Study of Security Issues and Development of Risk Minimization Techniques for Web Applications.

A Ph.D. Dissertation

Presented by

Jayamsakthi Shanmugam

Research Scholar.

Objectives and Scope of Study.

- Study of web application security issues.
- Study of risks associated with web applications.
- Development of solutions to the web application vulnerabilities to minimize risk.

Introduction.

- Risks involved in web application.
  - Recognition - Authentication of the customer.
  - Authorization – Ability to create a legitimate legal relationship for a customer.
  - Mutual Signing and acceptance by the customer and by the web applications on the terms and conditions.
  - Irrevocable evidence that the conditions were accepted by all parties.
  - Privacy.

….. Continued

Introduction – Continuation.

- Major Web Security Issues
  - Cross Site Scripting
  - Injection Flaws
  - Malicious File Execution etc.

….. Continued

Introduction – Continuation.

Three types of XSS Vulnerability:

- Non Persistent XSS
- Persistent or Stored XSS
- DOM based XSS

Earlier Research on XSS Vulnerabilities and Solutions.

- Server based solutions.

- Client based solutions.

… Continued

Limitations of earlier research contributions – Continuation.

- Web applications are developed in various languages and the solutions proposed are language specific.
- The solutions did not consider the applications that receive input from various interfaces apart from the web browser.
- Client side solutions do not incorporate updates on a regular basis.

… Continued

Limitations of earlier research contributions – Continuation.

- Existing solutions are web page based. To secure from the new threat, the solution need to be incorporated in all web pages.- A very enormous and a difficult task.
- Existing server side solutions proposed so far are prone to zero-day attacks.
- When a new web page is introduced, the security mechanisms need to be introduced at a web page level.

… Continued

Problem formulation - Factors considered.

Factors considered for providing the solution:

- Does the system provide any financial service?
- What is the frequency of the changes in the web application?
- Can the system be tolerant to false negatives?
- Is performance an important criteria for the web application?
- Could zero-day attacks be permitted by the application?
- Is the source of input only the web browser or is it also expected from other interfaces?

… Continued

Problem formulation - Factors considered – Continuation.

- The research aims to provide a solution for the web applications developed in different languages.
- The solutions provided to secure the web pages when a new threat is introduced should be independent of the web page implementation.
- The security layer should hide the entry points to the web pages.

… Continued

Problem formulation - Factors considered – Continuation.

- No update should be necessitated at the client side.
- The solution should have the flexibility to accept tags as input.
- The security layer should completely be separated from the web application.
- Provisions should be made to introduce new web pages dynamically, with no modification.

Testing Methodology and data collection.

- Adopted dynamic testing approach to test the solution.
- Around 2200 vulnerable inputs are collected from white hat, black hat and research groups to test the solution.
- 108 test cases are developed for testing.
- Solutions have been tested with 6000 malicious inputs and 5000 non vulnerable inputs.
- The average time has been taken for 10 cycles of execution for each proposed approach.

Solutions Proposed.

- A solution to block Cross Site Scripting Vulnerabilities based on Service Oriented Architecture
- Server side solution to block Cross Site Scripting (XSS) for variety of web applications.
- Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities.
- Thread based Intrusion Detection and Prevention System for XSS and Application Worms.
- Improved trust metrics and variance based authorization model in e-commerce to identify server side hacking.

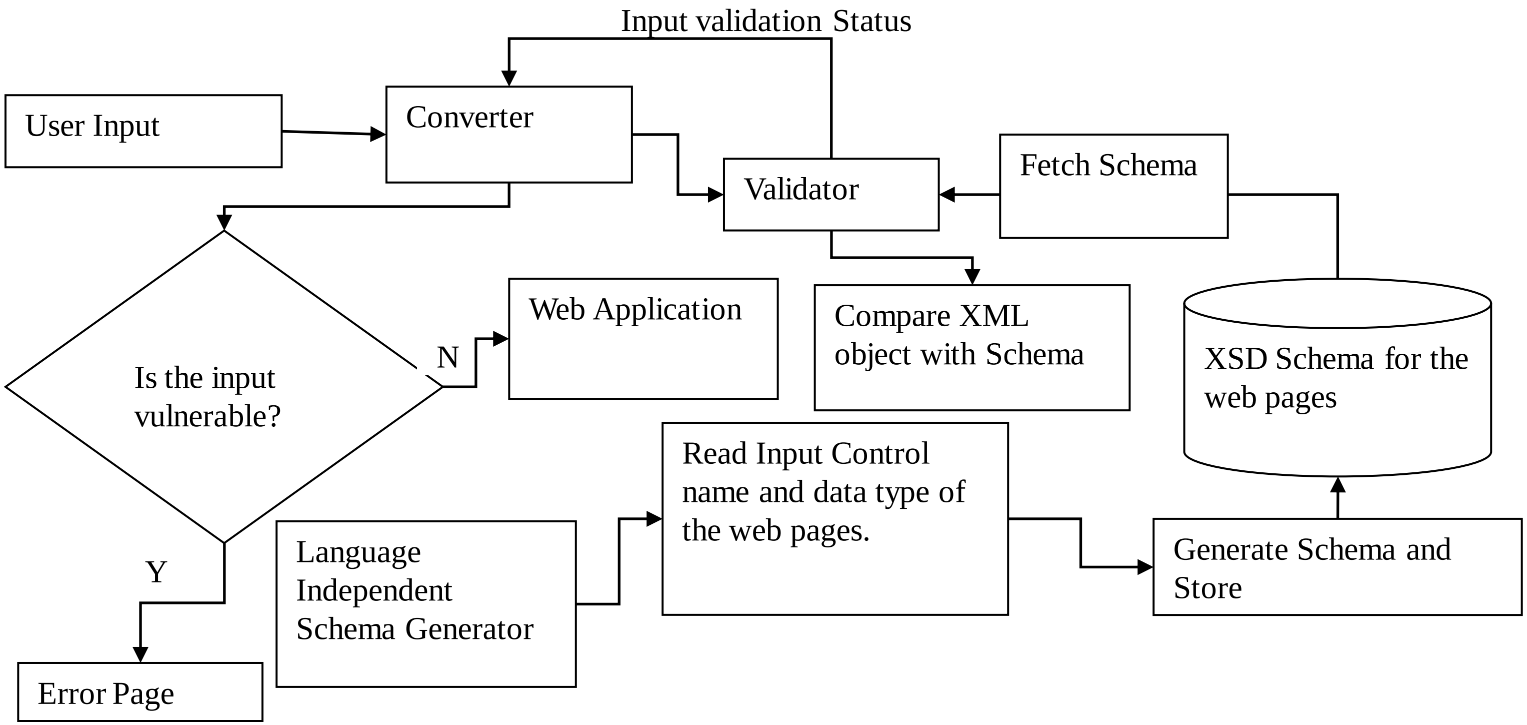
Service Oriented Architecture based solution.

Why SOA is needed?

- Web pages are developed in different languages like PHP, ASP, JSP, HTML, CGI-PERL, .Net etc.
- No Single solution is available that can be applied on all web applications to protect it from XSS vulnerabilities.

Service Oriented Architecture based solution – Technical Details.

SOA based XSS Blocker flow


Input validation Status
User Input
Converter
Validator
Fetch Schema
Is the input vulnerable?
N
Web Application
Compare XML object with Schema
XSD Schema for the web pages
Read Input Control name and data type of the web pages.
Y
Language Independent Schema Generator
Generate Schema and Store
Error Page

Service Oriented Architecture based solution – Implementation and evaluation details.

- Schema Generator is developed in .Net.
- Implemented on a web application developed in JSP/Servlets deployed in a JBOSS Server.
- Tested with 2000 XSS vulnerable inputs collected from various research groups.

… Continued

Service Oriented Architecture based solution – Implementation and evaluation details – Continuation.

- Set the input limit to 250 characters for testing purposes.
- 108 variants of XSS vulnerabilities are tested and the solution is found effective.
- Few false negatives were diagnosed due to encoded input.

Service Oriented Architecture based solution – Merits.

- Solution is independent of the languages in which the web pages are developed.
- Minimal configuration.
- Addresses XSS vulnerabilities from all interfaces.
- Security mechanism is centralized.
- Whenever the schema generator is changed, only the XSD forms need to be changed for the web application stored for each web page. Web pages do not require changes in it.
- Solution approach is modularized.

Service Oriented Architecture based solution – Limitations.

- When a new threat is introduced by hackers, XSD files for the web application needs to be generated again to protect the application from the new threat. It is a overhead for application maintenance.

- The solution is vulnerable to Zero-day threats.

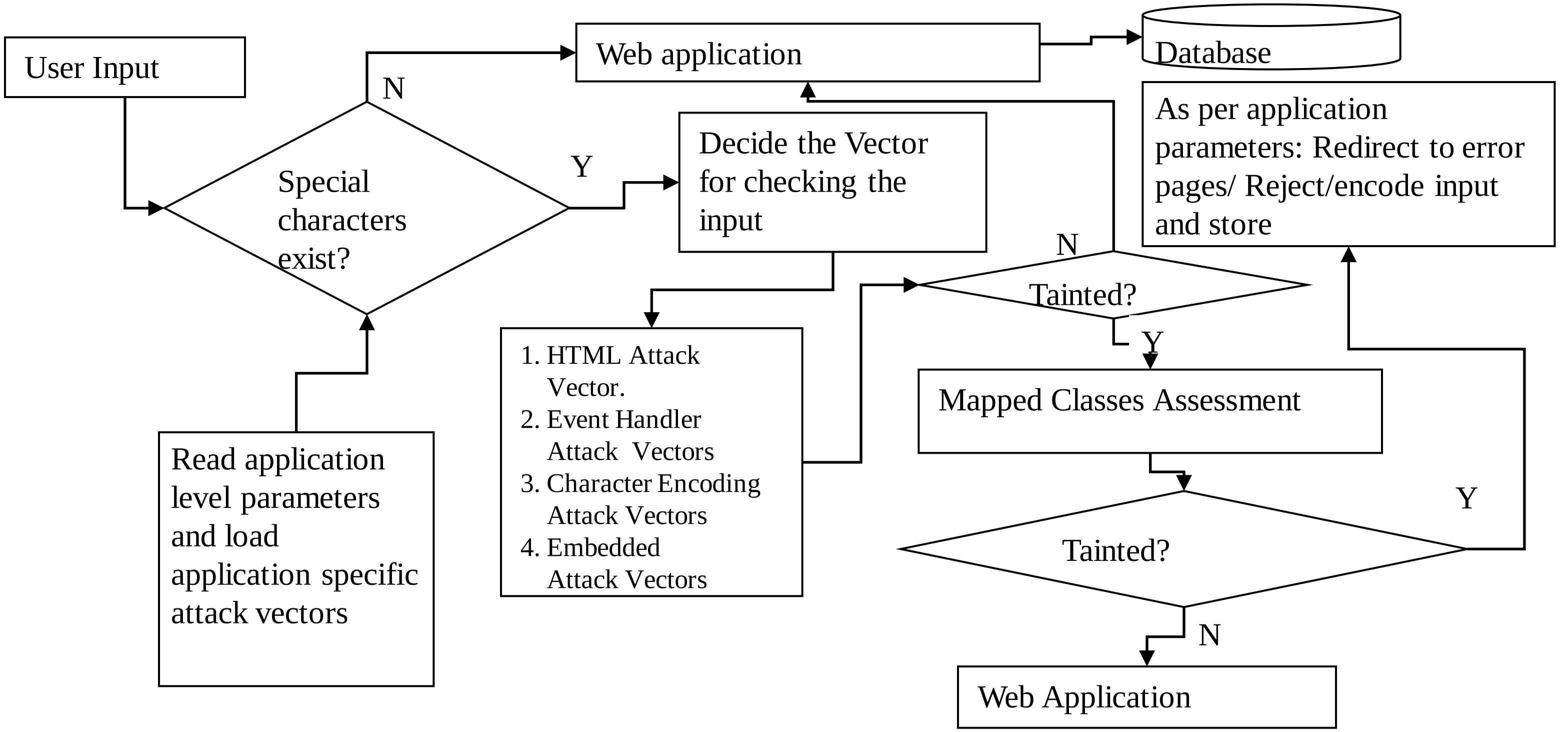
Server side solution to block XSS for variety of web applications.

Need for the development of a Server Side Solution for variety of web applications.

- Security mechanisms applied on one web application may not be applicable for other as web applications are built for various purposes.
- Web pages in the web application are changed every day.
- New web pages are introduced frequently in the web application.

Server side solution to block XSS for variety of web applications – Technical details.

Flow of user input through components.


User Input
Special characters exist?
N
Y
Web application
Database
Decide the Vector for checking the input
As per application parameters: Redirect to error pages/ Reject/encode input and store
N
Tainted?
Y
1. HTML Attack Vector.
2. Event Handler Attack Vectors
3. Character Encoding Attack Vectors
4. Embedded Attack Vectors
Mapped Classes Assessment
Read application level parameters and load application specific attack vectors
Y
Tainted?
N
Web Application

Server side solution to block XSS for variety of web applications – Implementation details.

- Implemented on a web application developed in JSP/Servlets deployed in a JBOSS Server.
- Application attributes are defined to reduce the processing time of the server.
- Extensible Mark up Language is used for defining the application attributes, attack vectors and corresponding mapping functions proposed in this approach.

Server side solution to block XSS for variety of web applications – Evaluation details.

- Application performance is assessed without the application of security layer and with the application of security layer.
- Both the approaches are tested with 6000 malicious inputs, 5000 non vulnerable inputs. The average time has been taken for 10 cycles of execution of each approach.

Experimentation Results:

	Vulnerable input processing time in milliseconds to process 6000 vulnerable inputs	Non vulnerable input processing time in milliseconds to process 5000 inputs	Random generator program test for 5000 inputs, represented in milliseconds with a mixture of vulnerable and non vulnerable inputs.
Security Mechanisms applied	2300	569	870
No Security Mechanisms applied	2000	500	836

Server side solution to block XSS for variety of web applications – Merits.

- The configuration of attack vectors, object maps and application level parameters are all one time configurations for the application.
- There is a complete separation between web application and the security implementation. So, the functionality of the web application can be added or modified or removed without modifying the security layer.
- Modularized.
- Only the security layer needs to be modified for the new threats.

Server side solution to block XSS for variety of web applications – Limitations.

- Applicable for the application where the web application is the only source of input.
- The solution is vulnerable to zero-day attacks.
- Applicable for non-financial web applications.

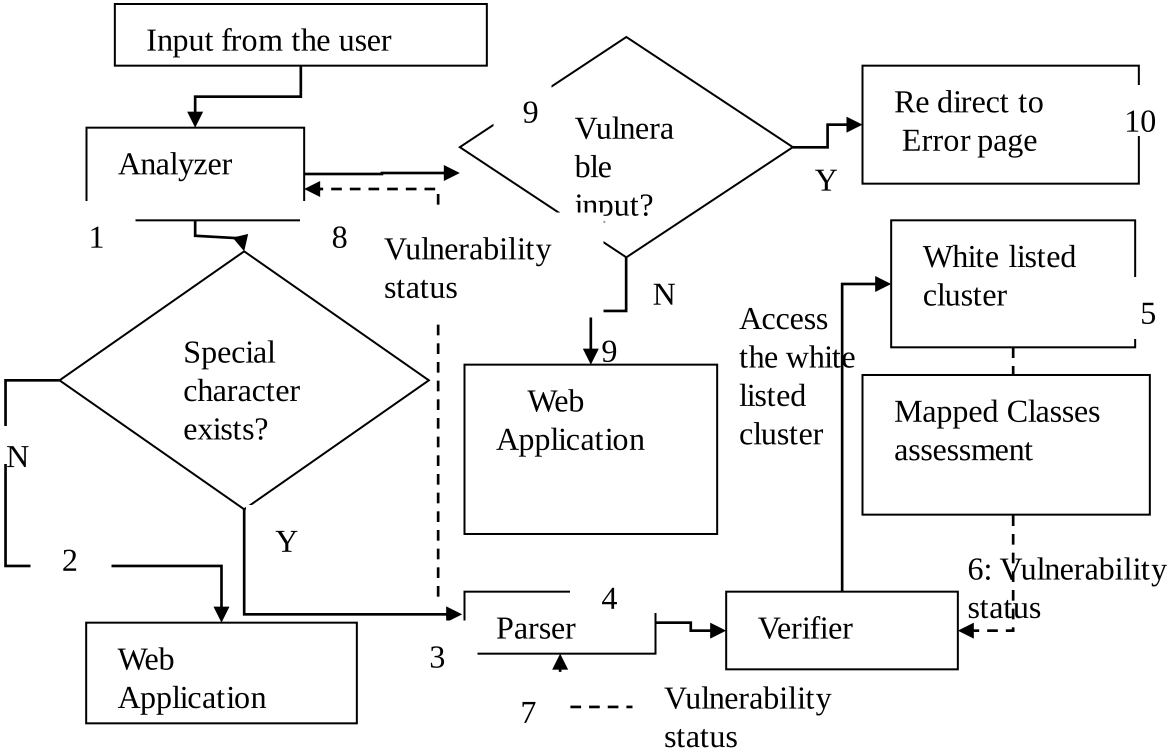
Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities.

Need for the development of this solution.

- □ High response time is needed for financial web applications.
- □ Security solution should protect the web application from Zero-day threats.
- □ When web pages are introduced or modified in the web application, it should not require changes to incorporate security mechanisms.

Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities– Technical details.

Flow of user input through components.



Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities– Implementation details.

- Struts framework web.xml is modified to redirect the HTTP requests to the analyzer class.

- Implemented on a web application developed in JSP/Servlets deployed in a JBOSS Server.

- XML is used for the definition of clusters.

Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities– Evaluation details.

Experimentation Results:

	Vulnerable input processing time in milliseconds to process 6000 vulnerable inputs	Non vulnerable input processing time in milliseconds to process 5000 inputs	Random generator program test for 5000 inputs, represented in milliseconds with a mixture of vulnerable and non vulnerable inputs.
Security Mechanisms applied	2300	549	850
No Security Mechanisms applied	2000	500	836

- 0.33 to 0.35 milliseconds to process a single vulnerable request.
- .008 milliseconds difference to process a non-vulnerable input.

Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities – Merits.

- This solution is applicable for financial web applications where the web application should be protected from zero-day attacks.
- The response time is reduced as it checks only the goodness of the user input.
- The white listed cluster entries are low compared to the black listed tags and does not require an update frequently.
- The solution is modularized, configurable and maintainable.

Behavior-based anomaly detection on the server side to reduce the effectiveness of Cross Site Scripting vulnerabilities – Limitations.

- This solution could lead to false positives if the tag is not included in the white listed cluster.

- Needs an update in white listed cluster when a new tag/attribute or method is introduced to reduce the false positives.

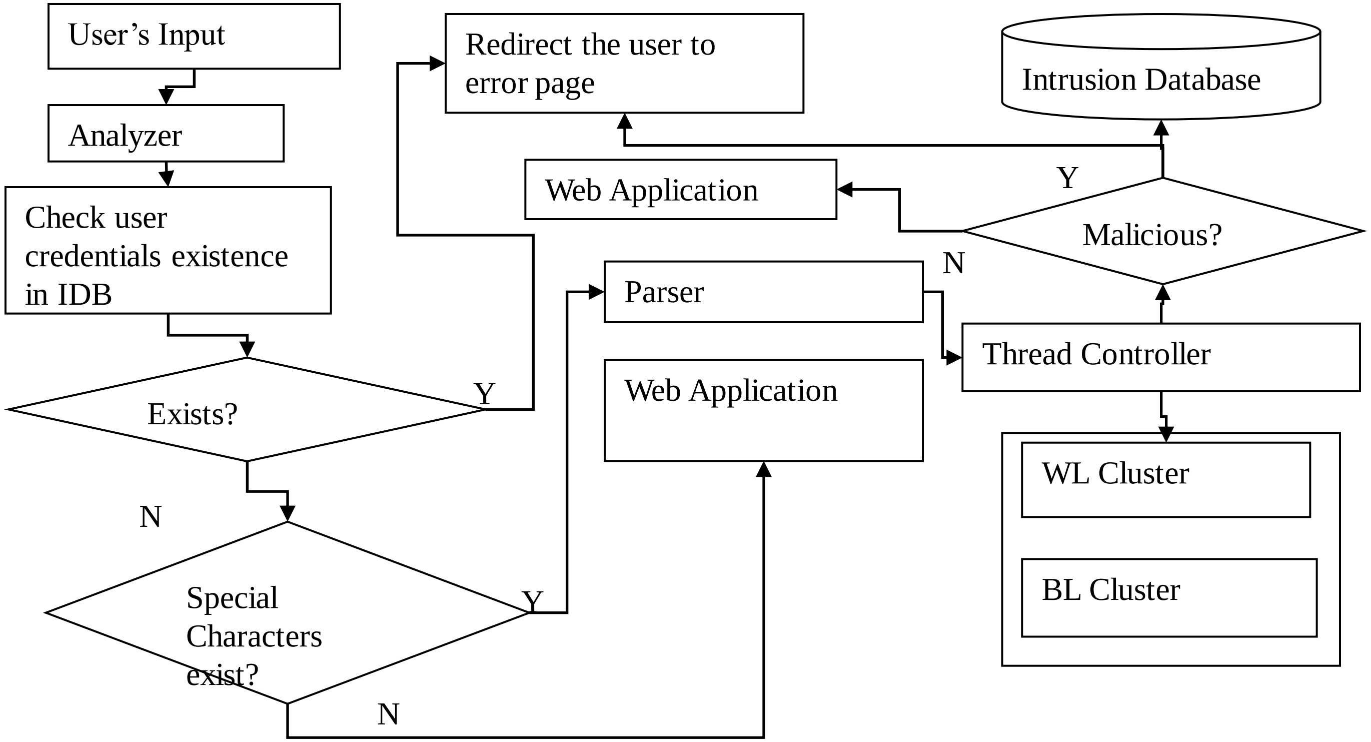
Thread based Intrusion Detection and Prevention System for XSS and Application Worms.

Need for the development of a thread based intrusion detection and prevention system:

- When there is a need to detect the application intrusion for XSS vulnerability.
- Web application availability is expected to be high.
- Chances are high that the web application can be affected by web application worms.

Thread based Intrusion Detection and Prevention System for XSS and Application Worms– Technical details.

Flow of user input through components.



Thread based Intrusion Detection and Prevention System for XSS and Application Worms– Implementation details.

- Introduced application parameters for IDB.
- States are defined to identify the intensity of the threat.
- Proposed URL based, User based, IP based, session based blocking mechanisms.

Thread based Intrusion Detection and Prevention System for XSS and Application Worms– Evaluation details.

- Application performance is assessed without the application of security layer and with the application of security layer.
- Both the approaches are tested with 6000 malicious inputs, 5000 non vulnerable inputs. The average time has been taken for 10 cycles of execution of each approach.
- Optimized the detection of vulnerability by sorting the tags in black listed cluster based on the vulnerable input collected.
- Compared with other products available in market namely striptags, PHP Input Filter, HTML Safe, Safe HTML checker, and HTML Purifier on number of aspects.

Experimentation Results:

	Vulnerable input processing time in milliseconds to process 6000 vulnerable inputs	Non vulnerable input processing time in milliseconds to process 6000 inputs with white listed tags	Random generator program test for 6000 inputs, represented in milliseconds.
Thread based approach after applying the security mechanisms	2500	1400	1890
Without the thread based approach	2000	1000	1500

Thread based Intrusion Detection and Prevention System for XSS and Application Worms – Merits.

- This solution combines the positive security model and negative security model to reduce the processing time.
- Protects the application from zero-day attacks.
- Parameters are introduced to identify the intrusion on the server side and prevention mechanisms are proposed.

Thread based Intrusion Detection and Prevention System for XSS and Application Worms – Limitations.

- Security mechanism needs an update in the clusters when a new tag or attribute or a method is introduced. When a new threat is introduced, the black listed clusters need an update.

Improved trust metrics and variance based authorization model in e-Commerce to prevent fake transactions.

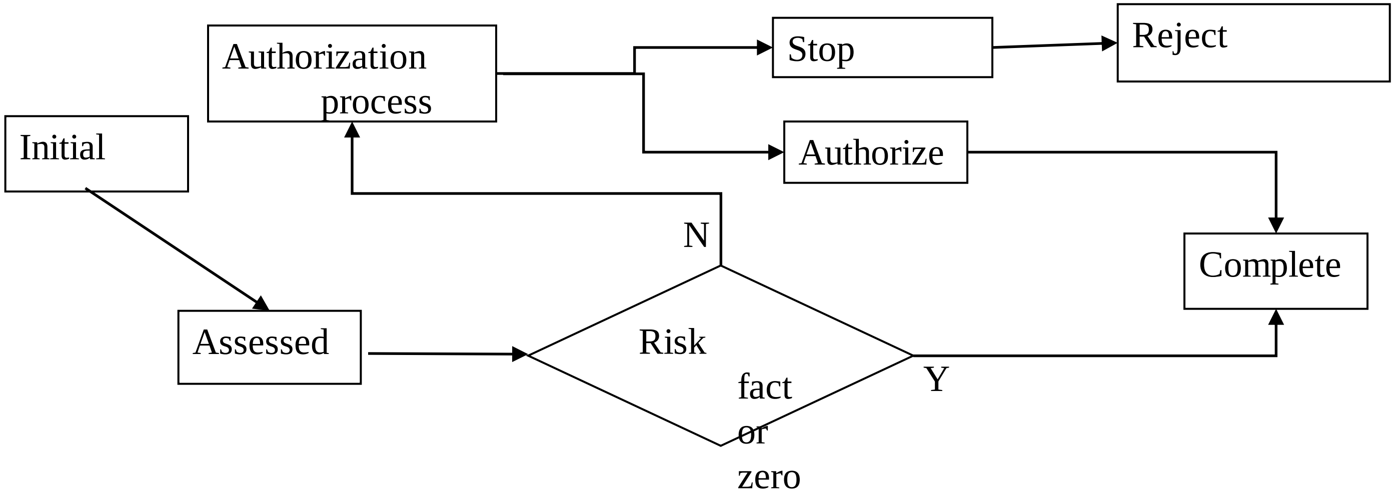
- Trust metrics namely cost, location, frequency of transactions and password reset history are introduced.

- Standard deviation based model to assess the transaction deviation.

- Authorization process flow and authorization levels proposed.

Improved trust metrics and variance based authorization model in e-Commerce to prevent fake transactions – Functional flow.

- Functional flow diagram of the transaction states



Improved trust metrics and variance based authorization model in e-Commerce to prevent fake transactions - Merits.

- Redefining the location trust metric and including Password reset history in trust metric over comes the risk encountered by the earlier trust metrics based authorization techniques.
- The model is not prone to contour analysis since parameter analysis takes place in a secured internal environment of e-commerce network. The proposed authorization model will save the e-commerce transactions from the hands of hackers.

Improved trust metrics and variance based authorization model in e-Commerce to prevent fake transactions - Limitations.

- Introduced metrics should be collected for each customer.
- For each transaction the deviation is calculated and hence the processing time would increase.

Conclusion and Contributions of this research.

In this research, we addressed XSS vulnerabilities and anti-XSS solutions are proposed systematically.

- Proposed a Service Oriented Architecture based solution to prevent XSS vulnerabilities for the web applications developed in different languages.
- In addition, SOA based solution addresses the XSS vulnerabilities that arise from other input sources apart from the web browsers.
- The security solutions are proposed for financial and non-financial web applications and further the solutions are based on the need for which the web application is built.

… Continued

Conclusion and Contributions of this research - Continuation.

- XSS threats are categorized under four heads namely HTML element attack, Character encoding attack, embedded character attack, and event handler attack.

- Application parameters are introduced at the server side and they are characterized with four characteristics namely Severity level, Maximum number of characters allowed, encoding and character-set to address the varied nature of web application.

- Configurable method is introduced to protect the application from zero-day threats.

… Continued

Conclusion and Contributions of this research – Continuation.

- In earlier contributions web pages are modified to incorporate the security mechanisms at a page level. In this work, clusters are introduced at the server side to eliminate the need for modifying the web pages when a threat is introduced.

- Behavior based anomaly detection is proposed to improve the performance for HTTP requests.

- Intrusion Detection Parameters and Intrusion Detection States are elicited at application level.

… Continued

Conclusion and Contributions of this research – Continuation.

- Blocking mechanisms are proposed to block hacker's evasion mechanisms.

- Defined a new trust metric, Password reset history parameter in addition to the trust metrics defined in the earlier works and suggested a new approach using these parameters.

- Authorization levels are suggested to identify the bogus transactions at the server side.

… Continued

Conclusion and Contributions of this research – Continuation.

- Control limit is purported and a risk factor is defined to assess the deviation of the trust parameters.

- For Authorization of transactions, the layers primary, Intermediate and Terminal layers are introduced.

- Payment Verification Matrix is described by newly defined risk factor values mapped to the trust parameters to derive authorization level.

Further Scope of Research.

- This research addresses the current XSS vulnerabilities and if new XSS attacks are introduced then new anti-XSS techniques should be proposed.
- Web applications need to be protected against the XSS evasion mechanisms that receive input from various interfaces.
- New algorithms can also be proposed to process the attack vectors to decrease the processing time of user input in the web applications.

List of Publications and Presentations.

1. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, "Improved Trust Metrics and Variance Based Authorization Model In E-Commerce", in Proceedings of Advances in Intelligent Web Mastering, Proceedings of the 5th Atlantic Web Intelligence Conference – AWIC'2007, published in Journal: Advances in Soft Computing, ISBN 978-3-540-72574-9, Springer, France, pp. 322-328, June 25–27, 2007.**

2. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko "A Solution to Block Cross Site Scripting Vulnerabilities Based on Service Oriented Architecture", in Proceedings of 6th IEEE international conference on computer and information science (ICIS 07) published by IEEE Computer Society in IEEE Xplore, ISBN: 0-7695-2841-4, Australia, pp. 861-866, July 11-13, 2007.**

3. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, "XSS Application Worms: New Internet Infestation and Optimized Protective Measures", in Proceedings of 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007), published by IEEE Computer Society in IEEE Xplore, ISBN: 978-0-7695-2909-7, China, Volume 3, pp. 1164-1169, July 30 - Aug 1, 2007.**

… Continued

List of Publications and Presentations – Continuation.

4. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, "Risk Mitigation for Cross Site Scripting Attacks Using Signature Model on the Server Side", in Proceedings of Multi Symposiums on Computer and Computational Sciences 2007 (IMSCCS07), published by IEEE Computer Society in IEEE Xplore, ISBN: 978-0-7695-3039-0, Iowa, USA , pp. 398-405, August 13-15th 2007.**

5. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, "Behavior-Based Anomaly Detection on the Server Side to Reduce the Effectiveness of Cross Site Scripting Vulnerabilities", in Proceedings of 3rd IEEE International Conference on Semantics, Knowledge, and Grid, published by IEEE Computer Society in IEEE Xplore, ISBN: 978-0-7695-3007-9, China, pp. 350-353, October 29-31 2007 .**

6. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, "Thread Based Intrusion Detection and Prevention System for Cross Site Vulnerabilities and Application Worms", in Proceedings of 1st International Conference on Data Engineering and Management (ICDEM 2008), ISBN 978-81-906267-0-5, Trichirapalli, India, pp. 244-247, February 09th 2008 .**

… Continued

List of Publications and Presentations – Continuation

7. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “Improved Server Side Solution for Mitigating Cross Site Scripting Attacks for Variety of Web Applications”, in Proceedings of 1st International Conference on Data Engineering and Management (ICDEM 2008), ISBN 978-81-906267-0-5, Trichirapalli, India, pp. 248-251, February 09th 2008.

Published National Conference Papers:

8. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “A Framework for Fast Web Based Application Development Using MVC and AJAX”, published in the Research Papers on Advanced Networking Technologies and Security Issues, in Proceedings of AICTE Sponsored National Seminar on Advanced Networking, Technologies and Security Issues (FISAT) conference, Kerala, pp. 177-183, August 8th – 10th 2007.

9. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “Server Side Solution to Prevent Zero-Day Cross-Site Scripting Attacks for Web Applications”, published in the Research Papers on Advanced Networking Technologies and Security Issues, in Proceedings of AICTE Sponsored National Seminar on Advanced Networking, Technologies and Security Issues (FISAT) conference, Kerala, pp. 150-158, August 8th – 10th 2007.

… Continued

List of Publications and Presentations – Continuation.

10. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “A Server Side Solution to Block Cross Site Scripting Vulnerabilities Based on XML and XSD”, published in the Research Papers on Advanced Networking Technologies and Security Issues, in Proceedings of AICTE Sponsored National Seminar on Advanced Networking, Technologies and Security Issues (FISAT) conference, Kerala, pp. 159-170, August 8th – 10th 2007.

11. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “Web Application Worms-Latest Developments and Solutions: A Survey”, in Proceedings of the National conference on Information technology: Present practices and challenges, New Delhi, pp. 250-255, August 31st-Sep 1st 2007.

12. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “An Assessment on Prevention Mechanisms for XSS Vulnerability Based on Detection Software Types”, in Proceedings of the National conference on Information technology: Present practices and challenges, New Delhi, pp. 243-249 August 31st-Sep 1st 2007.

13. Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “Intrusion Detection and Prevention System for Cross site vulnerabilities based on Negative Security Model”, in Proceedings of the National Conference on Advanced Data Computing Communications and Security, ENVISION - 2007, All India Council for Technical Education (AICTE) sponsored National Conference, Gujarat, pp. 269-276, October 2007.

List of Publications and Presentations – Continuation.

Articles accepted for Publication:

14. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “Cross Site Scripting-Latest developments and solutions: A survey”, accepted for publication in the International Journal of Open Problems in Computer Science and Mathematics (IJOPCM).**

15. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “A Novel Method to Mitigate Filter Evasion Mechanisms for Cross Site Scripting Threats”, accepted for publication in the International Journal on Computer Science and Information Technology (IJCSIT).**

Article submitted for review:

16. **Jayamsakthi Shanmugam, Dr. M. Ponnavaikko, “A Scalable Approach to Secure the Web Applications from Cross Site Scripting Threats”, submitted for review in the International Journal of Information Technology (IJIT).**

ACKNOWLEDGEMENTS.

Supervisor : Dr. M. Ponnavaikko,

Vice Chancellor, Bharathidasan University, Trichy.

Examiners:

Dr. Karbhari Vishwanath Kale.

Prof. N. V. Muralidhar Rao.

Dr. Vasudha Bhatnagar.

Prof. L. K. Maheshwari, Vice-Chancellor (BITS) and Director (Pilani campus).

Prof. Ravi Prakash, Dean, Research & Consultancy Division, BITS, Pilani.

Dr. Rahul Banerjee.

DAC Members of BITS Pilani.

PhD Monitoring Group of BITS Pilani.

Dr. Sharad Srivastava.

Dr. Dinesh Kumar.

Dr. Srinivasa Prakash Regalla.

Dr. S. D. Pohekar.

Thank you.