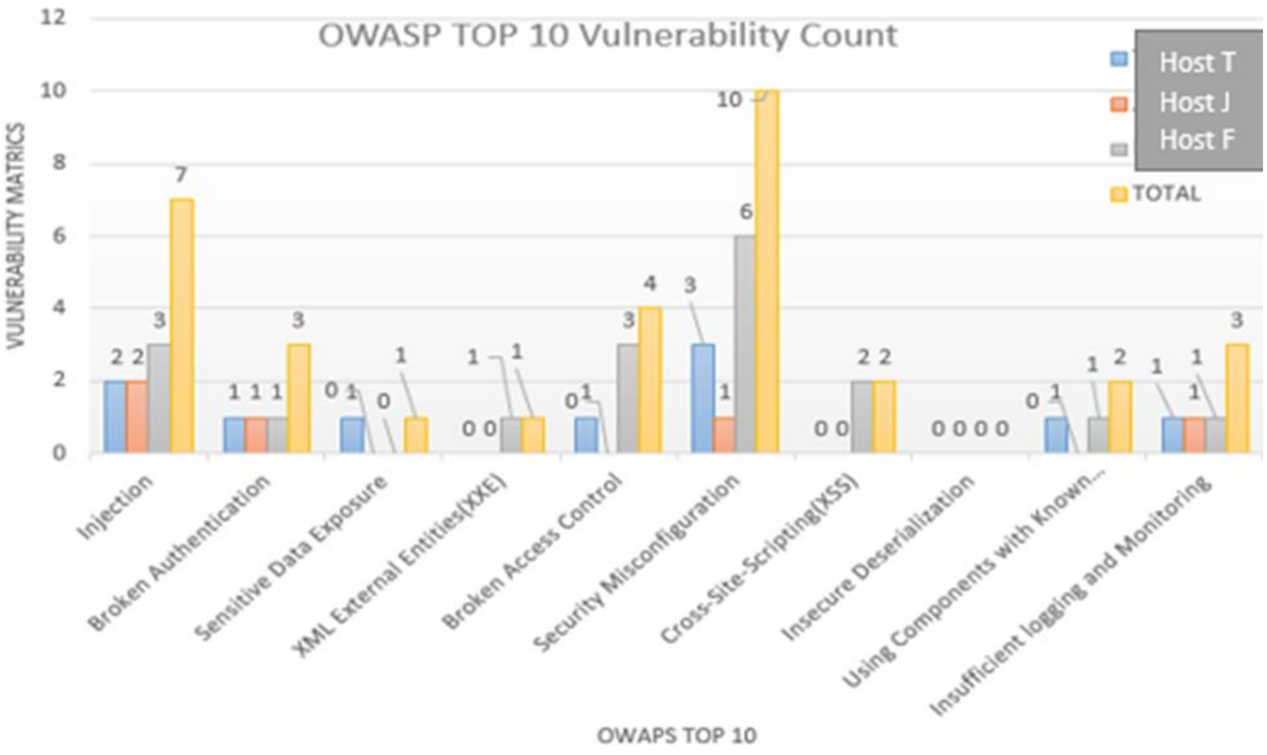


Figure 4.2: Categorized Vulnerability Results Based on OWASP top 10 Level

From figure 4.2, it is shown that Host F has 3 counts of SQL injection, 1 broken authentication security vulnerability, 1 XML External Entities (XXE), 3 counts of broken access control, 6 counts of security misconfiguration, 2 counts of Cross-Site Scripting (XSS), 1 count for both Utilizing components that have known security flaws, inadequate logging, monitoring without a record of sensitive data exposure, and unsafe deserialization.

It has been summarized as shown in figure 4.2 that a total of 7 security vulnerabilities counts were recorded for SQL injection, 3 for broken authentication, 1 for each, both Sensitive_Data_Exposure and XML_External_Entities (XXE), 4 for Broken_Access_Control, 10 for Security Misconfiguration, 2 for Cross_Site_Scripting (XSS), 2 for Using Components with Known Vulnerabilities, 3 for Insufficient logging and Monitoring with 0 vulnerability recorded for Insecure Deserialization. With all these levels of vulnerabilities, there is the need to use the proposed hybrid framework architecture using machine learning to mitigate these vulnerabilities.

4.3 Machine Learning Testing And Results

SQL-query string dataset was used to test the proposed approach. A fictitious dataset has been constructed for testing purposes. The dataset was supplied with Original SQL-query string(“O”) and Suspicious SQL-query string(“S”) entries and tested, with the findings displayed in Tables 4.1 and 4.2. The average of 100 searches of the original SQL-query string and 100 queries of the suspected SQL-injection query string is used to compute the detection time (in seconds).

A summary run information of training and test dataset of 13686 instances and 4 attributes: Serial, XSS, SQLiA, and Label with a run time to build the model: 19.58 seconds on LibSVM classifier algorithm. The use of the hybrid framework architecture for mitigating vulnerabilities attacks was an effective solution to identified vulnerabilities since a percentage of 72.2271 was correctly classified in detecting both original (O) and suspicious (S) SQL-query strings.

4.4 Run Information

The following is information and configuration of the classification processes involved during the setting up of the SVM Library configuration in python using anaconda.

Scheme: anaconda. classifiers. functions. LibSVM_S 0_K 2_D 3_G 0.0_R 0.0_N 0.5-M 40.0_C 1.0. which is the LibSVM version used. Relation: MARTIN_ML_O_S. which is the dataset file name, with Instances of 13686 the total number of instances used for the classification process, Attributes of: 4 namely: Serial, XSS, SQLiA, Label. The Test mode:10-fold cross-validation with Classifier_Model (full training set). Original code by Yasser EL_Manzalawy (= WLSVM), LibSVM_wrapper, and 19.58 seconds were needed to create the model.

4.5 Libsvm Algorithm Runs on the Dataset with Detailed Summary Results

Table 4.1 shows a summary classification results from the dataset supplied with Original SQL-query string (“O”) and Suspicious SQL-query string(“S”). it can be seen clearly from table 4.1 that the total number of instances to be classified as 13686 with correctly classified instances of 9885 given a percentage of 72.2271 as correctly classified in detecting both original (O) and suspicious (S) SQL-query string. The incorrectly classified instances of 3801 were given a percentage of 27.7729. whilst the Kappa statistics are given as 0.527. Another measurement for accuracy is the kappa coefficient. It functions as a contrast between the outcomes of segmentation and values selected randomly. Those values it permits range from 0 to 1. If the kappa coefficient is 0, there's no agreement between both the reference image and the classified image.

The mean absolute error of the classification is 0.2777 and the root means the squared error is given as 0.527. A relative absolute error of the classification was 55.881% and a root relative squared error of 105.7176%. Since the incorrectly classified instances of 3801 has a percentage of 27.7729 against correctly classified instances of 9885 given a percentage of 72.2271, it can be inferred that implementing the proposed framework architecture will help in mitigating the vulnerabilities attacks on e-commerce with machine learning.

Table 4.1: LibSVM Algorithm Run on the Dataset with Detailed Summary Results

Stratified cross-validation Summary Report			
No.	Item	Instance	Percentage
1	incidents categorized properly	9885	72.2271 %
2	incidents categorized Improperly	3801	27.7729 %
3	Kappa_Statistics	0.4391	
4	Absolute_Mean_Error	0.2777	
5	Root_Mean_Squared_Error	0.527	
6	Absolute_Relative_Error	55.881 %	
7	Root_Relative_Squared_Error	105.7176 %	
8	Total No. of occurrences	13686	

Table 4.2 show the efficacy of the proposed system of its accuracy in classifying both original (O) and suspicious (S) SQLiA query. The weighted average was also taken into consideration in using the machine learning approach with the SVM algorithm. The following results in table 4.2 show clearly the performance of the SVM algorithm and its efficiency in detecting vulnerabilities and rejecting them. The formula for the True Positive Rate (TPR), also known as sensitivity, is TP/TP+FN. The TPR measures the likelihood that a real positive will show up as both unique and suspicious. To determine the rate of false negatives, divide the rate of false positives by (c + d). This is known as the false positive rate (TPR). By dividing the total number of instances of an outcome's presence (a + b) by the number of false-negative test results for the outcome (b), the Rate of false negatives is equal to (b / (a + b)). The accuracy is shown in the table. By dividing the total number of true positives by the sum of true positives and false positives, precision is obtained. in table 4.2. Recall, another name for the true positive rate, is the proportion of true positives that are anticipated among all the positives in a dataset. It also goes by the label of sensitivity. The measure is computed using the following formula: recall=TP/(TP+FN)

A model's accuracy may be assessed using an f-score of the F-Measure metric based on precision and recall. By changing a value in the F1-score, a general case F-score, the F-score may be changed. The F-score of a model serves as a gauge of its accuracy. The accuracy of the F-score increases with its value. The F-score of a model serves as a gauge of its accuracy. A lower F-score indicates less accuracy, and table 4.2 makes clear that the F-Measure has increased reasonably. A measure for measuring how well a classifier model performs is the Receiver Operating Characteristics Curve or ROC curve. is indicated as 0.719 for both the original, suspicious and weighted averages. As illustrated in table 4.2, the precision-recall curve is a visualization of the precision (y-axis) and recall (x-axis) for various thresholds. With a PRC of 0.629 for both original, suspicious and weighted averages. While requiring the sum of distances between different locations to be larger than 1, the Mahalanobis Metric for Clustering (MMC) minimizes the total of squared distances between comparable points. It can be seen in table 4.2 as 0.440.

Table 4.2 LibSVM Algorithm Summary Results on Accuracy by Class

DETAILED ACCURACY BY CLASS								
TPR	FPR	PRECISION	RECALL	F-MEASURE	MCC	ROC	PRC	CLASS
0.674	0.236	0.710	0.764	0.691	0.440	0.719	0.629	Original
0.674	0.326	0.732	0.764	0.748	0.440	0.719	0.686	Suspicious
0.722	0.285	0.722	0.722	0.722	0.440	0.719	0.660	Weighted_Avg.

4.5 Confusion Matrix Results

The Confusion Matrix Results of A and B are classified as 4252 2061 | A = ORIGNAL, 1740 5633 | B = SUSPICIOUS. In a confusion matrix, the ratio of accurate guesses to inaccurate predictions is shown. For a binary classifier, this would be the ratio of the true negatives and true positives (accurate predictions) to the false negatives and false positives (incorrect predictions). As can be seen in table 4.3, the Confusion Matrix Results of A and B are classified as 4252 2061 | A = ORIGNAL and 1740 5633 | B = SUSPICIOUS.

Table 4.3: Confusion Matrix Result

Confused Matrix Results	
a b ← classified as	
42522061	a=ORIGINAL
17405633	b= SUSPICIOUS

4.6 Classifier Visualize: Threshold Curve Of Original Roc

Plot area under the receiver operating characteristic (ROC) = 0.7188 (Class Value for ORIGINAL). As shown in figure 4.4, a plot of Y-ROC (TPR) against X-ROC (FPR) for the accurate classification of the class value of Original (O) with ROC accuracy of 0.7188 original SQLite query. The ROC curve is produced by comparing For various threshold levels, the true positive rate (TPR) versus the false positive rate (FPR). The ROC is a matrix that evaluates the model's capacity to discriminate between binary (0 or 1) classes. In machine learning, the true-positive rate is sometimes referred to as sensitivity, recall, or probability of detection The false-positive rate, often known as the likelihood of a false alarm, can be calculated as (1-specificity). Good categorization is shown by the points above the diagonal line (better than random). If the model is skewed towards the top left corner of the ROC's Y-axis, the model's performance increases. (TPR) represented as B, however, if the curve is skewed towards the lower right of the X-axis of the ROC (TPR) is an indication that the model or algorithm is not performing at its best. The curve in figure 4.4 shows a reasonable amount of accuracy of the ROC as 0.7188. This shows that using machine learning (SVM algorithm) in designing a hybrid framework architecture for mitigating vulnerability in e-commerce has yielded positive results.

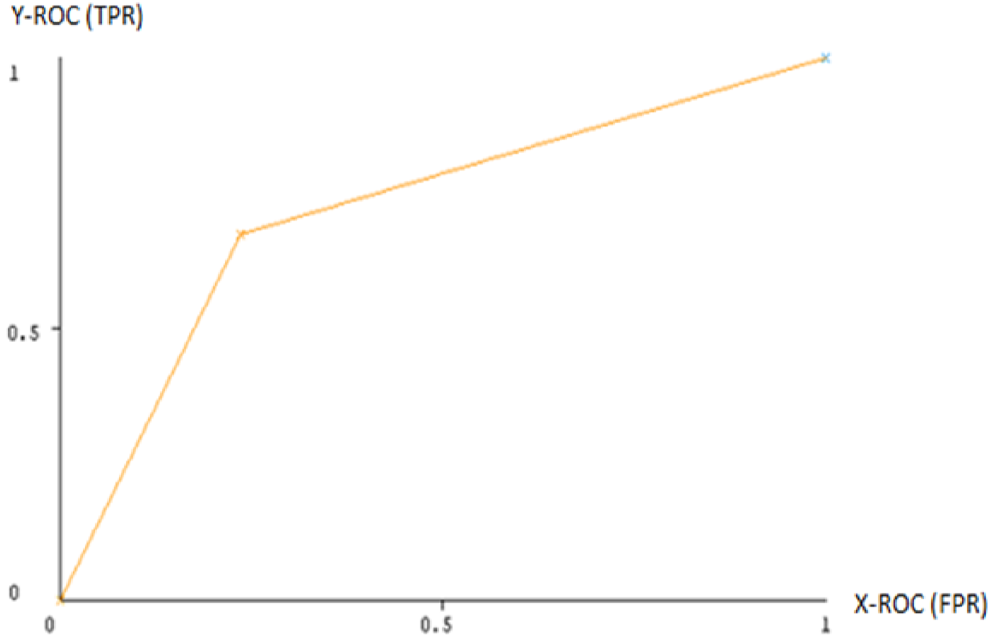


Figure 4.4: Plot the Receiver Operating Characteristic (ROC) Area for the Original Class Value.

4.7 Classifier Visualize: Threshold Curve Of Suspicious Roc

Plot area under the receiver operating characteristic (ROC) = 0.7188 (Class Value for SUSPICIOUS). As shown in figure 4.5 a plot of Y-ROC (TPR) against X-ROC (FPR) for the accurate classification of the class value of Suspicious (S) with ROC accuracy of 0.7188 suspicious SQLiA query. The ROC curve is a matrix that evaluates the model's capability to discriminate between binary (0 or 1) classes. It is created by comparing the true positive rate (TPR) vs the false positive rate (FPR) at various threshold values. The true-positive rate in machine learning is sometimes referred to as sensitivity, recall, or chance of detection. The likelihood of a false alarm commonly referred to as the false-positive rate may be calculated as (1-specificity). The categorization is accurate, as evidenced by the points above the diagonal line (better than random). The model performs better if the curve is skewed towards the top left corner of the ROC (TPRY-axis;)'s on the other hand, if the curve is skewed towards the bottom right of the X-axis, the model performs worse. The design or algorithm is not working at its best. The curve below demonstrates that the ROC has an acceptable level of accuracy of 0.7188 which satisfied that using a support vector machine in a hybrid architecture framework has aided in mitigating vulnerability attacks on e-commerce.

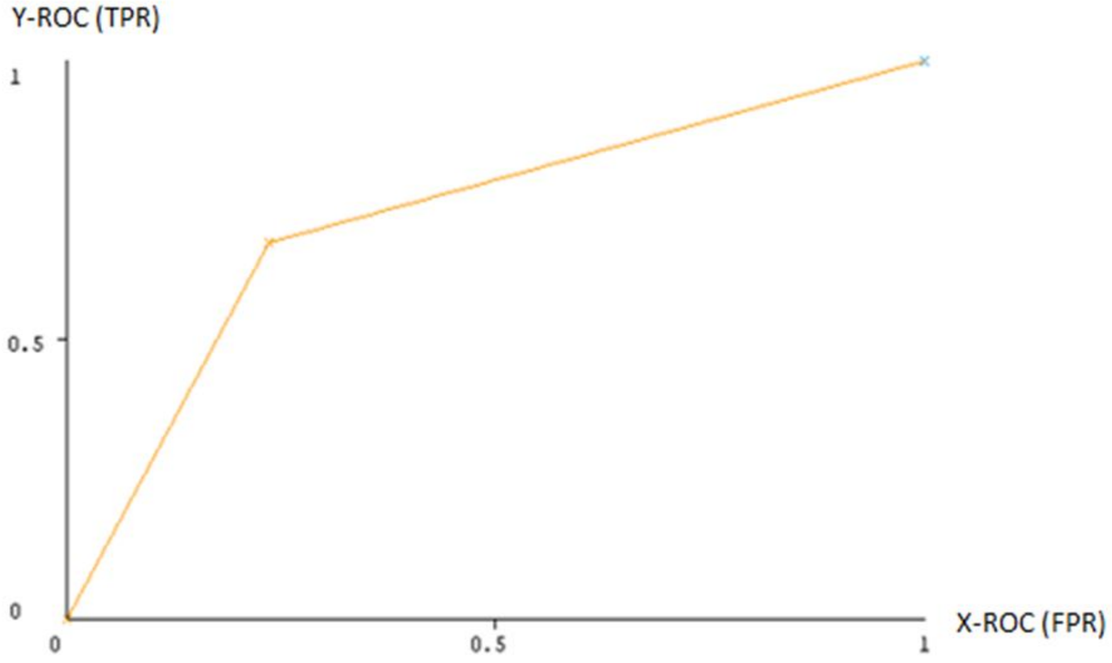
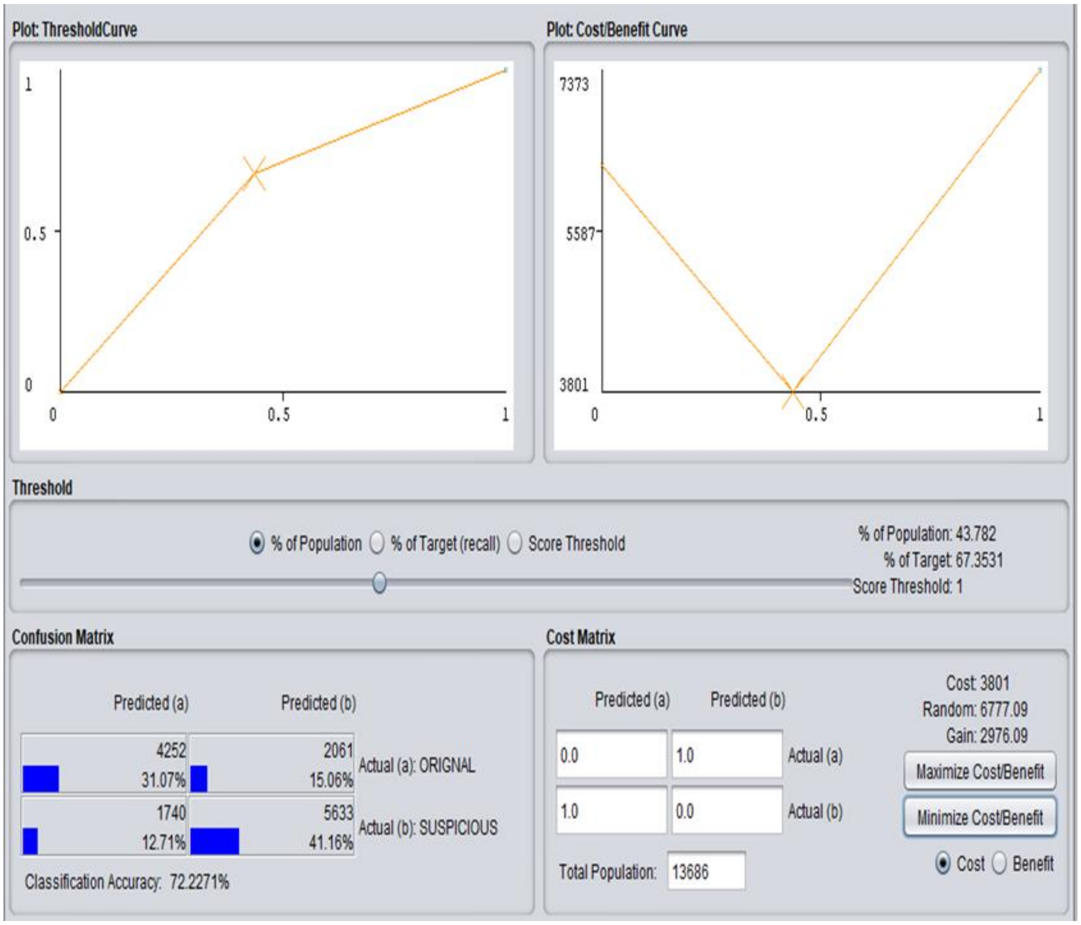


Figure 4.5: Plot Area Under the Receiver Operating Characteristic (ROC)for the Class Value of SUSPICIOUS

4.8 Plot Threshold Curve And Cost/Benefit Curve Of Class Original And Suspicious

Figure 4.6 is a graphical representation of the model's advantages totaled up and the expenses connected with it subtracted. The plot labeled cost/benefit curve depicts the cost/benefit analysis curves. The threshold curve is depicted against the cost/benefit curve as shown in Figure 4.6. The 'X' point on the cost/benefit curve represents the lowest cost/benefit. This is also the place where the classifier's accuracy is highest, at 72.2271 percent. As indicated in the confusion matrix, the classifier has paid a cost of 3801 units for incorrectly categorized instances 3801 for suspicious and 9885 for successfully classified instances of the original. If the SQLiA were to be randomly classified, the cost would be 6777.09 units. The gain or profit obtained from using the classifier is therefore 2976.09 units, which is a substantial profit and this phenomenon has juxtaposed the fact that using a support vector machine algorithm will be an effective way of designing hybrid framework architecture in mitigating vulnerability on e-commerce.



● % of Population ○ % of Target (recall) ○ Score Threshold

% of Population: 43.782
% of Target: 67.3531
Score Threshold: 1

Predicted (a)	Predicted (b)	
4252 31.07%	2061 15.06%	Actual (a): ORIGNAL
1740 12.71%	5633 41.16%	Actual (b): SUSPICIOUS

Classification Accuracy: 72.2271%

Predicted (a)	Predicted (b)	
0.0	1.0	Actual (a)
1.0	0.0	Actual (b)

Total Population: 13686

Cost: 3801
Random: 6777.09
Gain: 2976.09

● Cost ○ Benefit

Figure 4.6: Plot Area of Threshold Curve for the Class Value of ORIGINAL and SUSPICIOUS

5. CONCLUSION

This idea is to create a secure application that uses SVM to classify original and suspect query strings and XSS, for training and classification, a dataset of various sizes is used. The performance of this system is demonstrated by several characteristics such as accuracy, detection and training times, TPR, TNR, FPR, and graphical descriptions are all taken into account. This system has the best accuracy performance among the existing systems, with a score of 72.2271 %.

For most people throughout the world, the e-commerce strategy has radically changed the way they do business. One of these systems is the typical retail system. E-commerce has been impacted by a shift away from a manual-based method to doing business using electronic devices such as computers and portable devices. In the commercial world, this innovative technique is now regarded as a must-have application. The reason for this is that an e-commerce business is more cost-effective to set up and operate than a traditional system. It's also simple for customers to find things, make informed decisions, and pay for them; it also acts as a conduit for other customers throughout the world.

The security of such platforms must be strengthened and not underestimated to produce an operational and effective e-commerce platform. Attackers have been successful in exploiting computer systems in a variety of ways, from software to hardware. The security requirements for an e-commerce system are to secure the stakeholders' content, services, personal data, and payment information by preserving their privacy and confidentiality at all times.

The findings of earlier research on the security of e-commerce platforms were discussed in this study. The scope, however, is restricted to vulnerabilities. The stated research questions and observations were addressed with the primary goal of discovering vulnerabilities in three (3) hosted e-commerce platforms under the OWASP Top Ten report.

To sum up, the study found that assessing security risks in Ecommerce platforms can provide in-depth information on existing and new vulnerabilities in these platforms. Following that, the identified vulnerabilities will be remedied, as Ecommerce platforms with no or few vulnerabilities will provide a smooth business transaction experience for both the business owner and the customers.

We, therefore, recommend that there must be a critical look at backend Ecommerce application vulnerability and security enhancements in the future.

REFERENCES

[1] K. Akomea-Agyin and M. Asante, "Analysis of security vulnerabilities in wired equivalent privacy (WEP)," International Research Journal of Engineering and Technology, vol. 6, no. 1, pp. 529-536, 2019.

[2] I. Baako, S. Umar, and P. Gidisu, "Privacy and security concerns in electronic commerce websites in Ghana: a survey study," International Journal of Computer Network and Information Security, vol. 11, no. 10, p. 19, 2019.

[3] M. K. Kissi and M. Asante, "Penetration testing of IEEE 802.11 encryption protocols using Kali Linux hacking tools," International Journal of Computer Applications, vol. 975, p. 8887, 2020.

[4] V. Appiah, M. Asante, I. Kofi Nti, and O. Nyarko-Boateng, "Survey of Websites and Web Application Security Threats Using Vulnerability Assessment," 2018.

[5] O. Safianu, F. Twum, and J. Hayfron-Acquah, "Information system security threats and vulnerabilities: evaluating the human factor in data protection," International Journal of Computer Applications, vol. 143, no. 5, 2016.

[6] M. J. Islam, M. Mahin, A. Khatun, S. Roy, S. Kabir, and B. C. Debnath, "A comprehensive data security and forensic investigation framework for cloud-iot ecosystem," GUB Journal of Science and Engineering, vol. 4, 2019.

[7] C. E. Leiserson et al., "There’s plenty of room at the Top: What will drive computer performance after Moore’s law?," Science, vol. 368, no. 6495, 2020.

[8] R. S. Devi and M. M. Kumar, "Testing for Security Weakness of Web Applications using Ethical Hacking," in 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 2020: IEEE, pp. 354-361.

[9] P. Ferrara, A. K. Mandal, A. Cortesi, and F. Spoto, "Static analysis for discovering IoT vulnerabilities," International Journal on Software Tools for Technology Transfer, vol. 23, no. 1, pp. 71-88, 2021.

[10] A. Vetterl, "Honeypots in the age of universal attacks and the Internet of Things," University of Cambridge, 2020.

[11] R. Ma, P. Shi, Z. Wang, and L. Wu, "Resilient filtering for cyber- physical systems under denial- of- service attacks," International Journal of Robust and Nonlinear Control, vol. 30, no. 5, pp. 1754-1769, 2020.

[12] W. Feng, Y. Qin, S. Zhao, and D. Feng, "AAoT: Lightweight attestation and authentication of low-resource things in IoT and CPS," Computer Networks, vol. 134, pp. 167-182, 2018.

[13] R. A. Teimoor, "A Review of Database Security Concepts, Risks, and Problems," UHD Journal of Science and Technology, vol. 5, no. 2, pp. 38-46, 2021.

[14] S. Jain and D. Chawla, "A relative study on different database security threats and their security techniques," Int. J. Innov. Sci. Res. Technol., vol. 5, no. 5, pp. 794-799, 2020.

[15] J. Jabez, S. Gowri, S. Vigneshwari, J. A. Mayan, and S. Srinivasulu, "Anomaly detection by using CFS subset and neural network with WEKA tools," in Information and Communication Technology for Intelligent Systems: Springer, 2019, pp. 675-682.

[16] B. Jiang, "Computer Security Vulnerabilities and Preventive Measures," in International Conference on Application of Intelligent Systems in Multi-modal Information Analytics, 2020: Springer, pp. 752-759.

[17] M. A. M. Yunus, M. Z. Brohan, N. M. Nawi, E. S. M. Surin, N. A. M. Najib, and C. W. Liang, "Review of SQL Injection: Problems and Prevention," JOIV: International Journal on Informatics Visualization, vol. 2, no. 3-2, pp. 215-219, 2018.

[18] R. Pandey, V. Jyothindar, and U. K. Chopra, "Vulnerability Assessment and Penetration Testing: A portable solution Implementation," in 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), 2020: IEEE, pp. 398-402.

[19] M. A. Helmiawan, E. Firmansyah, I. Fadil, Y. Sofivan, F. Mahardika, and A. Guntara, "Analysis of Web Security Using Open Web Application Security Project 10," in 2020 8th International Conference on Cyber and IT Service Management (CITSM), 2020: IEEE, pp. 1-5.

[20] T. A. Taha and M. Karabatak, "A proposed approach for preventing cross-site scripting," in 2018 6th International Symposium on Digital Forensic and Security (ISDFS), 2018: IEEE, pp. 1-4.

[21] N. El-Bakkouri and T. Mazri, "Security Threats in Smart Healthcare," The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 44, pp. 209-214, 2020.

[22] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," Neurocomputing, vol. 408, pp. 189-215, 2020.

[23] J. Choi, H. Kim, C. Choi, and P. Kim, "Efficient malicious code detection using n-gram analysis and SVM," in 2011 14th International Conference on Network-Based Information Systems, 2011: IEEE, pp. 618-621.

[24] S. Ray. "Understanding Support Vector Machine(SVM) algorithm from examples (along with code)."

Authors

Martin Doe is the head of the ICT department at Adventist SHTS, Kofiase. He had MPhil in Information Technology at the Department of Computer Science in the Kwame Nkrumah University of Science and Technology, Ghana.He is currently a Ph.D student at the University of Business and integrated development Studies, Ghana. His research area includes Cyber Security and computer networks and internet of things.

Christian Bakaweri Baatuomu is a Ph.D. Student of Computer Science in the Department of Computer Science at The Kwame Nkrumah University of Science and Technology, Ghana. His research areas are Cyber Security, and Machine Learning. Network Algorithm.

Prof. Michael Asante is an Associate Professor in Computer Science at the Department of Computer Science at the Kwame Nkrumah University of Science and Technology, Ghana. His research areas include Computer Security, Cyber Security, and networking.