Comparative Evaluation of Vulnerabilities using Fuzzy-ELM Approach

Narendra M Kandoi
Research scholar, computer Science & Engineering
S.G.B. Amravati University Amravati,
Maharashtra, India

Dr. Vilas M Thakare
Research Guide, S.G.B. Amravati University Amravati,
Maharashtra, India

***Abstract*— A web application is comprised of a social event of contents, which live on a web server and interface with databases or distinctive sources of dynamic substance. The security of a web application is mainly depends on the integrity and privacy of its assets inside the web program. For instance, session identifiers in cookies should be secured against access by untrusted principals. Web applications are very presented to attacks from anyplace in the environment, which can be led by utilizing broadly accessible and basic devices like a web program. It is normal to discover web application designers, power engineers and managers without the required learning and involvement in the security space. In this paper, the comparative analysis performed on recall and precision value for Fuzzy-ELM, SVM, Random Forest and Naïve bayes. Different vulnerabilities used for the illustration of recall and precision to show the efficiency.**

Keywords—Software Vulnerability, Fuzzy-ELM, SVM, Random Forest, Naïve bayes, Recall, Precision.

I. INTRODUCTION

The web application security and web mash up development for providing the security to web applications using new proposed methods. In this research work, the main focus is given on the Aspect-Oriented programming (AOP) that supplied the power to modularize cross-cutting issues and security in a software gadget. The system provided right here, provides the benefits of AOP i.e. Simultaneous parallel decay of orthogonal concerns that is termed as application transparency and another gain is that the software program programmers and designers would consciousness be able to on their particular circumstance. One of the maximum common issues is modularity and protection of web software and web mashup which turn out to be one of the most up to date buzzwords in the net applications area, and many of agencies and institutions are dashing to offer mashup answers The goal attempted in this research work is to use AOP as one of the major tool to design a unique security framework that even if the developer had not considered security as a one of the component of web application in the beginning, at a later stage it should be free (safe) from major vulnerabilities and attacks [1].

The Separation of worries is a recognized technique for the department of a software program mission's problem domain into numerous different factors known as modules. Latest programming languages like OOP make it hard, and occasionally not possible, to isolate certain worries into conceptual modules for next translation into pc code. Protection is one challenge that cannot be distributed by traditional methods as it has a tendency to get tangled with the alternative code in a software program device. Aspect - orientated programming (AOP) makes it viable to isolate this and different worries that have been previously inseparable into modules. It is crucial to realize the number of the history and motivation behind the usage of thing-oriented programming (AOP) to individual a protection concern from the enterprise good judgment in an organization environment [1]. And also the aims and objectives of the proposed research work in constructing an AOP based software system to develop concrete methodology a way to separate the security aspect from the main logic of the system [2].

Internet software vulnerabilities are tough to remove due to the fact most web applications a) Go through fast development levels with extremely short turnaround time, and b) Are developed in-residence by using company MIS engineers, maximum of whom have less education and experience in comfy software program development. ultimately, present day technology which includes anti-virus software program and network firewalls offer comparatively at ease safety on the host and network stages, however now not at the application stage whilst network and host-degree access points are distinctly at ease, the public interfaces to internet applications become the focal point or targets of attacks [2].

II. COMPARATIVE STUDY

Table 1 Comparative Study

Paper Title	Methodology	Advantages	Disadvantages
Aspect Oriented Programming to Improve Modularity of Object-Oriented Applications [3].	AOP, seclusion, crosscutting concerns, AspectJ, partition of concerns.	AOP gives an instrument to enhance the modularization of question arranged applications. It evades the code tangling and dispersing issues caused by crosscutting concerns.	A few concerns can't be specifically modularized in great question situated languages in light of the fact that those dialects have not sufficient expressiveness to actualize them in autonomous modules.

Utilizing Aspect Programming to Secure Web Applications [4].	AOP security, SQL infusion, cross website scripting, outline of web applications, reuse of aspect, dynamic weaving.	AOP is a decent candidate for this component.	Each time the application issues a database call, the question is approved to forestall startling inquiries to execute, maintaining a strategic distance from " always true" conditions and the utilization of semicolons and remarks in the query.
Security and Communication Abstractions for Web Browsers in Mashup OS [5].	Program, Web, same-root approach, assurance, interchanges, security, multi-primary OS, abstractions.	Strike a balance between usability and security. Match all normal confide in levels:	In the colossal measure of data the tedious to coordinate the all levels.
A Permission System for Secure AOP.	Language based security, viewpoint arranged programming, consent framework, execution history.	The essential objectives of AOPS is to control and limit the mischief that advices of untrusted perspectives can cause at run time by upholding approaches on these angles.	To address these issues, we exhibit an approach of runtime arrangement authorization, in light of execution history.

III. WEB APPLICATION AND ITS SECURITY

Web application is a software program which is put away on the Server and got to through an any web program like Chrome, Firefox, IE, Edge, Safari, and so on.

Web applications can be individual sites, blogs, social networks, news, web mails, bank offices, forums, e-commerce business applications and so on. The ubiquity of web applications in our way of life and in our economy is critical to the point that it makes them a trademark center for harmful identities that need to manhandle this new streak [6]. A web application is comprised of a social event of contents, which live on a web server and interface with databases or distinctive sources of dynamic substance.
Essential objectives of web application security are it is planned to prevent.

- Unauthorized personnel from getting to data at higher grouping than their approval
- To keep personnel from declassifying data.

Utilizing the framework of the Internet, web applications permit service providers and clients to share and control data in a platform-independent way. The advancements used to assemble web applications consolidate PHP, Active Server Pages (ASP), Perl, Common Gateway Interface (CGI), Java Server Pages (JSP), JavaScript, VBScript, et cetera. A few general classes of web application innovations are positions, correspondence conventions, customer side and server-side scripting dialects, web server API and program modules [7]. The Web Application Security Consortium (WASC) characterizes a web application as "a software application, carried out by a web server, which reacts to dynamic website page requests for over HTTP" [8]. A web application has a circulated n-layered engineering. Normally, there is a customer (web program), an application server (or a couple of utilization servers), a web server and an persistence (database) server. Fig.1 represents a sampled perspective of a web application. There might be a firewall between web client and web server [9].

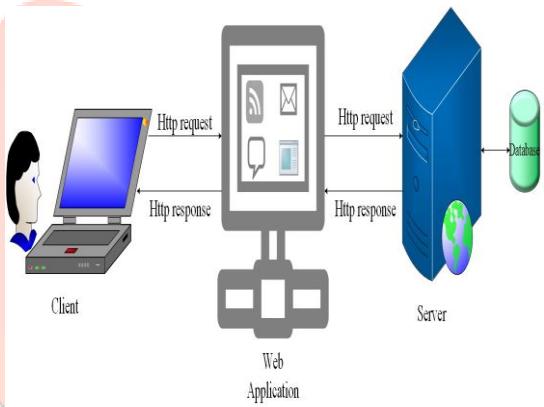


Fig.1 Web Environment

The security of a web application is mainly depends on the integrity and privacy of its assets inside the web program. For instance, session identifiers in cookies should be secured against access by untrusted principals; code from untrusted sources must be authorized before it is permitted to change any trusted content on a web page. Without appropriate access control in web applications, we can't spare the trustworthiness of substance and security could be compromised [10].

On the off chance that we consider each site page as a "system" we require a satisfactory protection model in programs to intercede the associations inside such a framework. The security inspiration of web application engineers and directors ought to reflect the magnitude and importance of the benefits they should ensure [11]. In spite of the fact that there is an expanding issue about security (frequently being liable to directions from governments and companies), there are critical elements that make securing web applications a troublesome work to accomplish:

1) The web application market is nourishing quickly, bringing about a gigantic multiplication of web applications, in light of various frameworks, languages and protocols generally energized by the (evident) simplicity one can create and keep up such applications.
2) Web applications are very presented to attacks from anyplace in the environment, which can be led by utilizing broadly accessible and basic devices like a web program.
3) It is normal to discover web application designers, power engineers and managers without the required learning and involvement in the security space.

4. Web applications give the way to get to profitable endeavor resources. Commonly they are the primary interface to the data put away in back-end databases; different circumstances they are the way to within the enterprise network and computers [12]. Of course, the general circumstance of web application security is very positive to attacks. Indeed, estimations point to an extensive number of web applications with security vulnerabilities and thus, there are several reports of effective security breaks and abuses. Organized crime is normally thriving in this promising business sector, on the off chance that we consider the millions of dollars earned by such associations in the underground economy of the web [13]. To battle this situation we require intends to assess the security of web applications and of attack counter measure instruments. To deal with web application security, new apparatuses should be produced and strategies must be enhanced, updated or designed [14]. In addition, everybody engaged with the improvement procedure ought to be altogether assessed, checked and authorized before going into production. Be that as it may, these best procedures are unfeasible to apply to a huge number of already available heritage web applications, so they ought to be continually examined and ensured by security devices when their lifetime [15]. This is especially significant because of the extraordinary dynamicity of the security situation, with new vulnerabilities and methods for exploitation being found each day. Obviously, security technology is sufficiently bad to stop web application security attacks and experts should be concerned about the assessment and the affirmation of their success. Practically speaking, there is a requirement for better approaches to viably test existing web application security methods keeping in mind the end goal to evaluate and enhance them [16]. Web application vulnerabilities are difficult to dispense because the most Web applications experience quick development platforms with greatly short turnaround time and are created in-house by corporate MIS engineers, many of whom have less training and involvement in secure software development [17]. In conclusion, current advances, for example, system firewalls and antivirus software offer similarly secure assurance at the host and network levels, yet not at the application level when network and host-level section focuses are moderately secure, the public interfaces to Web applications turn into the target or focus of attacks [18].

IV. LITERATURE SURVEY

Taivalsaari A *et al.* [19] Authors refers the security lattice-based approach to mashup security, where origin of a distinct elements of the mashup used the levels in security of lattice. Classification allows the controlled information release during the elements. They formalize an origination of blended delimited convey arrangement and provided presumptions of the commonsense (static and in addition runtime) implementation of mashup data stream security policies in the web browser.

Abhijit Sanyal *et al*. [20] the author objective of paper is represents that Aspect Oriented Programming AspectJ combined with a Spring AOP supplied very powerful mechanism for the stronger enforcement of a security. AOP permits meshing the security angle into the application providing the extra security usefulness or presenting totally new security instrument. Implementation of the security with AOP is an adaptable strategy to create isolated, extensible and reusable cut of code called viewpoints. In this relative analysis of paper, they contend the Spring AOP provided stronger implementation of security than AspectJ. They have shown the both Spring AOP and AspectJ attempt to supply a comprehensive AOP solution and supplements to each other.

Faisal Anwer *et al*. [21] this paper authors presents a way of securing a system without any knowledge of the system source code. The security module adds to the current framework validation and authorization in view of perspective situated programming and the freedom union structure, a forthcoming industry standard giving single sign on. In an underlying preparing stage the module is adjusted to the application which is to be secured. In addition the utilization of equipment tokens and proactive figuring is illustrated. The high modularization is achieved through utilization of AspectJ, a programming dialect augmentation of Java.

W. Huang, *et al.* [22] they have characterized and prototyped a run-time approach requirement show in view of execution history to shield programs from untrusted angles. The dynamic nature of the approach has the advantage that up and coming run-time data permits more precise decision making. They have assembled a model for AspectJ and show its utilization in a reasonable illustration. Their evaluation shows that practical use of such a solution is feasible and that run-time overhead can be limited.

Y. Xie *et al*. [23] this paper creator displays a perspective situated system, to execute the Principle of Least Privilege on

work area clients. This attempt to ensure that illegitimate - access is blocked and legitimate resource access to is permitted. They try to discuss a case study applying his approach on desktop applications such as a Web browser and an RSS feed aggregator.

N. Jovanovic *et al*. [24] have proposed a graph-based aspect-oriented conceptual data model, `Semantic Graph Data Model' (SGDM), for electronic application. This not only offers a pictorial view with better understand capacity but also gives the procedure for describing the semantic sources. SGDM demonstrates that in an organized way with thinking through angle introduction. The proposed demonstrate is backings to the programmed space show improvement for same business issues by viable domain-knowledge reuse. The SGDM supports the object-oriented paradigm and also flexible to displaying the semi-structured as well as hypertext information through a multi layered graph structure. The Product Specific Model (PSM) SQL: 2003 implementation of our proposed model SGDM is being done through GME tool.

Y. Minamide et al. [25] the diverse methodologies of web service compositions this is created by the researcher; Aspect Oriented Web Service Composition (AOWSC) is the most formal method for arrangement. Composition is an arrangement of same competitor web benefit, synchronously tied up with each other, conveyed to the requestor at run time in light of the service request. In standard with perspective introduction, the competitor web services are created as angles, runtime chose at run time this depends on the service request, formed through a weaving component and along these lines handle composition. The weaving instrument is the core of AOWSC, as it determines the request in that hopeful web benefit must be synchronously appears concerning some other.

V. B. Livshits et al. [26] Author proposed utilization of perspective introduction to manage QoS over the existing web server, Jigsaw. They show how the aspect-based architecture can be effective for the supplying the web server with different improvements when preparing approaching request. QoS parameters have been overseen by associating requests with advantaged and by bringing into the web server mind resource utilization and procedure to execute. Suggested perspectives are associated with the gather time to existing classes, consequently keeping the QoS enforcing code not quite the same as the web server modules.

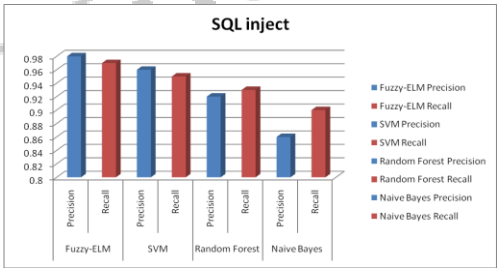
Jonas Magazinius et al. [27], Authors allude the security lattice section based way to deal with mashup security, where starting point of a particular components of the mashup utilized the levels in security of lattice. Classification permits the controlled data discharge amid the components. They formalize an origination of mixed delimited deliver policy and provided assumptions of the useful (static and additionally runtime) implementation of mashup data stream security strategies in the web browser.

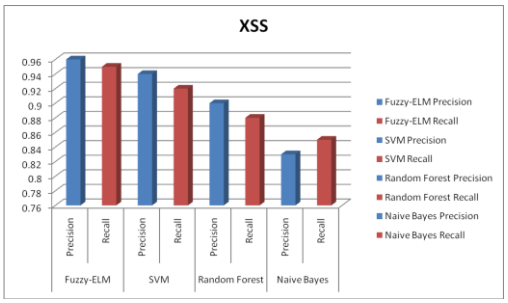
Chen Yanhun et al. [28], the creator objective of paper is speaks to that AOP AspectJ joined with a Spring AOP provided capable system for the more grounded implementation of a security. AOP permits meshing the security perspective into the application providing the extra security usefulness or presenting totally new security component. Implementation of the security with AOP is an adaptable strategy to create isolated, extensible and reusable cut of code called aspects. In this similar analysis of paper, they contend the Spring AOP provided more grounded authorization of security than AspectJ. They have demonstrated the both Spring AOP and AspectJ endeavor to supply an exhaustive AOP arrangement and supplements to each other.

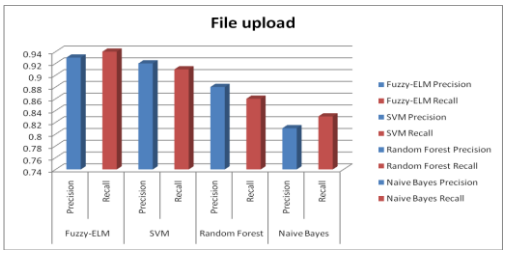
V. RESULT ANALYSIS

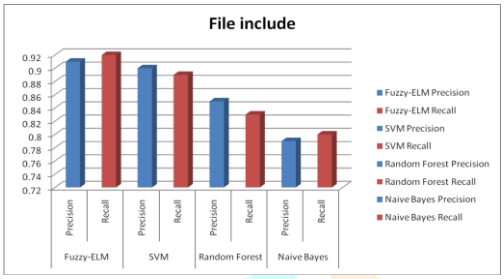
The proposed approach has been experimented on one online application case study program in that show that the development of a tool permits to free the developers from tedious and error-prone tasks since they have just to push a button to generate the AspectJ code of an application which consists of several hundred of lines. Moreover, the application of the proposed approach on the cases studies permitted to point out security rules violations that are related to operation sequencing and also the occurrence of a forbidden operation before the execution of the corresponding secure one. What differs from one case study to another is the number of classes, associations, attributes and rules that implies more/less JAVA classes to generate; the complexity of the obtained code remains similar.

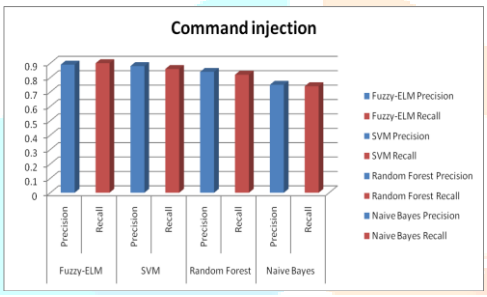
From the graphs below show the Fuzzy-ELM classifier efficiently classify the vulnerabilities compared to the other classifiers such as fuzzy-ELM, SVM, Random Forest, Naive Bayes.

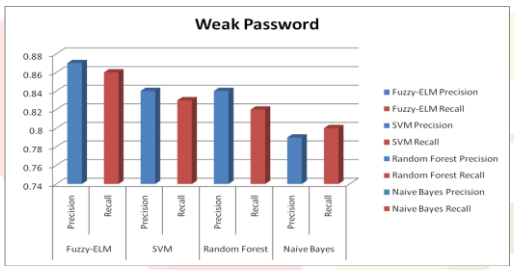












VI. CONCLUSION

Web application vulnerabilities are difficult to dispense because the most Web applications experience quick development platforms with greatly short turnaround time and are created in-house by corporate MIS engineers, many of whom have less training and involvement in secure software development. Protection is one challenge that cannot be distributed by traditional methods as it has a tendency to get tangled with the alternative code in a software program device. AOP makes it viable to isolate this and different worries that have been previously inseparable into modules. It is crucial to realize the number of the history and motivation behind the usage of AOP to individual a protection concern from the enterprise good judgment in an organization environment

References

[1] Heba Kurdi A. Computer Science Department Imam Muhammad Ibn Saud Islamic University Riyadh, Saudi Arabia. Review on Aspect Oriented Programming" International Journal of Advanced Computer Science and Applications. 2013; 4(9):22

[2] Jose Felix M. Principality of Asturias, Computer Science Department, Oviedo, Spain Francisco Ortin, University of Oviedo, Computer Science Department, Oviedo, Spain, Aspect-Oriented Programming to Improve Modularity of Object-Oriented Applications. Journal of Software. 2014; 9(9). doi:10.4304/jsw.9.9. Pages 2454-2460.

[3] Wang HJ, Fan X, Howell J, Jackson C. Protection and Communication Abstractions for Web Browsers in MashupOS Proc. 21st Int'l Conf on O.S. USA. 2007; 14(17):1-16.

[4] Jim T, Swamy N, Hicks M. Defeating Script Injection Attacks with Browser-Enforced Embedded Policies. Proc. Of 16th Int'l Conf on World Wide Web Canada. 2007; 8(12):601-610.

[5] Jackson C, Wang H. Subspace: Secure Cross-Domain Communication for Web Mashups. Proc. 16th Int'l Conf on World Wide Web Canada. 2007; 8(12):611-619.

[6] Fonseca, Jose, Marco Vieira, and Henrique Madeira, "Evaluation of web security mechanisms using vulnerability & attack injection", IEEE Transactions on Dependable and Secure Computing, Vol. 11, No.5,pp. 440-453, 2014.

[7] Fong, Elizabeth, and VadimOkun, "Web application scanners: definitions and functions", In System Sciences, 40th Annual Hawaii International Conference, IEEE, pp. 280b-280b, 2007.

[8] Black, Paul E., Elizabeth Fong,VadimOkun, and RomainGaucher, "Software assurance tools: Web application security scanner functional specification version 1.0", Special Publication,pp. 500-269, 2008

[9] Guinard, Dominique, VladTrifa, StamatisKarnouskos, PatrikSpiess, and DomnicSavio, "Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services", IEEE transactions on Services Computing, Vol. 3, No. 3,pp. 223-235, 2010.

[10] Bansal, Chetan, KarthikeyanBhargavan, Antoine Delignat-Lavaud, and Sergio Maffeis, "Discovering concrete attacks on website authorization by formal analysis1", Journal of Computer Security, Vol. 22, No. 4,pp. 601-657, 2014.

[11] Fonseca, José, Marco Vieira, and Henrique Madeira, "Vulnerability & attack injection for web applications", In Dependable Systems & Networks, IEEE/IFIP International Conference, pp. 93-102, 2009.

[12] Uckelmann, Dieter, Mark Harrison, and Florian Michahelles, "An architectural approach towards the future internet of things", Architecting the internet of things, Springer Berlin Heidelberg, pp. 1-24, 2011.

[13] Sharma, Aashish, ZbigniewKalbarczyk, James Barlow, and RavishankarIyer, "Analysis of security data from a large computing organization", In Dependable Systems & Networks (DSN), IEEE/IFIP 41st International Conference, pp. 506-517, 2011.

[14] Murugesan, San, "Web application development: Challenges and the role of web engineering", Web engineering: modeling and implementing web applications, pp. 7-32, 2008.

[15] Creech, Gideon, and Jiankun Hu, "A semantic approach to host-based intrusion detection systems using contiguous and discontinuous system call patterns", IEEE Transactions on Computers, Vol. 63, No. 4,pp. 807-819, 2014.

[16] Cole, Eric,"Network security bible", Vol. 768, John Wiley & Sons, 2011.

[17] Alzahrani, Abdulrahman, Ali Alqazzaz, Ye Zhu,Huirong Fu, and Nabil Almashfi, "Web Application Security Tools Analysis", In Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing (HPSC), IEEE International Conference on Intelligent Data and Security (IDS), IEEE 3rd International Conference, pp. 237-242, 2017.

[18] Huang, Yao-Wen, Fang Yu, Chung-hung Tsai, Christian Hang, Der-Tsai Lee, and Sy-Yen Kuo, "System and method for securing web application code and verifying correctness of software", U.S. Patent No. 8,555,269, 2013.

[19] Taivalsaari A, Mikkonen T. Mashups and modularity: Towards secure and reusable web applications Proc. Int'l Conf. Publication Year, 2008, 25-33.

[20] Abhijit Sanyal, Sankhayan Choudhury. An Aspect-oriented Conceptual Level Design for Semantic Web based Application, International Conference on Computer & Communication Technology ICCCT. 2011; 352(357):978-1-4577-1386-611.

[21] Faisal Anwer, Mohd Nazir, Khurram Mustafa. Safety and Security Framework for Exception Handling in Concurrent Programming, Third International Conference on Advances in Computing and Communications, 2013, 308-311. Month, year DOI 10.1109/ICACC.2013.65

[22] Y.-W. Huang, F. Yu, C. Hang, C.-H. Tsai, D.-T. Lee, and S.-Y. Kuo, "Securing web application code by static analysis and runtime protection," in WWW'04: Proceedings of the 13th international conference on World Wide Web, 2004, pp. 40–52.

[23] Y. Xie and A. Aiken, "Static detection of security vulnerabilities in scripting languages," in USENIX'06: Proceedings of the 15th conference on USENIX Security Symposium, 2006.

[24] N. Jovanovic, C. Kruegel, and E. Kirda, "Pixy: A static analysis tool for detecting web application vulnerabilities (short paper)," in Oakland'06: Proceedings of the 27th IEEE Symposium on Security and Privacy, 2006, pp. 258–263.

[25] Y. Minamide, "Static approximation of dynamically generated web pages," in WWW'05: Proceedings of the 14th international conference on World Wide Web, 2005, pp. 432–441.

[26] V. B. Livshits and M. S. Lam, "Finding security vulnerabilities in java applications with static analysis," in USENIX'05: Proceedings of the 14th conference on USENIX Security Symposium, 2005, p. 18.

[27] Jonas Magazinius. Andrei Sabelfeld Chalmers, Aslan Askarov Cornell University: A Lattice-based Approach to Mashup Security Proc of Int'l conf ACM, 2010, 15-23.

[28] Chen Yanhun, Wang Xingpeng. A Security Risk Evaluation Model for Mashup Application "Proc. Int'l Conf on Information Management, Innovation Management and Industrial Engineering Publication Year, 2009, 212-215.