International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT22857

Implementation of PSO Algorithm for Detection and Removal of XSS Attack

Bhanwar Lal[1], Irfan Khan[2]

[1]Research (MTech) Scholar (CSE), Shekhawati Institute of Engineering and Technology, Sikar, Rajasthan, India
[2]Assistant Professor (CSE), 1Shekhawati Institute of Engineering and Technology, Sikar, Rajasthan, India

Article Info

Publication Issue :
Volume 8, Issue 5
September-October-2022

Page Number : 39-51

Article History
Accepted: 01 Sep 2022
Published: 09 Sep 2022

ABSTRACT

In recent years, managing the security over the web has gained its importance. Use of appropriate security handling techniques help to solve controversies and to extract interesting scenarios based on the content of the web page. Many varieties of vulnerabilities prevail and Cross-Site Scripting (XSS) vulnerability is ranked among the top ten risks found over the web which is a mandatory issue that requires a solution. XSS vulnerability injects malicious code in many ways that rise during the browsing session. Analysis should be made over the web page to identify whether the page is vulnerable or not. A dataset is formulated that contains malicious and benign data. Malicious data are obtained from the XSS archive [source: www.xssed.com] which contains the vulnerable XSS web pages and benign data are the web pages that are obtained through queries from the Google search engine. The major constraint is the number of Lines of Code (LOC) present in the web page. Five samples from the dataset were considered and algorithms are applied. About 24 attributes are used by the classifier. The samples vary in terms of content and size. Different optimization techniques are applied and the results are analyzed. Evaluation measures like Detection Rate (DR), False Detection Rate (FDR) and F Score (FS) are calculated based on the Confusion Matrix. The final content obtained after the „XSS Handler phase" that is to be displayed on the browser is tested using black box testing technique and also using XSS and SQL Injection Scanner tool. The tool is capable of identifying promising XSS code available in web pages. Based on the experiments, it was observed that the generation of paths using PPACO achieves better results in terms of DR, FDR and FS than other algorithms.

Keywords: ACO, PSO, XSS, SQL, FDR, FS, Dataset, Train, Test, Attack, Genetic Algorithm.

I. INTRODUCTION

Nowadays web is a common platform for all categories of people involving e-payment, e-business, e-service and many more. All these applications require an interface with the users during the access of their session. The behaviour of the applications

may vary from time to time and application to application. Detecting the nature of the application is a hectic job which is to be done carefully. Wasserman (2008) spots that the web anomaly is more common and researchers are in a need to concentrate on its data input operation that deviates to an abnormal operation. Handling the web information involves a lot of risk since the user is unaware of the anomaly. So it is necessary to classify the normal and abnormal information. It is found from the past study that the ML and DM algorithms are more suitable to predict based on the history. Moreover the results of those algorithms are independent of the domain in which they work over and depend only on the accuracy of the collected dataset.

The focus of this work is to identify whether a given web page is benign or malicious towards XSS in order to secure the web application. Malicious is an activity which is a slight deviation that collapses the objective of the application. XSS attack is one amongst the top 10 vulnerabilities (Source: www.owasp.org). Lee et al. (2000) have devised a framework for modelling and constructing intrusion detection system which was the first customized detector for web systems. This framework was the first to teach the procedure of collecting the characteristics based on the data input, which paved way for the development of the Recommender Systems (RS) in the recent years.

Lots of information and knowledge are found around the globe in terms of web pages which grows year after year. Without an automatic extraction method, it is difficult to extract the hidden knowledge and information trapped between huge bytes of data. Classification Algorithms being a part of ML predicts based on the pre-known circumstances for analysing the future unknown circumstances. To show the significant differences between the classifiers, various statistical tests have been conducted using the dataset obtained from the XSSed (source: www.xssed.com) website.

A web page is associated with one or more attributes which forms tuples for a dataset that may reveal interesting correlations and associations amongst them. A classification system may be built with a better and suitable classifier. Many evolutionary techniques are applied for detection. Though the existing methods like genetic algorithm performs detection well, results are not that much appreciable for datasets with high dimensions and enormous LOC. Prevention is a hectic task in a web page that is to be accomplished after the detection procedure is done.

This motivated the research to focus on some of the critical issues, the challenge being the detection and prevention of XSS attack. The following may be the concerns that improve the performance are:

Proper attribute identification has to be observed for a web page to apply classifiers.
It is important to improve the performance by boosting the accuracy in the classification.
It is important to identify suitable error measures and error estimates for high dimensional dataset.
Generating paths is an iterative and time consuming process. It is essential to develop an approach for generating significant paths using optimization algorithms.
It is essential to discover the significant paths and to reduce the number of uninteresting paths without a compromise in its usefulness to the web user.

1.1 Problem Statement

The research works aims to improve the performance of preventing the XSS attacks without any compromise to the usage of the web user.
Generating a significant page tree is not a simple problem where the quantity and quality are dependent on the attributes, their control flow and the uniqueness of the dataset used for experimentation. This research considered algorithms that approached towards the formulation of binary

classes thereby analysing different precision, time and error measures. The research also monitors with a static comparison method in which attacks can be prevented from the user. The research aims to use an evolutionary algorithm that generates a probabilistic decision which tracks and performs the prevention action, thereby avoiding the attack

1.2 DATASETS USED FOR EXPERIMENTATION

The research work uses a user defined dataset for classification of information from the web page. The class of the data in the experimental dataset are of binary type. There are about 24 attributes and 500 tuples where the 24th attribute is the decision class which predicts the availability of the XSS attack (benign or malicious). The other 23 attributes contain the information about the presence or absence of a particular fact in the webpage leading to the decision. Information involving about 250 tuples was collected from the website (Source: www.xssed.com) that holds the malfunctioned data. The other 250 tuple web contents are obtained using Google search engine with specific input terms as keywords which can be deferred as the benign data. Commercial applications involving a number of users are also handled.

An event management application which has 3818 LOC handles the conduction of an event from its initial stage to end stage. These types of web sites support the end user in organising their functions.
Classifieds is a web application which holds advertisements about jobs, rental, buy, sell, personal, and a lot more. A web page coded with5745 lines is considered in the classifieds category.

A web application which manages property needs with a common reservation system for hotels of any cadre named Roomba was considered with 3438 LOC. This system is found to be prominently affected with attacks often.

A group influenced by a topic with more number of authors and viewers sharing and commenting their view over the web through a social networking media named Personal Blog is also considered as one of the datasets. This type of application suffers from a lot of traffic due to huge number of accesses and the selected dataset contains 17149 lines in terms of code.

JGossip is an application that belongs to the category of message boards and Bulletin Board System (BBS) which is a powerful but simple forum to share information. The forum has 79685 LOC with varied categories of discussions initiated and managed within.

These datasets used within the research are selected since they are of commercial type involving registered or nonregistered users of any number in any category and are prone to attacks of any kind. Variation in fundamental when dataset disables editing on a single machine and manufacturing becomes an unauthorized control. As you prepare for your management needs, many issues can be strategically kept away from it. For example, in any case, all the different settings for the two machines must be realized. Most of those path scaling problems can be solved in basic use if more manageable power is required. There are various free and exclusive answers available to help you manage the most scope. Each has its own advantages and disadvantages Model: Distribute maintenance to some PCs.

1.3 Mechanical Studies

AI deals with the transformation of research and some information using measurable using hull. Different trails are suitable for different trials and there are important factors to consider. The three well-known strategies are called ordering, re-organizational inspection, and recommended frameworks.
Model: Apply order calculation.

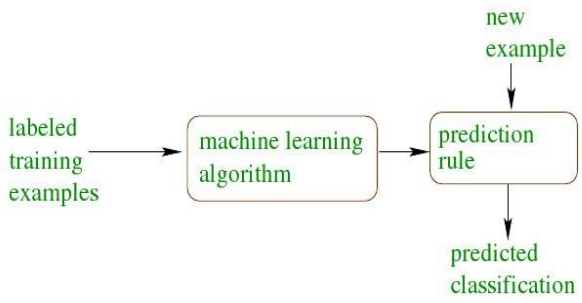

Fig 1.1 Demonstration of Model

1.4 Predictive modeling process

In information mining given information, Di = (xi, yi) is partitioned first for prescient demonstrating into three Sets:
Preparing set - In which perceptions are utilized to prepare the model with at least one calculation.
Approval set - In which approval information is utilized to foresee the model and locate the best model. This procedure is otherwise called tuning.
Testing set - This set is utilized to anticipate the last model execution. There are different methods to part the information, for example, even/odd, Venetian blinds, irregular, and visual review. In this paper, information dissected is medicinal services information so arbitrary split strategy is used to characterize train, approval, and test datasets.

1.5 Objectives

The problem considered in the work of the research is common to all XSS detections with effective construction of classifier, which is purely dependent on the characteristics of the datasets. Information present in the web pages are collected as datasets which are different from one another in terms of number in Lines of Code and the domain category. These datasets are used for estimating the performance of the classifier. The encryption and optimization methods are applied for the detection process and their performances are analysed. The handling procedure is also tested with a XSS and SQL Injection Tester tool (XSS & SQLi). The proposed PPACO algorithm was also experimented for five datasets which are generic in nature under the measures like Detection Rate (DR), False Discovery Rate (FDR) and F Score (FS).

2.1 CLASSIFIER AND ITS VARIANTS

Many techniques are found to extract the patterns from a huge database. Han & Kamber (2001) have identified that classification is one of the predictive techniques that predicts the group membership in a two-step process. The most successful factor is that the algorithms do not worry about the nature of the dataset. This feature paved a way for choosing the classification algorithms for this research work.

Likarsh et al. (2009) gave a proposal to detect malicious content present within Javascript tag using classification algorithms considering the obscure features. The classifier itself acts as a filter and provides a laudatory solution. Similarly Rieck et al. (2010) initiated a method named CUJO (Classification of Unknown Javascript Code) filtered patterns and devised a technique for capturing malignant code on the required computer. From the analysis, it is found that classification algorithms are most suitable to detect malicious content by performing analysis with various factors. Rule based classifier, DT, Nearest Neighbour (NNe), Artificial Neural Network (ANN) are some of the prominent classifiers present. These algorithms try to classify the tuple during the arrival of a tuple.

The most common DM algorithms like Simple Bayes (SB), Naïve Bayes (NB), Iterative Dichotomiser3 (ID3) and C4.5 (represented as J48 in WEKA) that induces to provide a DT. SB and NB which are generalized as Bayes Classifiers do not construct a decision tree. DTs are simple and easy to interpret. Since DTs are constructed only with the limited attributes obtained from the web page, the case of over fitting does not exist.

- Preventive Maintenance: Safeguarding the activities and exchanges at customary stretches for maintaining a strategic distance from information misfortune.
- Security Monitoring: Examining the irregular client conduct and following the malignant one among different clients.
- Risk Assessment: Assessing the danger of specific exchanges from past understanding, consistent management and examination.
- Fraud Detection: These strategies investigate the irregular qualities of a average client for recognizing the misrepresentation and cybercrimes.
- Network Monitoring: Observing the system disappointment tracks the quality and the shortcoming of a system way by expanding screening.

II. METHODS AND MATERIAL

The framework consists of two major phases. Initially the pre-processed dataset is labelled into positive and negative tuples based on the class label. The first phase is to model the various classifiers based on the training set of data. The classification of data by the classifiers is conducted in the second phase. Then the performances of the classifiers are compared and the one with the highest accuracy is chosen for the future processes.

Flow diagram of Proposed Model

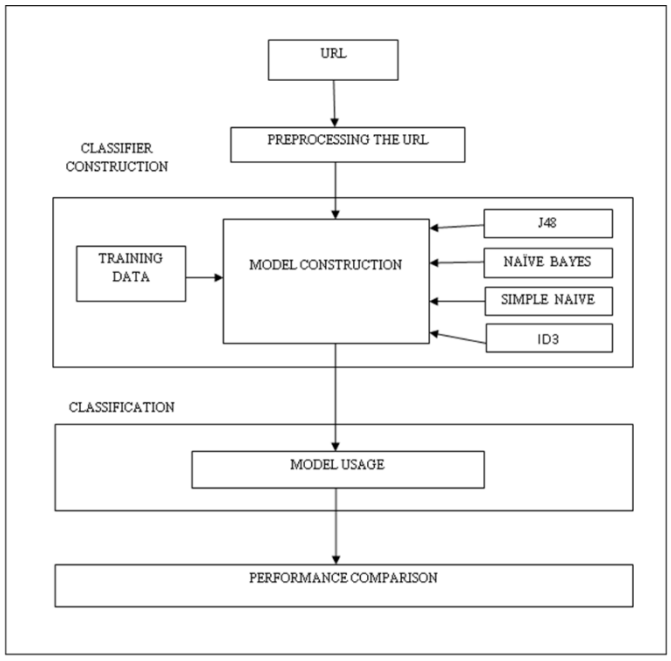

Figure 1.1 Workflow of the Classifiers

3.1 Performance Measures

For DM and ML algorithms particularly to the statistical classification problem, a table format visualizes the performance of the algorithm which is an error matrix or a confusion matrix or a matching matrix. The matrix is required in order to show the significant differences in the performance of classifiers. The classifier returns either a 0 or 1 that is denoted as „N" or „P" which is used for the 2 X 2 table formulation with False Positive(FP), False Negatives (FN), True Positives (TP) and True Negatives (TN). The following statistics namely True Positive Rate (TPR) (Equation (3.7)), False Positive Rate (FPR) (Equation (3.8)), Precision (PR) (Equation (3.9)), DR (Equation (3.10)) and FS Measure(Equation (3.11)) are calculated respectively.

4.2 Information Preprocessing

It is a cycle of evacuating all the loud and missing information from the informational collection.

At the point when information is given as information, it is important to preprocess the data. Text preprocessing is the cycle of getting ready and cleaning the information of dataset for characterization. It assists with decreasing the clamor in the content, improve the exhibition of the classifier and accelerate the characterization cycle. Preprocessing information has following 3 stages

•Tokenization: It is a sort of pre-preparing where running content is divided into words or sentences. Before any genuine content preparation is to be finished, the text should be sectioned into phonetic units, for example, words, accentuation, numbers, alpha-numeric, and so on this cycle is called tokenization. Tokenization, when applied to records, is the cycle of subbing a touchy information component with a non-sensitive same alluded as a token that has no outward or exploitable significance or worth. A record is considered as a string, and afterward parceled into a rundown of tokens. Stop words, for example, "the", "a", "and", and so forth are every now and again happening; in this manner the inconsequential words should be taken out.

•Stop word evacuation: In figuring, stop words are words which are sifted through previously or after preparing of normal language information (text). Stop words normally allude to the most widely recognized words in a language. The most widely recognized words are in text reports are relational words, articles, and favorable to things and so forth, that doesn't give the significance of the records. These words as treated as stop words. Model for stop words: the, in, a, an, with, and so forth [7] Hence it is vital to eliminate those words which show up excessively often that give no data to the errand. Stop words are eliminated to save both existences. Stop words are a fundamental piece of data recovery measure. The expulsion of stop

words increments execution and indexed lists. The stop words need to be eliminated for an explanation since they give no particular data for order reason.

• Stemming: It is the cycle for diminishing inferred words to their stem or root structure for example is primarily eliminates different additions therefore in the decrease of a number of words. For Example, the words client, clients, utilized, utilizing all can be decreased to "USE". This will diminish the necessary time-space

4.3. Train and assemble the AI model

In this progression, the dataset is isolated into two sections: preparing dataset and testing dataset. Preparing dataset contains 60% and the testing dataset contains 40% which are chosen haphazardly.

4.4 Highlight Selection

In the wake of preprocessing and Transformation the significant advance of text order is highlight determination. The fundamental thought of highlight choice is to choose a subset of highlights from the first record. The information contains numerous highlights, yet all the highlights may not be important so the element choice is utilized in order to kill the unessential highlights from the information absent a lot of loss of the data. Highlight the choice is otherwise called ascribes determination or variable selection[13]. It is performed by keeping the words with most elevated score according to the foreordained proportion of the significance of the word.

4.5 Arrangement

The archives can be arranged by administered what're more, solo strategies. At the point when the class mark of each the report is realized that is managed when the class name of the archive is not known that is called unaided.

4.6 Execution Measure

This is the last advance of information text characterization. This is tentatively done, as opposed to systematically. In this progression measures the exhibition. Numerous measures have been utilized like exactness and review.

4.7 XSS Detector

The PSO optimizer handles the web page that is given as input and the nodes are identified using the parser. The velocity values are initiated for every node and iterations are performed until all the nodes are involved in updating at least once or the criterion is satisfied. Tracing the path is a hectic process if the LOC of the input page is high. The optimizer formulates the path thereby undergoing a sequence of steps.
The process of this phase is briefed in Table 5.1.

Table 5.1 Path Traversal using PSO Optimizer
PSO Optimizer(Input, Output)
// Input: Uniform Resource Locator (URL)
// Output: Significant Paths Selected
// ic –iteration count; mi- maximum number of iteration; –best fitness
(1) {
(2) Initialize and based on pre-processing
(3) Calculate the fitness for
(4) Assign with
(5) Assign with where j is the index of best next element
(6) Assign ic with zero
(7) Loop until ic is less than mi
(8) Find if for every element
(9) Assign with when
(10) Assign with and with
(11) Continue the loop
(12) Update
(13) Update
(14) Calculate the fitness of
(15) Increment ic by 1
(16) Return
(17) }

III. EXPERIMENTS AND RESULTS

5.1 Dataset

Any webpage is an eligible data for the conduct of the experiment. For comparison, datasets mentioned in Section 1.4 are considered. The featured datasets belong to the XSSed class and are thereby considered for involving in the first phase of the optimizer.

5.2 Experimental Setup

The class label is obtained after pre-processing as in Section 3.3.2 over the requested web page. If found XSSed, the handler needs to apply rules to the page content in an effective manner.

The following parameters namely and velocity of the element are set by the parser by analyzing the considered web page. In this research work, the optimizing algorithm should not consider the insignificant features and utilize only the relevant ones without information loss. Parsing causes the process of generating the nodes and the significant paths are identified by PSO as explained in section 5.3.2. Two threshold values are set with values 60% (PSO-60) and 80% (PSO-80) respectively. The path retrieved by means of the PSO with specific threshold can be considered to be the significant path.

After the procedure is over, the handler phase takes over control and sanitation take place. Furthermore the page is reconstructed and presented to the browser. The performance of the optimizer is tested under the two threshold values using the XSS & SQLi tool.

IV. RESULTS AND DISCUSSION

The result of the experimentation done with PSO-60 and PSO-80 optimizer for the test domain is shown in Table 5.2. It is observed that PSO- 80 could yield more significant paths than PSO-60 since more number of paths is prevented through the handler process.

Table 5.2 Comparison of Evaluation Measures (%) for the Various Test Domains Using PSO

Criteria	Detection Rate		False Discovery Rate		F Score (FS)	
Test Domain	PSO-60	PSO-80	PSO-60	PSO-80	PSO-60	PSO-80
Events	46.91	49.38	15.56	20.00	60.31	61.07
classifieds	43.90	46.34	12.9	16.18	58.38	59.68
Roomba	28.10	30.72	12.24	14.55	42.57	45.19
Personal Blog	29.76	32.14	10.71	18.18	44.64	46.15
Jgossip	31.73	32.69	13.91	19.05	46.37	46.57

PSO-80 holds better for all the considered test domains in terms of DR. While analyzing the FDR score, the values of PSO-60 is found low and thereby PSO-60 scores to be the best algorithm. All the five-test domain for FS votes for PS0-80 with an increased difference of about 1% to 3% from PSO-60. Hence the optimizer PSO-80 suits better than PSO-60 for auto sanitization at the client end.

Implemented Algorithms

5.4.1 ACO Algorithm

ACO is an algorithm that can be applied to NP-hard combinatorial problems that seeks optimization. In this work, it is used to optimize the path traversal with significance in less number of iterations. This optimization problem can be formulated as a mathematical model.

A combinatorial optimization problem (S, f, Ω) is mapped on a problem with the following characteristics:

- $C = \{c_1, c_2 \ldots c_n\}$ of basic components, where C is a finite set.
- X contains the states of the problem, defined in terms of all possible sequences x over the elements of C, where X is a finite set.
- S is a subset of X. $(\boldsymbol{S \subseteq X})$, where S is a set of candidate solutions.
- $\boldsymbol{X' \subseteq X}$, defined via a problem dependent test that verifies that it is not possible to complete $\boldsymbol{X \in X'}$ a sequence into a solution satisfying all the constraints in Ω, where X‟ is a set of feasible paths.
- S^* is called optimal solution paths with $\boldsymbol{\subseteq X}_{S^*}$ $\boldsymbol{\subseteq S}$ and S^* ,

  where S^* is a nonempty set.

5.4.2 Dataset Pre-Processing

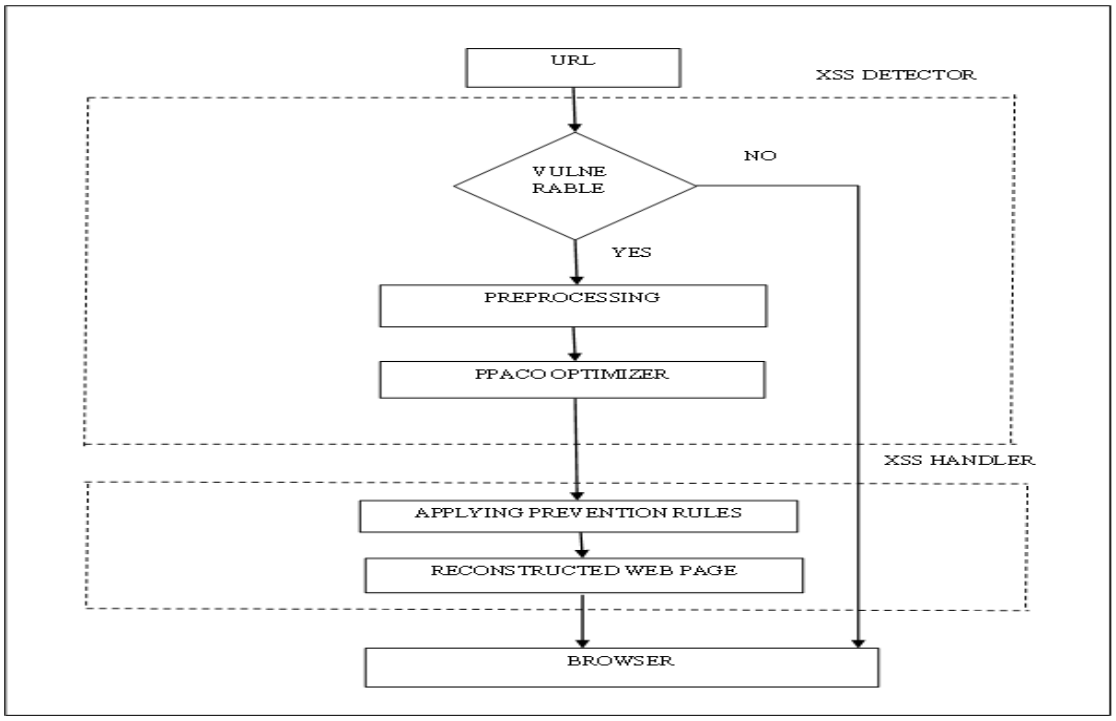

Figure 5.2 Workflow of the PPACO Optimizer

Results and Discussion

Various results obtained from the two thresholds of ACO with PPACO using XSS & SQLi is shown in Table 6.1. From the table, it is found that PPACO yields better results than other forms of ACO. It is also identified that ACO-80 is better than ACO-60 criteria for the evaluation measures since more number of the significant paths are considered for prevention by ACO-80 than ACO-60.

On analysing the results, it was identified that the value of PPACO is much better than ACO because it concentrates on the overall optimal paths of the complete web page whereas ACO 80 and ACO 60 consider only 80% and 60% of the totally tracked paths. Moreover, it is proven that the test domains used in these algorithms contain paths being listed at the end with most of the XSS occurrence leading to a good measure. A chance of impressive variation in result may occur between ACO 60 and ACO 80 if the test domains list the most likely XSS occurrence path in the top whereas PPACO will undergo no change in its values.

Table 6.1 Comparison of Evaluation Measures (%) for the Various Test Domains Using ACO and PPACO

Criteria	Detection Rate (DR)			False Discovery Rate (FDR)			F Score (FS)		
Test Domain	ACO-60	ACO-80	PPACO	ACO-60	ACO-80	PPACO	ACO-60	ACO-80	PPACO
Events	55.56	67.90	90.12	11.76	6.78	6.41	68.19	78.57	91.82
classifieds	44.72	55.28	82.93	15.38	2.86	13.56	58.52	70.46	84.65
Roomba	31.37	35.95	86.27	9.43	6.78	7.04	46.60	51.89	89.49
Personal Blog	29.76	41.67	79.76	19.35	10.26	16.25	43.48	56.91	81.71
JGossip	35.26	39.74	89.1	14.73	8.15	7.33	49.89	55.48	90.85

6.1 Statistical Validation

Anova test (DeCoster 2002) is a commonly used statistical test to inspect the significant difference in the performance between the optimizers. This test is performed to assess whether PPACO is significantly different from ACO-80 at 95% confidence level.

Tables 6.3, 6.4 and 6.5 show the results of one-way Anova test performed over test domains for the DR, FDR and FS respectively. The probability (P) value denotes the probability under the null hypothesis which states that the performance of PPACO and ACO-80 are the same. From Table 6.3, it is found that and smaller P value (i.e. P=0.000316<0.05) indicates the rejection of the null hypothesis, which means that the performance of PPACO is significantly different from ACO-80. Since the null hypothesis is rejected, post-hoc test is performed to identify the significant difference between the optimizers in terms of DR.

Table 6.3 Statistical Validation of DR using One-Way Anova Test

Source of Variation	Sum of Squares (SS)	Degrees of Freedom (df)	Mean Square Error (MS)	F–value (F)	Probability (P)-value	F Critical Value (F_{crit})
Between Groups	3520.8770	1	3520.8770	36.26002	0.000316	5.317655
Within Groups	776.8064	8	97.1008			

From Table 6.4, it is found that $F = 1.812549 < F_{crit} = 5.317655$ And smaller P value (i.e. P=0.215103>0.05) indicates the acceptance of the null hypothesis, which means that the performance of PPACO is similar to that of ACO-80.

Table 6.4 Statistical Validation of FDR using One-Way Anova Test

Source of Variation	Sum of Squares (SS)	Degrees of Freedom (df)	Mean Square Error (MS)	F–value (F)	Probability (P)-value	F Critical Value (F_{crit})
Between Groups	24.8377	1	24.8377	1.812549	0.215103	5.317655
Within Groups	109.6258	8	13.7032			

From Table 6.5, it is found that $F = 21.262750 > F_{crit} = 5.317655$

And smaller P value (i.e. P=0.001730<0.05) indicates the rejection of the null hypothesis, which indicated that the performance of PPACO differs significantly from ACO. Since the null hypothesis is rejected, post-hoc test is performed to identify the significant difference between the optimizers in terms of FS.

Table 6.5 Statistical Validation of FS using One-Way Anova Test

Source of Variation	Sum of Squares (SS)	Degrees of Freedom (df)	Mean Square Error (MS)	F–value (F)	Probability (P)-value	F Critical Value (F_{crit})
Between Groups	1567.7540	1	1567.7540	21.262750	0.001730	5.317655
Within Groups	589.8594	8	73.7324			

In this work, paired t-test is conducted for the post-hoc analysis. Paired t-test results obtained for ACO 80 and PPACO is shown in Table 6.6 and Table 6.7 for DR and FS respectively. From Table 6.6, it is evident that one-tailed P-value is 0.001323, which is less than the level of significance (0.05). So the null hypothesis is rejected and the alternate hypothesis is accepted which states that PPACO produces better detection rate than ACO-80. From Table 6.7, considering the value of one-tailed P-value which is 0.004007 proves that it is less than the level of significance (0.05). This means that the null hypothesis is rejected and the alternate hypothesis is accepted. The alternate hypothesis is that PPACO produces better detection than ACO-80 in terms of FS.

Table 6.6 Post-Hoc Analysis of DR using Paired t-Test

t-Test: Paired Two Sample for Means		
	ACO-80	**PPACO**
Mean	48.1080	85.6360
Variance	175.6122	18.5893
Observations	5	5
Pearson Correlation	0.308364022	
Hypothesized Mean Difference	0	
df	4	
t Stat	-6.655652	
P(T≤t) one-tail	0.001323	
t Critical one-tail	2.131846	
P(T≤t) two-tail	0.002646	
t Critical two-tail	2.776445	

Table 6.7 Post-hoc Analysis of FS using Paired t-Test

t-Test: Paired Two Sample for Means		
	ACO-80	**PPACO**
Mean	62.6620	87.7040
Variance	128.6440	18.8208
Observations	5	5

Pearson Correlation	0.174301
Hypothesized Mean Difference	0
df	4
t Stat	-4.905260
P(T≤t) one-tail	0.004007
t Critical one-tail	2.131847
P(T≤t) two-tail	0.008014
t Critical two-tail	2.776445

V. CONCLUSION

Classification is an effective XSS detection technique for the client side using a web browser. The training data includes benign and malicious web pages as samples with a vast difference in LOC, functionality, domain and appearance. In this research, the web pages act as transactional dataset for the 500 training instances with a class attribute for building an effective classification system. Experiments are conducted and it was observed that the training instances had obtained prediction accuracy for the classifiers. From the results, it was proved that J48 scored better prediction accuracy of about 88.19% and 89.44% in detecting the XSS under 10 CV and SF test options than the other classifiers. It was also observed that the training time is negligible between the classifiers and has no effect in deciding the optimal classifier for XSS detection.

Sanitization is required if a web page is detected to be XSSed. ESAPI rules are applied after performing a SC with the XML database available with the browser. The sanitized pages are tested with the tool XSS & SQLi. The two encryptors namely MD5-SC and SHA-SC equally performed with an average of 95% in DR when tested with five domains of varied LOC. MD5-SC, being the fastest in terms of computation with SHA- SC is selected for associating with the browser.

Detecting the XSS in an optimistic way with quick response to the end user is essential for the research since sanitization performs effective prevention. Two social inspired techniques employed in this research gave a good guidance in understanding the nature of the paths that are significant. It is detected that the tracking of the path is not sufficient to improve the DR for all the test domains under thresholds of 60% and 80%.

The optimization algorithm designed in this research is able to fit into the „XSS detector" phase. The proposed heuristic PPACO algorithm in this research helped for faster convergence leading to a small set of significant paths had achieved a better value for the DR, FDR and FS values.

The results of the proposed PPACO are compared with PSO and ACO algorithms. Using Anova and paired t-test, a substantial improvement in DR, FDR and FS was observed in the performance of PPACO. The detection using the PPACO generated path outperforms the other algorithms except MD5-SC and SHA-SC. The MD5-SC and SHA-SC presents a potential increase in value for DR, FDR and FS than PPACO because it applies prevention rules for every node without considering the LOC. As a result of this, the waiting time for the presentation of the content after sanitization at the browser end for the user also increases invariably.

Furthermore, the research concludes that a combination of PPACO along with ESAPI rules referred as the auto sanitization process is the best opted solution for a web browser. Furthermore, a web

page can be effectively XSS detected by J48 classifier in terms of DR, FDR and FS.

VI. REFERENCES

[1]. Adi, E 2012, „A design of a proxy inspired from human immune system to detect SQL injection and cross-site scripting", Procedia Engineering, vol. 50, pp. 19–28.

[2]. Adi, E & Salomo, I 2010, „Detect and sanitise encoded cross-site scripting and SQL injection attack strings using a hash map", Australian Information Security Management Conference.

[3]. Ahmed, AA & Ali, F 2016, „Multiple-path testing for cross site scripting using genetic algorithms", Journal of Systems Architecture, vol. 64, pp.50-62.

[4]. Alfaro, JG & Arribas, GN 2007, „Prevention of Cross-Site Scripting Attacks on Current Web Applications", OTM confederated International Conference On the Move to Meaningful Internet Systems, pp. 1770-1784.

[5]. Anupam, V& Mayer, A, 1998,"Secure Web Scripting", IEEE Journal of Internet Computing, vol. 2, no. 6, pp. 46-55.

[6]. Arulsuju, D 2011, "Hunting Malicious Attacks in Social Networks", Proceedings of 3rd International Conference on in Advanced Computing, pp. 13–17.

[7]. Avancini, A, Ceccato, M & Kessler, FB 2012, „Grammar Based Oracle for Security Testing of Web Applications", 7th International Workshop on Automation of Software Test , pp. 15–21.

[8]. Barhoom, TS & Kohail, SN 2011, „A new server-side solution for detecting cross site scripting attack", International Journal of Computational Information System, vol. 3, no. 2, pp. 19–23.

[9]. Bates, D, Barth, A & Jackson, C 2010, „Regular Expressions Considered Harmful in Client-side XSS Filters", Proceedings of the 19thInternational Conference on World Wide Web, pp. 9.

[10]. Bau, J, Wang, F, Bursztein, E, Mutchler, P & Mitchell, JC 2012, Vulnerability Factors in New Web Applications: Audit Tools, Developer Selection & Languages", Stanford Technical Report

[11]. Bojinov, H, Bursztein, E & Boneh, D 2009, "XCS: Cross Channel Scripting and its Impact on Web Applications", Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 420– 43

[12]. Booker, LB, Goldberg, DE& Holland, JH 1989, „Classifier systems and genetic algorithms", Artificial Intelligence, vol. 40, no. 1-3, pp. 235-282

[13]. Brinhosa, RB, Westphall, CM & Westphall, CB 2012, „Proposal and Development of the Web Services Input Validation Model", IEEE Network Operations and Management Symposium, pp. 643–646.

[14]. Cao, Y, Yegneswaran, V, Porras, P & Chen, Y 2011, "POSTER:A Path Cutting Approach to Blocking XSS Worms in Social Web Networks", Proceedings of the 18th ACM Conference on Computer and Communications Security, pp. 745–747

[15]. Cyber Security Survey 2016, Available from:<http://www.businessinsider.com/cybersecurity-report-threats-and opportunities- 2016

Cite this article as :

Bhanwar Lal, Irfan Khan, "Implementation of PSO Algorithm for Detection and Removal of XSS Attack ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8 Issue 5, pp. 39-51, September-October 2022. Available at doi : https://doi.org/10.32628/CSEIT22857
Journal URL : https://ijsrcseit.com/CSEIT22857