A Survey and Vital Analysis of Various State of the Art Solutions for Web Application Security

Anna Thankachan[1], R. Ramakrishnan[2], M.Kalaiarasi[3],
PG Scholars Sri Manakula Vinayagar Engineering College (Puducherry) [1, 3]
Associate Professor Sri Manakula Vinayagar Engineering College (Puducherry) [2]
annathankachan@gmail.com[1], rr_mca@sify.com[2], majorkalai@yahoo.com [3]

*Abstract***—** ***For the past many years, web service is gaining an innovative momentum as a computing paradigm for providing flexible and dynamic services on demand. Web services and applications hold the potential to transform the landscape of internet services and information by making more attractive and creative which enhanced the productivity of e-business. On the other hand however, the number of security vulnerabilities and dangerous security flaw has increased immensely due to its growing popularity and global distribution. To address these risks, Web services require increased security protection. Web application security protects the confidentiality, integrity and availability of resources in cyberspace .It is important to maintain a high level security to ensure safe and trusted communication of information between various organizations. This paper surveys the area of web application security, the characteristics of some of the best known techniques of art solutions for web application security and their weakness***

Keywords—Web services, Web application, Vulnerability, Web security, Security properties

I. INTRODUCTION

Web services as an emerging information technology, it puts forward a new kind of information system common framework based Internet, it allows users remote call information among different departments information systems and process the data in the TCP/IP environment. Web services with the XML standard form a loose coupling application integration method, make the enterprise system form a kind of flexible business model, can adapt to the changing business environment [58]. Web services communication uses SOAP message as carrier. Therefore, Web services security communication must ensure the SOAP message end-to-end security .Web Services Security should meet five basic objectives: identity authentication, permission Management, data integrity, confidentiality and anti- denial. Identity authentication is an authentication process in order to enable. an entity confirms that the other one whether it is what he claimed entity; permission Management ensures that user which has the appropriate access permission will be able to access the appropriate services; data integrity confirms that whether data had been disposed, it prevents data of

There are many ways to protect a web application, such as implementing a secure coding practice, managing secure configuration, performing vulnerability assessment and deploying a web application firewall, but there is no silver bullet that it will protect the application entirely. Web applications need a defense-in-depth approach to avoid and mitigate security vulnerabilities. This approach assumes that every security precaution can fail, so security depends on having several layers of mechanisms that cover the failures of each other.

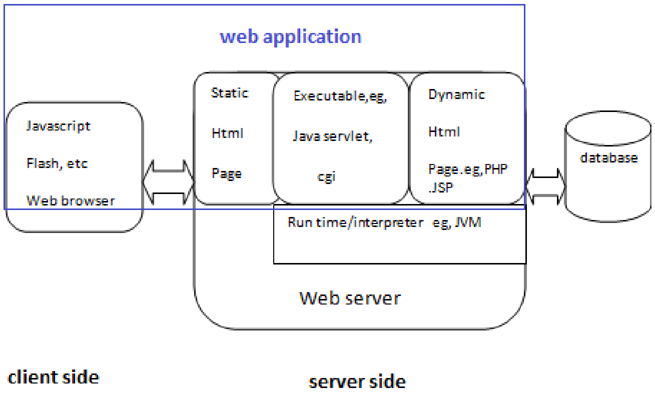

Fig 1.Web Application Overview

The Web application stage is a composite ecosystem composed of a large number of components and technologies, including HTTP protocol, web server and server-side application development technologies (e.g., CGI, PHP, ASP), web browser and client-side technologies (e.g., JavaScript, Flash). Web application made and acted upon such a complex infrastructure faces intrinsic challenges expressed by the features of those components. Current widely-used web application development and testing frameworks, on the other hand, offer limited security support. Thus secure web application development is an error prone process and requires significant efforts, which could be improbable under time-to-market pressure and for people with insufficient security skills or awareness. As a result, a high percentage of web applications deployed on the Internet are exposed to security vulnerabilities. A known report [60] says that over 80% of the

ISBN No.978-1-4799-3834-6/14/$31.00©2014 IEEE

websites on the Internet have had at least one serious vulnerability. Forced by the urgent need for securing web applications, a considerable amount of research efforts have been dedicated into this problem with a number of techniques developed for hardening web applications and mitigating the attacks. Many of these techniques make assumptions on the web technologies used in the application development and only address one particular type of security flaws; their prototypes are often implemented and evaluated on limited platforms.

The state of the art in web application security, with the aim of analyzing the existing techniques into a big picture that promotes future research is surveyed in this paper. Based on the theoretical security framework by Bau and Mitchell [8], the survey is organized into three components for implementation of the security of a web application. They are system model, threat model and security property. System model explains how a web application works; threat model describes the power and resources attackers possess; security property defines the feature of the web application behavior intended by the developers. This paper contains three phases in web application development, and identifies three levels of security properties. They are: input validity, state integrity and logic correctness. Failure of fulfillment of the above security properties is the main cause of corresponding vulnerabilities. The existing research works is classified into three classes: security by construction, security by verification and security by protection, based on their design principles and assurance of security properties in the web application

II. WEB APPLICATION SECURITY PROPERTIES AND VULNERABILITIES

Under the given threat model, a secure web application has to satisfy desired security properties. The following threat model is usually considered: 1) the web application itself is benign and hosted on a trusted and hardened infrastructure the trust computing base.2) the attacker is able to control or manipulate either contents or the sequence of web requests sent to the web application. a secure web application should conserve security properties. Input validity means the user input should be validated before it can be make use of by the web application; state integrity means the application state should be kept unhampered; logic correctness means the application logic should be executed correctly as planned by the developers. For occasion, if the web application fails to hold the input validity property, a crosssite scripting attack can be raised by the attacker to steal the victim's session cookie. The victim's web session is hijacked and tampered by attacker, resulting in the violation of state integrity property. Many countermeasures have been developed to secure web applications and defend against the attacks. These methods address one or more security properties and instantiate them into concrete security specifications or policies. The existing counter measures are organized into three classes [57]. They are

Security by construction: this class of techniques aim in constructing secure web applications, ensuring that no potential vulnerabilities exist within the applications and preserving the desired security property .These techniques solve security problems from the root .so they are most robust.

Security by verification: this class of techniques aims to verify if the desired security properties hold for a web application and identify potential vulnerabilities within the application. This procedure is also referred to as vulnerability analysis. Efforts have to be then spent to harden the vulnerable web application by fixing the vulnerabilities and retrofitting the application either manually or automatically. Techniques within this class can be applied to both new and legacy web applications.

Security by protection: this class of techniques aims in protecting a potentially vulnerable web application against exploits by building a runtime environment that supports its secure execution. They usually either place safeguards that separate the web application from other components in the Web ecosystem, or instrument the infrastructure components (i.e., language runtime, web browser, etc.) to monitor its runtime behavior and identify potential exploits. These techniques can be independent of programming languages or platforms, thus scale well. However, runtime performance overhead is inevitably introduced.

Security vulnerabilities continue to contaminate web applications, allowing attackers to access sensitive data and exploiting web sites as a hosting ground for malware. Some of the major vulnerabilities are discussed below

A. Injection

Injection flaws, such as SQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query in a web application. The attacker's hostile data can trick the interpreter into executing unintended and hacking commands or accessing unauthorized data [61]

B. Cross Site Scripting (XSS)

XSS allows attackers to execute script in the victim's browser which can hijack user sessions deface web sites, or redirect the user to malicious sites .this type of flaws occur whenever an application takes an untrusted data and sends it to a web browser without proper validation and escaping. [61]

C. Broken Authentication and Session Management

Application functions in a web application related to authentication and session management are often not implemented correctly. In this case it allow attackers to compromise passwords, keys, session tokens, or use implementation flaws to assume other users' identities [61]

D. Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data [61].

E. Cross Site Request Forgery (CSRF)

A CSRF attack forces a logged-on victim's browser to send a forged HTTP request, including the victim's session cookie and any other authentication information, to a vulnerable web application. This allows the attacker to force the victim's browser to generate requests the vulnerable application are legitimate requests from the victim [61].

F. Security Misconfiguration

Security depends on having a secure configuration defined for the application, framework, web server, application server, and platform. All these settings should be defined, implemented, and maintained as many are not shipped with secure defaults.[61]

G. Insecure Cryptographic Storage

Many web applications do not properly protect sensitive data, such as credit cards, SSNs, and authentication credentials, with appropriate encryption or hashing. Attackers may steal or modify such weakly protected data to conduct identity theft, credit card fraud, or other crimes [61]

H. Failure to Restrict URL Access

Many web applications check URL access rights before rendering protected links and buttons. However, applications need to perform similar access control checks each time these pages are accessed, or attackers will be able to forge URLs to access these hidden pages anyway [61]

I. Insufficient Transport Layer Protection

Applications frequently fail to encrypt network traffic when it is necessary to protect sensitive communications. When they do, they sometimes support weak algorithms, use expired or invalid certificates, or do not use them correctly [61].

J. Unvalidated Redirects and Forwards

Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages without proper validation [61]

III. VARIOUS STATE OF ART SOLUTIONS AND CHARACTERISTICS

Majority of the web applications contain security vulnerabilities and treats which permit attackers to exploit them and organize attacks against them. Many efforts have been made to diminish these attacks through various security mechanisms in the form of intrusion detection systems, encryption devices, and web application firewalls. These conventional security solutions frequently apply a signature based approach and are only effective therefore against ''well-known' 'security attacks where signatures are already present in the solution database. Unchecked input validation is a major source of attacks at the web application.

According to the Open Web Application Security Project (OWASP) [61], major out of the current top vulnerabilities are related to input validation. Many costly security breaches to organizations can cause due to the Failure to protect web applications from invalid inputs. Hacking procedures can result in the theft of sensitive data, destruction of web sites, privilege escalation, and unauthorized access to a system. A hacker might be able to inject malicious code to bypass or modify the intended functionality of a program without the knowledge of user. An important role is played by Encoding schemes in misleading an attack vector and commonly facilitates the malicious intention of the hacker. Likewise there are many ways to launch SQL Injection attack.

Conventional or traditional security solutions such as web scanners provide the first line of defense against attack and detect well-known or already known security attacks using threat signatures. But Scanners lack semantics and are thus unable to make intelligent decision upon data leakage or business logic flaws [15]. This regularly results in false alarms to be raised and such approaches also fail to detect critical vulnerabilities [36]. Signature based solutions maintain a white and black list of processes. These contain the signatures of all kind of inputs and also those of malicious attack vectors. The use of such lists requires continuous updating of threat signatures. Failing to do so may result in limited or no protection against zero day attacks and the generation of too many false positive and false negative alerts or alarms [43].

Additionally, most network solutions ignore the payload and only scan the headers of a user request. Network solutions are commonly ineffective against application level attacks due to a lack of information regarding the application context. A huge amount of effort has put and expanded by the security industry to avoid these attacks through various state of the art security mechanisms in the form of scanners, intrusion detection systems, encryption devices, and firewalls. These measures have unfortunately so far been unable to achieve the security level that is necessary for web applications. The characteristics of some of the best known techniques of art solutions for web application security and their weakness are discussed here.

A. Anomaly Based Intrusion Detection Systems (AIDS)

Anomaly based intrusion detection systems analyze the behavior or manners of the input stream against a well-known profile and classify all abnormal behavior as malicious [33].

An anomaly detection technique that can automatically tune and optimize the values of parameters without predefining them through labeling was proposed by Song et al. [46] .The detection ability of such systems however depends deeply upon the training of the data model. A poorly refined model reduces the performance of detection. Anomaly based IDS are generally unable to provide a fool proof solution to application level attacks .This is because such models are also used to secure database servers against SQL Injection attacks. Pinzon et al. [33] has anticipated multi-agent hybrid architecture for detecting SQL Injection attacks in web application. Chen et al. [11], has proposed an unsupervised learning frame- work named the ‘‘community anomaly detection system’’ to detect threats found inside based on the access logs of systems.

Xuet al. [30] put forward an anomaly management framework for firewalls by implementing a rule-based segmentation technique. This has identified policy anomalies and an effective anomaly resolution was derived. Sheng et al. [56] has presented a behavior modeling approach .This approach separates service features of web application into operational and control behaviors. This is done for the automated verification of web services. Liu et al. [35] presented a spam-detection framework. This framework helped in detecting various kinds of Web spam, including novel ones. The spam detection approach works with the help of the user-behavior analysis. Fraiwan et al. [22] recommended a classification-based detection approach for the identification of web application attacks. This was based on the java script execution. Ultimately a conclusion was made by Rietta [45] that application level IDS are effective for detecting novel attacks but they are prone to high rates of false positives.

B. Signature Based Intrusion Detection Systems (SIDS)

Signature based intrusion detection systems use the signatures of already recognized and known attacks or vulnerabilities. Then it will apply raw pattern matching algorithms [9] to detect security threats and web application attacks. This type of detection mechanisms are functional at the both the network level and application level. All kinds of network traffic are analyzed at the network level. [62] At application level, its function is to monitor and control server logs [51].The drawback of signature based intrusion detection system is that Signatures of recognized or known attacks are mostly ineffective in preventing zero day attacks. The Security systems can be easily beaten or defeated by the polymorphic behavior of the attacks. Because of the rapid increase in the range and diversity of attacks, signature based IDS also face another challenge. It is the exponential increase in signature rules. To keep identity against each threat in the identity database up to date is a very time consuming task. Snort, for example, has more than 2500 signature rules in their database [47]. Network intrusion detection systems also face problems to work with encrypted communication on the web [74].

Several context based application intrusion detection systems [3] have the trouble of manually creating and updating the signatures in the database. Xu et al. [72] has proposed an approach for the automated generation of security tests. This approach is based on using formal threat models to sense Q3 invalid inputs. The issues related to identification of theft and financial fraud vulnerabilities in banks, healthcare sectors and in government entities are presented by Berghel et al. [11] Duan et al. [17]highlighted an effective spam zombie detection system which monitor the outgoing messages from compromising machines in a network by detecting them.

Vimercatiet al. [64] developed a flexible solution for protecting the information accessible through Web applications by the help of an approach of credential-based access control and trust management. Shar and Tan [55] provided a valid solution against Cross-Site Scripting vulnerability in web applications. a user authentication protocol that influence the short message service to thwart password stealing and reuse attacks was designed by Sun et al. [60] . Antunes and Vieira [4] put forward a defence-in-depth approach to secure Web applications. Auxilia and Tamilselvan [6] tried a n Anomaly detection using a negative security model in a web application. But it was found that it provide only a partial solution and gives little protection against zero day attacks.

C. Data Mining Techniques Or Statistical Intrusion Detection Systems (DMIDS)

Data mining techniques provide frameworks for the detection of web application attacks by the basis of statistical methods [67]. But they often fail however in analyzing malicious payloads by use of the application’s context. Various network based IDS [19, 46] used data mining techniques to find web application attacks. These methods judge character frequency and their occurrence probabilities in malicious data. These types of approaches lack the semantics that are required to understand the contextual character of an attack and its consequences. Balzarotti et al. [7] developed a combination of static and dynamic analysis techniques for the identification of weaknesses in the refinement procedures of web applications that allow an attacker to bypass security controls. This investigation approach illustrates very limited effectiveness. That means it put custom sanitization procedures in place and classically produces many false positive results. Shahzad et al. [54] has contributed in building of a genetic footprint by drawing out the information in the kernel control blocks of a process to enable the detection of malicious processes at run time .Santos et al. [52] has put forward a method that is based on the frequency of opcode sequences in contrast to signature-based techniques of detecting malware. This can be used for the discovery of malware which other data-mining based approaches cannot detect.

Amin et al. [2] set up a new email-filtering technique which is based on email's persistent-threat and recipient-oriented features that outperforms traditional detection methods. Feng et al. [20], introduced a security risk analysis model for the identification of the causal relationships among risk factors .The analysis of the complexity and uncertainty of vulnerability propagation were also done. Stein et al. [59] developed the fuzzy versions of the Bayesian formulas to contract with the conflicting and inconsistent information between the data and preceding knowledge. Yao et al. [73] has explained an anomaly detection methodology that uses a proximity graph and a Page Rank algorithm. Static analysis has demonstrated that it can detect taint-style vulnerabilities in web applications [32]. It provides a high analysis speed while generating a low number of false positives .however the scope of this approach is limited in terms of web application attacks and the methodology used.

D. Ontology Based Intrusion Detection Systems (OIDS)

Ontology-based IDS solutions are becoming exponentially used in information security of web application. Raskin et al. [42] introduced ontology for the data integrity of web resources and promoted the use of ontologies for information security. They state that by using ontology, intrusive performance can be systematically described with any level of detail that is necessary. Ontologies help in reducing the large variety of concepts to a smaller list of properties. The exact specification of security know-how can also be helpful in improving prevention and reaction capabilities. Landwehr et al. [34] described a classified taxonomy of intrusion according to location, means and genesis. A hierarchical model for the specifications attacks through the assessment of attack characteristics and attributes is considered by Ning et al. [37].

McHugh [39] concentrated on the classification of attacks according to protocol layeres. Guha and Mukherjee [31] focused on the analysis of each layer of the TCP/IP protocol stack as the groundwork for attack taxonomy. Ontology for detecting spam messages is used by Santos et al. [53] by capturing the context of an e-mail message and its internal semantics. The major problem in the systems mentioned earlier is that the ontology is just used to represent a simple representation of the attributes of the attack. The drawback of these systems is that they lack the reasoning ability which is very necessary to intelligently protect a system. This is because of their taxonomical structure and concentration on the network layer rather than application layer. Subsequently some of the most critical web application attacks are neglected by them. Denker et al. [14] tried to base the control access on an ontology that was developed in DAML and OIL [29]. Ontologies have not been effectively utilized because they are often limited in representing attack attributes only.

A concept of a target-centric ontology for intrusion detection and protection from network level attacks was proposed by Santos et al. [53]. Eastlake et al. [16] also introduced an ontological model for intrusion detection of network layer attacks. From a taxonomy point of view, intrusion detection possesses characters, classifications and languages that intelligently illustrate instances of taxonomy and convey information regarding an attack or intrusion in web application. The actual power and utility of the ontology is calculated by the facts that it can express, the relationships between collected data and the use of such relationships to deduce particular data which represents an attack of a particular type [63].Hung and Liu [31] introduced a new approach for designing and developing an intrusion detection application by using an ontology. The system conveys the intrusion detection in terms of the end users domain and allows a non-expert person to model the intrusion detection system easily by using the terminologies and concepts of intrusion detection.

An Ontology of Information Security is developed by Herzog et al. [28], which describes assets, threats, vulnerabilities, countermeasures and their relations. Querying and acquisition of new knowledge through inference and rule-based reasoning using OWL Reasoner is also supported. But It does not address web application threats such as SQL Injection and XSS attacks. Fenz et al. [21] concentrated on the utilization of a security ontology that can support the ISO/IEC 27001 certification and maintaining the security guide lines or policies. This work does not though actually deal with any web application vulnerabilities. W3C Semantic Sensor Network ontology using OWL 2 is presented by Compton et al. [13] to describe sensors and observations. Rowe et al. [50] introduced a behaviour ontology that captured the user behaviour within a given context and inferred the role of a user by using semantic-rule based methodology. Park and Kang [48] delivered an automatic rule acquisition procedure using rule ontology and explained that ontology-based rule acquisition approach works in a real-world application. a knowledge-based numerical mapping for nominal attributes that captures and quantifies their underlying semantics is explained by . Domingo-Ferrer et al. [16] .Arogundade et al. [5] developed ontology for the formal representation of eliciting safety and security requirements. The ontological approach helped to avoid any ambiguity and inconsistency in capturing safety and security requirements. Some better approaches by using an ontology to capture the domain knowledge of an application is given by Eastlake and Undercoffer et al. [17,63] but they carry some overhead due to lack of a viable search space reduction mechanism. Such solutions are usually provided in a general form for network level attacks and ignore web application attacks.

IV. CURRENT APPROACH: SEMANTIC BASED SYSTEMS

State of the art technologies have proven ineffective and are unable to provide robust security mechanisms at the application level. But semantic based systems are designed and developed to intelligently understand the application's context, behaviour, the data and the nature of attacks which are possible. Such systems validate the input to it at both the syntactic and semantic levels. Syntax based validation normally applies size or content restrictions. on the other hand, Semantic based validation mainly focuses or concentrates on specific data types, formats and an understanding of potentially malicious commands according to their context and likely consequences. In current years such semantic approaches have proved promise in terms of providing rich representations of web application domain knowledge [23].Viswanathan and Krishnamurthi [63] introduced a personalization approach for making semantic relationship paths by capturing the user's interest level in various domains through their web browsing history. Rubiolo et al. [50] proposed a technique which was based on an Artificial Neural Network model. It is developed for knowledge discovery through ontology matching on the Semantic Web.

The concept of semantic observations and remarks was applied recently to develop an identity management system which was deployed as a firewall for protecting digital assets [1]. Semantic annotations and ontologies can also help us to capture the exact specification of a security model in order to improve its prevention and reaction capabilities [27, 32, 63, 34, 42].intrusive behaviors can be systematically modelled to the required level of granularity .on the other hand the actual power and utility of this is determined by the articulacy of the ontology in modelling attack scenarios in a generalized way and at a certain level of abstraction. The ontology models developed in the preceding art mainly focused on lightweight representations of the attributes of the attacks in a taxonomic structure. The drawback of these systems is that these models seriously lack the necessary ontological modelling and subsequent reasoning capabilities.

Additionally such systems focus on applications within the network layer or access control mechanisms for digital assets. The system is capable of detecting complicated attacks efficiently and effectively. It is practically done by analyzing the specified portion user requests where attacks can be possible. The semantic rules used in the semantic systems allow us to capture the context of the application, the attack and the protocol used. These rules are also utilized for reasoning in ontological models in order to detect complex polymorphic variations of web application attacks. Semantic approach is capable of representing shared understanding information in structured form about the concepts within a definite domain. [44]

V. CONCLUSION

This paper provided an ample survey of recent research details in the area of web application security and various state of the art solutions for security. We analyzed important security properties that secure web applications should preserve. The major vulnerabilities that are existing in the network and the characteristics and weakness of various art solutions .Cyber security issues have increased exponentially in latest years .The survival of e-businesses and the privacy of individual's data are becoming increasingly a topic of research and discussion. State of the art technologies have verified ineffective and are unable to provide robust security mechanisms at the application level. The semantics based detection system that has been developed seems to be effective in making intelligent decisions by keeping the context of a domain application in view. Keeping in view the emerging web technologies and extensive usage of highly interactive content over internet, also new types of attacks are always emerging, it is imperative to develop efficient security mechanisms for the web application security. [3].

REFERENCES

[1] Waleed Alrodhan, Identity management systems, Digital Identity and Access Management: Technologies and Frameworks (2011) 209.

[2]] Rohan Amin, Julie Ryan, Johan van Dorp, Detecting targeted malicious email, Security & Privacy, IEEE 10 (3) (2012) 64–71.

[3] A. Anitha, V. Vaidehi, Context based application level intrusion detection system, in: International conference on Networking and Services, 2006,ICNS'06, IEEE, 2006

[4] Nuno Antunes, Marco Vieira, Defending against web application vulnerabilities, Computer (2012) 66–72.

[5] O.T. Arogundade, A.T. Akinwale, Z. Jin, X.G. Yang, Towards an ontological approach to information system security and safety requirement modeling

[6] M. Auxilia, D. Tamilselvan, Anomaly detection using negative security model in web application, in: 2010 International Conference on Computer

[7] D. Balzarotti, M. Cova, V. Felmetsger, N. Jovanovic, E. Kirda, C. Kruegel, G. Vigna, Saner: composing static and dynamic analysis to validate sanitization

[8] ,J. Bau and J. C. Mitchell, "Security modeling and analysis," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 18–25, 2011

[9] R.S. Boyer, J.S. Moore, A fast string searching algorithm, Communications of the ACM 20 (10) (1977) 762–772.

[10] A. Barth, J. Caballero, and D. Song, "Secure content sniffing for web browsers, or how to stop papers from reviewing themselves," in *Oakland'09: Proceedings of the 30th IEEE Symposium on Security and Privacy*, 2009, pp. 360–371.

[11] Jeremy J Carroll, Ian Dickinson, Chris Dollin, Dave Reynolds, Andy Seaborne, Kevin Wilkinson, Jena: implementing the semantic web recommendations,

[12] You Chen, Steve Nyemba, Bradley Malin, Detecting anomalous insiders in collaborative information systems, IEEE Transactions on Dependable and Secure Computing

[13] Michael Compton, Payam Barnaghi, Luis Bermudez, Raul Garcia-Castro, Oscar Corcho, Simon Cox, John Graybeal, Manfred Hauswirth, Cory Henson, Arthur Herzog, et al., The SSN ontology of the w3c semantic sensor network incubator group, Web Semantics: Science, Services and Agents on the World Wide Web, 2012.

[14] G. Denker, L. Kagal, T. Finin, M. Paolucci, K. Sycara, Security for daml web services: Annotation and matchmaking, The Semantic Web-ISWC 2003(2003) 335–350.

[15] E. Fong, R. Gaucher, V. Okun, P.E. Black, Building a test suite for web application scanners, in: Proceedings of the 41st Annual

Hawaii International Conference on System Sciences, IEEE, 2008. pp. 478–478.

[16] Josep Domingo-Ferrer, David Sánchez, Guillem Rufian-Torrell, Anonymization of nominal data based on semantic marginality, Information Sciences (2013).

[17] Zhenhai Duan, Peng Chen, Fernando Sanchez, Yingfei Dong, Mary Stephenson, JamesMichael Barker, Detecting spam zombies by monitoring outgoing messages, IEEE Transactions on Dependable and Secure Computing 9 (2) (2012) 198–210.

[18]] D. Eastlake, J. Reagle, D. Solo, M. Bartel, J. Boyer, B. Fox, B. LaMacchia, E. Simon, Xml-Signature Syntax and Processing, 2002.

[19] M. Ektefa, S. Memar, F. Sidi, L.S. Affendey, Intrusion detection using data mining techniques, in: 2010 International Conference on Information Retrieval & Knowledge Management, (CAMP), IEEE, 2010, pp. 200–203.

[20] Nan Feng, Harry Jiannan Wang, Minqiang Li, A security risk analysis model for information systems: causal relationships of risk factors and vulnerability propagation analysis, Information Sciences (2013).

[21]] S. Fenz, G. Goluch, A. Ekelhart, B. Riedl, E. Weippl, Information security fortification by ontological mapping of the iso/iec 27001 standard, in: 13th Pacific Rim International Symposium on Dependable Computing, 2007, PRDC 2007, IEEE, 2007, pp. 381–388.Symantec Internet Security Threat Report Trends for 2010, vol. 16(20), 2011.

[22] Mohammad Fraiwan, Rami Al-Salman, Natheer Khasawneh, Stefan Conrad, Analysis and identification of malicious javascript code, Information Security Journal: A Global Perspective 21 (1) (2012) 1–11.

[23] Tom Gruber, What is an ontology, Encyclopedia of Database Systems 1 (2008).

[24] B. Guha, B. Mukherjee, Network security via reverse engineering of tcp code: vulnerability analysis and proposed solutions, Network, IEEE 11 (4) (1997) 40–48.

[25] G. Wassermann, Z. Su "Sound and precise analysis of web applications for injection vulnerabilities", Proceedings of the conference on Programming language design and implementation (ACM SIGPLAN),Vol. 42 (6), 2007.

[26] http://www.ipa.go.jp/index-e.html

[27] Justin Clarke, SQL injection attacks and defense, 2009

[28] A. Herzog, N. Shahmehri, C. Duma, An ontology of information security, Techniques and Applications for Advanced Information Privacy and Security: Emerging Organizational, Ethical, and Human Issues (2009) 278–301. and ruleml, W3C Member Submission 21 (2004) 79.

[29] I. Horrocks et al, Daml + oil: adescription logic for the semantic web, IEEE Data Engineering Bulletin 25 (1) (2002) 4–9.

[30] Hongxin Hu, Gail-Joon Ahn, Ketan. Kulkarni, Detecting and resolving firewall policy anomalies, IEEE Transactions on Dependable and Secure Computing 9 (3) (2012) 318–331

[31] S. Hung, S. Liu. A User-Centric Intrusion Detection System by Using Ontology Approach, 2006.

[32] . Jovanovic, C. Kruegel, E. Kirda, Static analysis for detecting taint-style vulnerabilities in web applications, Journal of Computer Security 18 (5) (2010) 861–907

[33] C. Krugel, T. Toth, E. Kirda, Service specific anomaly detection for network intrusion detection, in: Proceedings of the 2002 ACM symposium on Applied computing, ACM, 2002, pp. 201–208

[34] C.E. Landwehr, A.R. Bull, J.P. McDermott, W.S. Choi, A taxonomy of computer program security flaws, ACM Computing Surveys (CSUR) 26 (3) (1994) 211–254.

[35] Yiqun Liu, Fei Chen, Weize Kong, Huijia Yu, Min Zhang, Shaoping Ma, Liyun Ru, Identifying web spam with the wisdom of the crowds, ACM Transactions on the Web (TWEB) 6 (1) (2012)

[36] G.W. Manes, D. Schulte, S. Guenther, S. Shenoi, Netglean: a methodology for distributed network security scanning, Journal of Network and Systems Management 13 (3) (2005) 329–344

[37] J. McHugh Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by Lincoln laboratory, ACM Transactions on Information and System Security 3 (4) (2000) 262–294.

[38] My Space "http://namb.la/popular/tech.html," 2005. .

[39] P. Ning, S. Jajodia, X.S. Wang, Abstraction-based intrusion detection in distributed environments, ACM Transactions on Information and System Security (TISSEC) 4 (4) (2001) 407–452.

[40] Sangun Park, Juyoung Kang, Using rule ontology in repeated rule acquisition from similar web sites, IEEE Transactions on Knowledge and Data Engineering 24 (6) (2012) 1106–1119.

[41] Cristian I Pinzon, Juan F De Paz, Alvaro Herrero, Emilio Corchado, Javier Bajo, Juan M Corchado, idmas-sql: Intrusion detection based on mas to detect and block sql injection through data mining, Information Sciences (2011).

[42] [42] V. Raskin, C.F. Hempelmann, K.E. Triezenberg, S. Nirenburg, Ontology in information security: a useful theoretical foundation and methodological tool, in: Proceedings of the 2001 Workshop on New Security Paradigms, ACM, 2001, pp. 53–59

[43] A. Razzaq, A. Hur, M. Masood, K. Latif, H.F. Ahmad, H. Takahashi, Foundation of semantic rule engine to protect web application attacks, in: 2011, 10th International Symposium on Autonomous Decentralized Systems (ISADS), IEEE, 2011, pp. 95–102.

[44] Abdul Razzaq , Khalid Latif, H. Farooq Ahmad, Ali Hur, Zahid Anwar, Peter Charles Bloodsworth, Semantic security against web application attacks, Information Sciences (2013)

[45] F.S. Rietta, Application layer intrusion detection for sql injection, in: Proceedings of the 44th Annual Southeast Regional Conference, ACM, 2006, pp. 531–536.

[46] Ivan Ristic, Modsecurity: Open Source Web Application firewall, 2010.

[47] M. Roesch, et al., Snort-lightweight intrusion detection for networks, in: Proceedings of the 13th USENIX conference on System administration, Seattle,Washington, 1999, pp. 229–238.

[48] Matthew Rowe, Miriam Fernandez, Sofia Angeletou, Harith Alani, Community analysis through semantic rules and role composition derivation, Web Semantics: Science, Services and Agents on the World Wide Web, 2012.

[49] (U.S.) O'Neill. Web Services Security technology and theory. Ran Xiao Man, Guo Wenwei translation. Beijing: Tsinghua University Press, 2003

[50] M. Rubiolo, M.L. Caliusco, G. Stegmayer, M. Coronel, M. Gareli Fabrizi, Knowledge discovery through ontology matching: an approach based on an artificial neural network model, Information Sciences 194 (2012) 107–119.

[51] T. Ryutov, C. Neuman, K. Dongho, Z. Li, Integrated access control and intrusion detection for web servers, IEEE Transactions on Parallel and Distributed Systems 14 (9) (2003) 841–850.

[52] Igor Santos, Felix Brezo, Xabier Ugarte-Pedrero, Pablo G Bringas, Opcode sequences as representation of executables for data-mining-based unknown malware detection, Information Sciences (2011).

[53] Igor Santos, Carlos Laorden, Borja Sanz, Pablo G Bringas, Enhanced topic-based vector space model for semantics-aware spam filtering, Expert Systems

[54] Farrukh Shahzad, M. Shahzad, Muddassar Farooq, In-execution dynamic malware analysis and detection by mining information

in process control blocks of linux os, Information Sciences (2011).

[55] Lwin Khin Shar, Hee Beng Kuan Tan, Defending against cross-site scripting attacks, Computer 45 (3) (2012) 55–62.

[56] Quan Z Sheng, Zakaria Maamarb, Lina Yaoa, Claudia Szaboa, Scott Bournea, Behavior modeling and automated verification of web services, Information Sciences (2012).

[57] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, Yarden Katz, Pellet: A practical owl-dl reasoner, Web Semantics: Science, Services and Agents on the World Wide Web 5 (2) (2007) 51–53.

[58] Jungsuk Song, Hiroki Takakura, Yasuo Okabe, Koji Nakao, Toward a more practical unsupervised anomaly detection system, Information Sciences (2011).

[59] M. Stein, M. Beer, V. Kreinovich, Bayesian approach for inconsistent information, Information Sciences (2013).

[60] [60] Hung-Min Sun, Yao-Hsin Chen, Yue-Hsun Lin, opass: A user authentication protocol resistant to password stealing and password reuse attacks, IEEE Transactions on Information Forensics and Security 7 (2) (2012) 651–663.

[61] Top10 OWASP, Top 10–2010, The Ten Most Critical Web Application Security Risks, The Open Web Application Security Project, 2010.

[62] C.Y. Tan, S.R. Tan, B. Morel, 19601 Information Warfare, in: 19601 Information Warfare, Carnegie Mellon, 2011, pp. 15.

[63] J. Undercoffer, J. Pinkston, A. Joshi, T. Finin, A target-centric ontology for intrusion detection, in: 18th International Joint Conference on Artificial Intelligence, 2004, pp. 9–15.

[64] Sabrina De Capitani Di Vimercati, Sara Foresti, Sushil Jajodia, Stefano Paraboschi, Giuseppe Psaila, Pierangela Samarati, Integrating trust management and access control in data-intensive web applications, ACM Transactions on the Web (TWEB) 6 (2) (2012)

[65] Xiaowei Li and Yuan Xue, A Survey on Web Application Security, ACM Transactions on the Web (TWEB) 6 (2) (2012)

[66] V. Viswanathan, Ilango Krishnamurthi, Ranking semantic relationships between two entities using personalization in context specification, Information Sciences (2012).

[67] X.F. Wang, J.L. Zhou, S.S. Yu, L.Z. Cai, Data mining methods for anomaly detection of http request exploitations, Fuzzy Systems and Knowledge Discovery (2005) 484. 484.

[68] Wang Ziyao.The core technology and application of SOA.Beijing: House of Electronics Industry,2008-5

[69] H. J. Wang, C. Grier, A. Moshchuk, S. T. King, P. Choudhury, and H. Venter, “The multi-principal os construction of the gazelle web browser,” in *USENIX’09: Proceedings of the 18th conference on USENIX security symposium*, 2009, pp. 417–432.

[70] WhiteHat Security, “WhiteHat website security statistic report 2010.”

[71] Web Application Security Statistics, “http://projects.webappsec.org/w/page/13246989/WebApplication Security Statistics

[72] Dianxiang Xu, Manghui Tu, Michael Sanford, Lijo Thomas, Daniel Woodraska, Weifeng Xu, Automated security test generation with formal threat models, IEEE Transactions on Dependable and Secure Computing 9 (4) (2012) 526–540.

[73] Zhe Yao, Philip Mark, Michael Rabbat, Anomaly detection using proximity graph and pagerank algorithm, IEEE Transactions on Information Forensics and Security 7 (4) (2012) 1288–1300.

[74] Xin Zhao, Atul Prakash, Wsf: An http-Level firewall for Hardening Web Servers, 2005.

ISBN No.978-1-4799-3834-6/14/$31.00©2014 IEEE