Advance Website Application Security Issues

GHULAM HUSSAIN JALBANI*, AKHTAR HUSSAIN JALBANI*, ROZITA JAMILI OSKOUEI **, FOZIA NOUREEN*, AND ZOJAN MEMON*****

Department of Information Technology, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Pakistan.

** Department of Computer Science & Information Technology, Mahdishahr Branch, Islamic Azad University, Mahdishahr, Iran.

*** Department of Computer Systems Engineering, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Pakistan.

**** Department of Information Technology, University of Sufism & Modern Sciences, Bhitshah, Pakistan.

Authors E-Mail: (ghjalbani@gmail.com, jalbaniakhtar@gmail.com, rozita2010 r@gmail.com, sagacious.786@gmail.com, zojan31memon@gmail.com)

ABSTRACT

As the usage of web applications are increasing day-by-day. Because of easy to develop web applications within weeks and are easily accessible from any part of the world for its users. Every type of business, information sharing, or social networks are on the web now. Such as eCommerce, online banking, social network websites, blogs, online taxi booking, and online education, etc. As the growing number of users or businesses on the web, the attackers are target to attack them. There are too many types of attacks on the web applications but in this paper, few high-risk attacks are considered. Like DDoS (Distributed Denial of Service) attack on the web applications, SQL (Structured Query Language) injection, XSS (Cross-Site Scripting), Cache poisoning, DNS (Domain Name Server) poisoning, HTTP (Hyper Text Transfer Protocol) response splitting, and command injection. How these attacks are performed on the web application defined in details. Due to these attacks user's privacy and sensitive data leakage. Web applications are also facing the issue of defacing at the organization and state level. Moreover, different tools of vulnerability scanning and protection for attacks will be described such as SQL Inject-Me, Xenotix XSS, SSLyze, XSSer. The controls to protect the web applications from these attacks and will give recommendations for web developers and system administrators. The web applications should be properly sensitized for input fields, developed as per current security standards. Current versions of web servers used modules, frameworks, and tools.

Key Words: **Web Applications, Web Application Security, Structured Query Language Injection,Cross-Site Scripting, Distributed Denial of Service Attacks, Hyper Text Transfer Protocol Response Splitting, Cache Poisoning.**

1. INTRODUCTION

In the current era, every business or any community wants its existence on the web application. There are too many types of services that are running on web applications. Such as eCommerce, online banking, food delivery, health management systems, social networking websites, online education, etc. Because the development of web applications is taking less time and cost. These web applications are easily accessible from any device on which browsers are running. So that the attackers are targeting web applications to compromise them and get more gain. Because as the number of users and business is growing day by day on the web applications. The attackers are compromising web applications with various types. For example, DoS/DDoS attacks, defacing the website, insertion of malicious code, redirecting websites, insertion of malware, social engineering, brute force attacks, and phishing attack.

The most dangerous and at the top of all attacks is an SQL Injection attack on web applications. SQL Injection is an injection type of attack in which attackers give command of SQL queries into a user input box of a web application to get administrator and unlimited rights. This SQL query of the attacker will be converted into the SQL code [1-2]. SQL Injection vulnerability is the entry point for attackers to exploit web applications. Due to the poor user input validation, this vulnerability exists [1]. For the real-time data processing applications, NoSQL databases are getting more attention to be used and with its usage, the data manipulation can be performed directly on data. Due to this functionality, interactive systems are developed more easily. The non-structural design is a big advantage to change attributes and due to this modification method is too much easy. However, its advantage is also a weak point for the security of databases because more attacks are reported as injection attacks on web applications. It does not mean that the SQL queries are not used in NoSQL so that there is no threat of injection attacks [3]. Another more common attack on websites is XSS by this attack the attacker runs malicious code into the targeted user's browser. By using various client-side scripting languages. A threat to web application security if the user's input is not

validated properly. The web applications are not validated with user input are vulnerable to various types of malicious attacks, which are related to the command or code injection attacks. Another attack on a web application is XPath injection also related to improperly coded applications. The web services code is compromised with the XPath injection attack. PHP (Hypertext Preprocessor) Object injection, function injection, and web parameter tampering [4]. To get control of the targeted user's browser in such a way to execute malicious code by an attacker to compromise the weakness of broken authentication and session management in web applications. In this attack, the attacker can get access, administrator user or cookies, and session IDs (Identifications) of the web browser for target web applications [5]. The primary method for user authentication in a web application is the username and password. This process is implemented with the help of session management in the applications these sessions are stored at web browser cache. By using this in web application these are looking user-friendly for the clients or web application users. They assume that only authorized users can be able to access them. The restriction is implemented on web resources like web pages and databases etc. with the help of access control. This is known as security configuration for blocking unauthorized access by the attackers [6]. To detect the DDoS attack in web application the feature selection method has been used. In this method, the ANN (Artificial Neural Network) and MLP (Multilayer Perceptron's) [7] are used for feature selection to analyze the event data. This machine learning technique is applied for automated systems to detect the DDoS attack on the web application. The Dark Web [8] is World Wide Web content yet it is not the section of the surface web because of which it is likewise not accessed through the normal browsers which are typically used to get to the surface web. The dark web is that part of the web where the majority of the illicit and sharing that stuff occurs. The Dark Web is likewise utilized as an unlawful proposes for Terrorism, Hacking and Fraud Services, Phishing and Scams, Child Pornography, and considerably more [9]. Dark Web is a part of the Deep Web. Dark Web offers hidden types of functionalities, which have the onion expansion. For instance, Facebook works with hidden services. Another model is the Duck Go internet searcher. There is an extraordinary sort of program to get to the Dark Web. The different programs which are utilized to get to the Dark Web are TOR (The Onion Router), FreeNet, Riffle, I2P (Invisible Internet Project), and Whonix. As the ease of web application increased with this feature at the same time risk loss of user information increased. If the web applications are developed with poor code and design may have the vulnerability of broken access control. The websites not secured may lead to well-known vulnerabilities like injection, cross-site scripting, cookie stealing, and session hijacking. As per the recommended logic level for application refers to only authorized users should be able to access information available at websites. The incomplete security configuration of website applications may open a loophole for this vulnerability. Misconfiguration can be on the application server, web server, database systems, and other website-related platforms. So that this vulnerability can occur at any level of the website. Due to this threat attackers can get unauthorized access to the application data or any operation. That may be in the form of the entire system compromised. These all mentioned attacks are a big threat to the security of websites developed without input validation and improperly coded applications.

Sections: The remaining paper is divided into the following sections; Section II gives a detailed view of attacks on websites and how those attacks are performed with an insight view of each type of these attacks. The different methods of each type of attack are also defined in depth. Section III will describe the attack detection tools for injection, session hijacking, or any other vulnerabilities in websites. Section IV will give an overview regarding different methods are used to detect attacks or attacker's methods to carry out the attacks. By using digital forensics, honey pots, and data mining. Section V discusses the firewalls that are in use at different OSI (Open Systems Interconnection) layers for attack detection or prevention for websites. In section VI the recommendations have been given for the secure web applications. Section VII concludes the website's attacks, detection tools, methods, firewalls, and proposed suggestions for the secure website and its development along with future directions.

2. METHODOLOGY

The processes are considered as a top-down method to get a more meaningful understanding of the current status regarding the website security. At first, step, have taken a look into literature those have mentioned the website security-related issue not published before than 2000. That already work done has been divided into attacks, detection tools, methods, firewalls, and testing methods with their proposed solutions for the prevention of these attacks. This paper considered OWASP (Open Web Application Security Project) the top 10 attacks mentioned in the 2016 or 2017 report or defined in research papers again and again. And have been focused on those most dangerous attacks on the websites and prevention techniques. The proposed methodology for the security of web application is given in Fig 1. In this technique the different type of firewalls should be deployed such as Web Application Firewall, Secure VPN (Virtual Private Network), Proxies and for log analysis machine learning methods. There should be proper sanitization of input fields, secure web application development standards should be followed, and awareness should be given to users regarding phishing attacks.

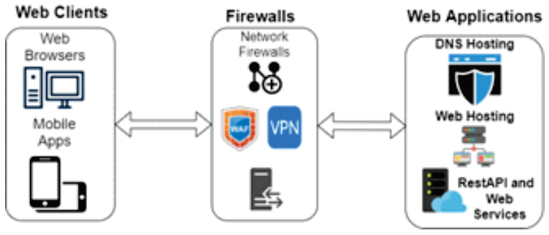

FIG. 1. PROPOSED WEB APPLICATIONS SECURITY INFRASTRUCTURE

After deeply reviewing research papers, the research has some suggestions for developers and server administrators regarding website security. These security-related issues are divided into attack types, attack detection tools, methods, and testing methods for finding any vulnerability in websites. As per the study, almost every research paper related to website security mentioned the SQL injection or any injection type attack. But they have not defined other more threading attacks in details such as broken authentication, broken access control, using components with a known vulnerability, and security misconfigurations. So that, to cover them these are divided this paper into attack types, tools used to detect those vulnerabilities, the process to find threats, and usual approaches for test method of website security. It has been described all of them in more detail on how to detect these attacks and for prevention methods from these mentioned attacks with fewer efforts.

This paper in more detail for the website security related. These suggestions have compared this work with the current research work, such as [10] in this survey paper author has just described the SQL injection attack in detail and gives the only overview of cross-site scripting, security misconfiguration. Same as another paper [11] author in the survey just followed OWASP's top 10 attacks and give an overview of each attack but that those are not discussed in more detail. In this research work has been tried to fill the gap regarding attacks description on website defined in more details and mentioned good practices for the secure web application development and configuration with a survey during 2014-2019.

3. ATTACK TYPES

There are too many types of attacks performed on web applications at the client-side or server-side. The more threatening attack on websites is injection after that cookie stealing, session hijacking, and security misconfiguration. These attacks are performed with different intent such as website defacing at the government level or personal level, stealing user information, DDoS attacks, financial gains, and intellectual property theft. These above-mentioned attacks are executed by attackers by looking into website vulnerabilities, poor validation, or no-limit applied for login attempts. There is still a big issue of plain text communication between client and server for web applications exists. Due to this session hijacking attack can be performed on those websites. With these, all attack the security tried CIA (Confidentiality, Integrity, and Availability) of a web application is compromised.

SQL Injection: SQL Injection attack is more threatening to the website security from the last decade and it is still a more persistent attack. Websites are more vulnerable to this attack and it is easy to lunch on web applications by attackers. Due to this more devastating impact on the privacy of users, personal information leakage, and attacker can control complete application or hosted server. The database server is manipulated by an attacker to run malicious code in a string format. This attacker can fetch the data or change any data in the database server. SQL injection attack takes place on the web application by sending invalidated input to the database server. The overview of SQL injection is defined in Fig. 2.

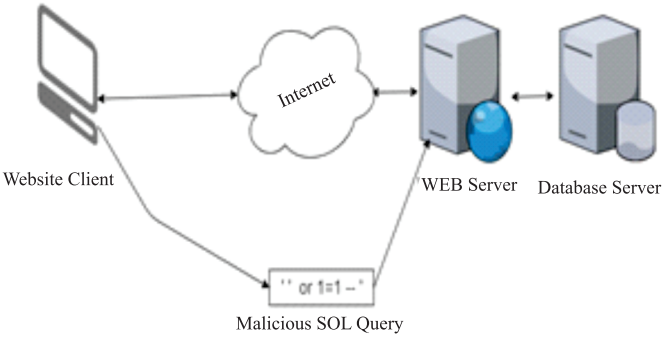

FIG. 2. SQL INJECTION ATTACK OVERVIEW

The driving force behind this attack is to find out injection parameters, avoid authentication, and getting information from the database server [12]. There are too many ways to lunch SQL injection attacks described below.

(i) ***Tautology Method:*** To find out injectable parameters, avoid authentication, and dig out data the tautology method is used [12]. It is a formula that always returns a true value. Such as web application has the functionality of login users with their username and password for authentication, these are stored into a database with the help of SQL query.

SELECT * FROM users WHERE username = 'admin'
AND secret = 'admin123'

The extraction of data by using tautology insertion in WHERE statement of a SQL query and by using the method of SQL injection the authentication for users can be avoided. The "' ' OR 1 = 1 - -" query statement is used by an attacker to perform a tautology based method for SQL injection attack. The above-mentioned query will be used on the web application login page and to extract the password also. The query will look like as given below.

SELECT * FROM users WHERE username = ' '
OR 1 = 1 - - ' AND secret = 'xyz123'

The (') sign is used to tell the server the end of a string and if it is used before the OR operator. The symbol (--) is used for comment on the remaining part of a

query, by this process query will be exit successfully. With this attacker can bypass authentication to login without password and query will return true.

(ii) ***Union Query Method:*** To avoid authentication and dig out data from the databases via a union query injection attack. The injection query with the help of the UNION operator will be injected by an attacker into a vulnerable web application which will return the dataset of the original format and injected queries also. Such as, we are assuming a web application is connected with a database that contains few important tables one is Users table for authentication of users and second is the Employees table which contains the information regarding all employees like as, name, address, cell number, salary package, designation. If in that web application login page have the vulnerability of SQL injection, then the attacker can inject malicious query in the login input box " ' UNION SELECT * FROM Employees -- " and the password, the query will look like as in database:

SELECT * FROM Users WHERE username = ' '
' UNION SELECT * FROM Employees – ' '
AND passwd = 'xyz123'

The null set will be returned by the first section of the query because there is not an empty record in Users table against username column which will be compared with this query. And another section of a query will return complete records from the Employees tables. As the function of the UNION operator, the dataset will be returned against both queries, due to this attacker will be able to fetch all the records from the Employees table. For a simple method to combine too many queries the number and data types should be the same for all table columns. For different numbers and data types within table columns query will be changed to perform this type of attack.

(iii) ***Wrong Queries Method:*** This type of method is used for the initial stage of attack to find out vulnerabilities in web applications such as parameter injection, to get information regarding which database is in use, and to dig out the data. The attacker will insert incorrect queries in the input box to get an error message from where he can get information regarding which database in being in use by web application [13]. It is also known as an error-based injection. To get information about data types and to view the names of tables and their column info within the database. The queries with wrong syntax, conversion type, or any other logical error will be created. To get information regarding data types of column the conversion type error is generated. At another end to get information regarding columns and tables to name the logical error will be generated. Such as, the hacker will inject a query "'UNION SELECT SUM(username) FROM Users -- "at login input box on a web application. The query will be look-alike below.

SELECT * FROM Users WHERE username= '
UNION SELECT SUM(username) FROM Users -
- ' AND passwd = 'xyz123'

The above query will try to convert the username column into an integer in the Users table. This wrong query with not a valid conversion type will generate an error message and this will show information regarding the database and username column.

(iv) ***With Persistent Procedures Method:*** To communicate with the operating system and increase the usability of the database almost all databases have a standard of stored procedures. These stored procedures are used for access control management and to validate the data. As the attacker gets information regarding which database is in use and the vulnerable parameters for SQL injection attacks. After that, he will create a special query to run an available stored procedure on the targeted database by this process the interaction with the operating system will be enabled. Such as, "'; SHUTDOWN; - - "query will be run in the field of a username on the login page and with any dummy password.

SELECT * FROM Users WHERE username= '';
SHUTDOWN; - - 'AND passwd = 'xyz123'

With the above SQL query, the database will be shutdown. For getting root right or other required privileges, to run remote commands, or lunch a denial of service attack the stored procedure injection attack will be performed.

(v) ***Piggy-Back Query Method:*** If any vulnerable web application which allows adding an extra query with the legitimate query is known as Piggy-Backed. The extraction or modification of data, run remote commands, or lunch of DoS attack will be also possible if any web application has this type of vulnerability. As mentioned in the stored procedure attack method " '; SHUTDOWN; - - " query will validate the username and password to perform a

denial of service attack. Such as, to change the value of any dataset will be possible with this injection query " '; DROP TABLE User Details - -' in the username input box and with dummy password, this query will look alike written below.

SELECT * FROM users WHERE username = '';
DROP TABLE UserDEtails – ' AND passwd
= 'xyz123'

The first section of the query will return the null string from the User table. The second section is a piggy-backed query that will delete the User details table from the database.

(vi) ***Blind SQL Injection:*** Secure databases that are not generating any errors on those web applications the Blind SQL injection attacks are performed [14]. In this type of attack, we can find injectable parameters and dig out the data from the database. The queries with the condition of true or false will be applied by an attacker to check the behavior of the web application. Due to this, it is known as the Blind SQL injection attack. By this process, an attacker can look for vulnerabilities in the web application. To demonstrate it such as, we are assuming a website link as follows:

http://myshop.com/showItems.php?id=1

The "OR 1 = 1" will be added and the second time "AND 1 = 2" will be appended at the end of the link after that the behavior of that targeted application will be checked. With this attacker can find out SQL injection parameters for the injection attack. As a first value added at the end of the link "http://myshop.com/ showItems.php?id=1 OR 1 = 1" if they see the same page with this and with second value "http://myshop.com/showItems.php?id=1 AND 1 = 2"display error page then this web application will be considered as vulnerable to SQL injection attack.

Non-SQL Injection: A threat of NoSQL injection to real-time web applications which may be in the form of critical data loss, user's information leakage, and in the form of the company's data loss. NoSQL database usage has increased as these types of databases are used in mobile applications. These are used for organization services apps, for personal health apps or for any web application as the main source of storing data. The attackers are trying to apply some type of techniques on NoSQL for injection as on SQL database servers or web applications using the SQL database. As the ease of use and better support for big data, due to this developer is using NoSQL as their first choice for applications [15]. The method of NoSQL injection is different from SQL injection on a web application. Such as SQL injection is performed inside the database, but this is not a case with NoSQL because it can be run on the application or a database. As it all depends on the NoSQL API (Application Program Interface) or the model of data is in use.

The NoSQL attack is performed as the string is parsed, analyzed, or connection established with NoSQL API. The functionality of all database attacks is the same for example, create, get data, modifications of records, and delete any record. Users can get more useful information via database queries. These queries can be crafted as an input box, popup windows, through which malicious query can be injected. That is a big issue because users have permission to input any notations in those input boxes or pop-up windows. These queries may contain malicious code in insert query, as a legitimate query run along with this malicious query will run also. Most developers do not give more attention to closing statements and validation to users' input properly which is a big cause of injection attack. These little issues of inserting malicious information, deleting any important record, and getting root rights, which may lead to compromise a complete system or disturb services of a web application for its legitimate users. The more common methods for NoSQL attacks such as notations and segments will be described in the following section with examples.

(i) ***MongoDB:*** The most commonly used NoSQL database is MongoDB which is a cross-platform document-oriented. It is used in large scale demanding applications or multiple data centers, those are processing unstructured, semi-structured, and polymorphic data [16]. MongoDB is simpler for development and functionality in the form of providing too many options within a single database as a managed platform. As the JSON (JavaScript Object Notation) or BJSON (Binary JSON) store a very loose data structure the same can be done in MongoDB also, but it can store more complex datatypes. It supports the object-oriented query language and relational database query languages that is a big advantage of MongoDB. With this functionality, too many types of single functions can be achieved and indexing of data can be performed. As with other new technologies, NoSQL also has similar security issues as they are launched [14].

(ii) ***NoSQL Injection Attack Methods:*** Attackers are looking into vulnerabilities of existing web applications to inject malicious SQL command in databases [17]. By injecting these malicious queries in

a web application, the security can be exploited. To understand these complex NoSQL injection attacks some examples given.

Firstly, will define the JavaScript injection method. Due to the easy and best use nature of JavaScript, it is used in data services and too many companies also prefer to use it for website development. It supports to NoSQL database and ease of usage for big data applications also. As the JSON and JavaScript are simpler and support structured queries with the help of this NoSQL database improved security by removing the SQL language completely and provides easy development. To compromise the databases or destroy them the injection attack is most commonly used, which may create issues for data service providers and users. With much more usage of JavaScript so that developers are trying to defend against injection and attackers are to perform injection attacks. A simple method of injection is the insertion of malicious code into the input field in a way to close the running statement and implement it. Such as web application that is using MongoDB as a database and that application has the functionality of sign in with usernames. As per normal usage any user is coming on that application and sign in with their respective username, but an attacker can inject malicious information into the database. That malicious query is used for the collection of all user info and save them into the database also, such as:

$inset = "db.getCollection('UserList').insert({' username': '$param});";
$response = $db.-> execute ($inset);

A legitimate user will insert well-known names like "Bob" in the dialog input box. But the attacker can insert abnormal and malicious longer strings into it like as "Bob'}) ;db.inject. insert({'Complete':'2". In this case input will be divided into two sections: Bob'}); and the second section will be db.inject.insert({'Complete':'2. Due to this division, these two sections will be closed in two parts by the semicolon. And also considered as two separate statements for execution in the places of single. As a result, in the first part, the original name will be inserted into collection UserList and in the second part new injection code will be generated and Complete:2 also injected into a collection. This is a very easy method to check the vulnerability of the NoSQL database and malicious statements injection on them. There is just an attacker who needs to know the interpretation of statements and execution within a system so that he can design similar malicious code and finish them.

The second method of injection is a JavaScript file. The developers are allowed to use JavaScript files for particular data processing and maintenance in MongoDB. With this functionality, the developers can create their files and these files can be run easily in MongoDB. Due to this benefit, the user-friendly graphical interface can be developed for the organization and it can be developed as per user demand. This advantage can be used for malicious code which will be executed and it leads to more threatening actions like reading, write, or get the root rights. The simple file load command is given below.

MongoDB > load("myfile.js")

These type of JS files gives a more controlled process for developers and users, without any filtration of inputs attackers can use the same process to get anything from database which they want. To gain access to MongoDB is again assuming the same vulnerable website. To execute a file, it needs a connection with the database, after that, the statement will be inserted into it. As the code in the file is not filtered so that without any proper validation the malicious code can be executed. Such as file may have an insertion of another data in the database.

mydb = conn("127.0.0.1:28010/mysitedb");
mydb.inject.insert({'Complete':'1'});
print("Injection Completed");

In the above JS code the local host server, listening on port number, and database name 127.0.0.1:28010/mysitedb are set and insertion query mydb.inject.insert({'Complete':'1'});also injected.

Cross-Site Scripting: Another attack that belongs to injection is XSS on websites. In this attack, the malicious code is injected into the web application by the attacker which will be executed into a user's browser or client through website pages on the target user's system. This malicious code is written in different client-side scripting languages such as HTML, Java, JavaScript, ActiveX, or Flash by an attacker which may be hosted by him or inject that malicious code on any vulnerable website.

The hosted website security zone will be compromised with a malicious script by an attacker to run his script into a targeted user's browser [18].This will give control of web application to an attacker with the right to read, write, and send any important information which is accessible by the browser. The website user account can be compromised through cookies; the user can be redirected to another website to increase traffic on that. With the help of this attack, another attack can be performed such as phishing or drive-by downloads. An overlook of an XSS attack given in Fig. 3.

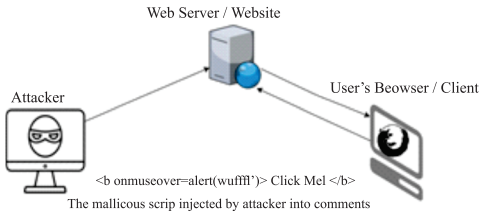

FIG. 3. CROSS-SITE SCRIPTING ATTACK

(1) ***JavaScript Payload Injection Method:*** This is one of the common methods to inject the malicious JavaScript code in the web application of targeted users [19]. In this type of method, the malicious JavaScript injected into input fields. It is only possible in vulnerable applications to accept these types of malicious scripts. As the attacker injects malicious JavaScript into a web application that will run into the targeted client browser. The legitimate user will consider that post as simple text. But the hacker has inserted malicious script in that post: <script>Alert ("You have been exploited")</ script>in input filed of comments or post. Now any legitimate user visits that web page then he will see the alert of "You have been exploited" in that post response. And this malicious code will be run as a client-side script on that web page with this process the XSS attack will be executed at the client browser.

(2) ***Important Information Getable with JavaScript:*** Due to the more serious impact of JavaScript on web applications because of this vulnerability. Different type of user's information can be theft from client's browsers such as:

- ***Stealing of Cookies:*** By performing an XSS attack on vulnerable web applications the attacker can capture session IDs [20].
- ***Phishing:*** With the help of XSS attack the fake login page can be created by the attackers. And this will be used to steal user's personal information by using the DOM (Document Object Model) functionality of the webserver.
- ***Key Logger:*** By applying keyboard log event listeners in a web application the attacker can log keystrokes of targeted users and get their personal information by getting access to a web server such as usernames, personal identification numbers, passwords, and credit card info.

Broken Authentication: In today's web application the most important feature is authentication and sessions, if any application has a vulnerability regarding this above-mentioned functionality then that is known as broken authentication [21]. If the user accounts are not managed very well then there will be an issue of broken authentication. This issue will occur mostly when the sessions are not managed properly as per current web security standards. The functionality of logout, better management of passwords, session timeout, security questions, and update of user information should be considered in web application. The web application developed without the above-mentioned considerations then the attacker can exploit them. With this exploitation, a hacker can steal more sensitive information such as user's login information, bank accounts information, and session IDs.

Broken Access Control: Web applications are being developed with different types of user rights level. It will define which section of web application users can be accessed or not with the help of access control management that is known as authorization [22]. This vulnerability also related to broken authentication or session management within a web application. If an attacker can exploit this vulnerability, then he can get root rights. And it will be a more damaging threat to web applications because the attacker can extract any data, can crash the server or denial of service for the organization's web services. As per the growing number of websites users and mobile apps, it is too much hard for a business to afford the services downtime or leakage of any user information.

Using Components with Known Vulnerabilities: The web applications are developed with the help of APIs, libraries, payment gateways, and other frameworks these all are integrated with complete access rights. Most web developers do not know stable versions of those libraries and on which other components these are depending. If any old version of API or any vulnerable framework is used in web applications which may be more damaging in the form of data loss, user's privacy leakage, and destroy any useful data of an organization.

Security Misconfiguration: The web applications are developed and run on more complex systems such as different types of software (IIS, Apache, and Nginx), database servers (MySQL, MongoDB, PostgreSQL, and SQL, etc.), load balancers, cloud services for web, and much more. To avoid the security misconfiguration vulnerability in web applications there is a need fora correct configuration of software or severs, proper access control management, the libraries, or server software should be up to date [10]. The vulnerability of misconfiguration or illegal access to web applications due to this attacker or adversaries' compromises and it may lead to complete control of the application network.

4. ATTACKS DETECTION TOOLS

SQL Inject-me: For the detection of SQL injection vulnerabilities, an Inject-Me tool can be used and this is a Firefox base. To test the web application security this is a complete set of tools and as an application used. The HTML form is submitted to this tool and that form will be converted into strings then these values are analyzed for the SQL injection vulnerabilities. These form fields are submitted as database escape characters after that it will check for error messages and that appears as output. This tool will not expose

the system to any vulnerability. It will only look for different ways of attacks on web applications or database systems. But it will not be used for password cracking or packet capturing. The SQL Inject-Me have not any functionality of port scanning or attacks on the firewall.

SQL Ninja: In this tool, the Microsoft SQL Server is used as back-end and it is developed for the SQL injection vulnerabilities detection in web applications. To give access remotely on vulnerable database servers is a core function of this tool. This tool can perform better also in more threatening infrastructure because of its powerful functionalities. With the help of this tool, a penetration tester can take full control of database server processes if any SQL injection vulnerability arises [23]. So, a few main functions are given below.

- The blueprint information of the remote SQL server will be fetched, like as version of the server, which type of queries are being used by an application and authentication type of database server.
- If accidentally xp_cmdshell custom is deleted, then it will be created again by this tool.
- Password cracking or brute force will be performed for 'SA' users.
- The port scan of the TCP/UDP (Transmission Control Protocol or User Datagram Protocol) will be performed on the victim's SQL server, to check either these ports are allowed by the victim's firewall and in use or not.
- The DNS-tunnel power-shell and ICMP (Internet Control Message Protocol)-tunnel command shell will be used if the TCP/UDP ports are blocked.

Havij: For the detection of SQL injection vulnerabilities in web applications, the penetration tester is using this automated tool. This is used for information gathering regarding which database server is in use by a web application. Due to its extraordinary process and power for the SQL injection vulnerabilities detection. The more than 95% successful rate of an attack on targeted vulnerable websites [24-25]. A few of them are given below.

- Extract data from the database, get backup of tables, and run SQL queries on the server.
- It can get back hashes of the database management system's users and passwords.

XSS Server: This tool is used to exploit XSS vulnerabilities of the server-side. The main function of the XSS server is to collect important information as the XSS malicious code runs on the user's system or as any infected website with XSS malicious code is being accessed by them. The different types of data can be captured such as client's cookies, IP addresses, details of links accessed by the user, session IDs, and user authentication information, etc.

XSSer: This is another tool for exploiting and detection of XSS vulnerabilities in websites. It is an automated framework and an open-source vulnerability testing tool. The XSSer has too many types of built-in methods to bypass the security checks within a web application. It comes with the different techniques to inject XSS malicious code. This tool can run on different types of platforms and Python is needed to run it [26].

OWASP Xenotix XSS: This tool is also used for exploitation and to detect XSS vulnerability. This tool supports a timing base testing for auto and manual testing also. Which may be drive-by downloading payloads, client-side keystroke logging malware, and XSS injection. The much more features of this tool are null level false positive, API for integration within a web application, triple type of Python scripting, payloads, stealing of browser cache, XSS keylogging, XSS runnable drive-by download, auto testing of XSS and encoding. As per the author [27], this tool is rich in functionalities for the testing of XSS vulnerabilities and exploitation for the penetration testing of web applications. A few more features are defined below.

- ***Payload:*** It can insert 380 plus payloads and support the HTML5 XSS code injection payload.
- ***XSS Key Logger:*** This will record the keystrokes of the user as they type any data into websites. This can be used for the monitoring activities of targeted user and root access rights.
- ***XSS Encoder:*** With the help of this functionality the web application firewalls and other security controls can be bypassed to encode the forms like link encoding, HTML, Base64, and HEX encoding. And for checking the XSS filters validation just like XSS character encoder where different type of available web applications is used to allow users for creating an XSS code.
- ***XSS Testing:*** It has an auto mode for testing every payload, which can be time-specific and that can be defined by the tester to check loading time of website it all depends on their Internet link speed.

SSLyze: This tool is used for the analysis of SSL (Secure Sockets Layer) certificates configuration regarding web servers by creating a connection and it is developed in Python as a stand-alone. It will support the organizations and penetration testers for the identification of any weak SSL certificate configuration because it is the best to design, accuracy, and fast. This tool can be used as a plug-in by high tech users [28]. A few features are defined below.

- Test for non-secure with handshaking
- Look for the weak cryptographic algorithms
- The SSLv2, SSLv3 and TLSv1 checking
- Copy SSL certificate data and validate basic info
- Also client-side certificate validation support

Qualys SSL Labs: Another tool that is used for the scanning of network security and vulnerability checking mostly for cloud services. The complete information regarding SSL based website can be obtained. The tester should be aware of the risk that he is giving full access rights to another organization to check his SSL configuration. The Qualy's research company needs to know regarding SSL/TLS (Secure Sockets Layer/Transport Layer Security), KPI (Key Performance Indicator), other supporting tools, complete documentation for user guide and settings of that targeted web application. From 2009 without any charges, the online analysis of websites has been done about in several billion. They are also surveying SSL certificate setting Internet-wide after a specific period.

OpenSSL: For the SSL and TLS protocol implementation mostly this open-source tool is used. The essential cryptographic algorithms and different types of utility programs are implemented by its core library. The different types of computer languages can be used for the OpenSSL library with the usage of wrappers. As per the estimated 66% share of OpenSSL in the market [29]. It is licensed as per the Apache-Style license. With the help of OpenSSL, web application traffic will be encrypted instead of plan-text. The communication between source and destination started with handshaking for the selection of strong cryptographic algorithms to encrypt and decrypt the data. By doing this if an attacker can get control of the session and try to read the traffic data it will be look-alike unreadable for him due to encryption. So, it may take too many days to decrypt that data because it can be decrypted by the corresponding key. Due to this, we can say this is a secure and proactive approach for the protection of data [30].

Netcraft: This tool is used by penetration testers for the information gathering regarding the targeted website. It will provide better information for the traces of that website. This is used for the identification of which information can be leaked from the website and that is not secured properly so that there is a possibility of an attack on that vulnerability for the organization. Netcraft runs your DNS and sees what you have to fix and what you have to protect against. Netcraft gives web server and web hosting pieces of the pie investigation, including web server and working framework location. It can likewise tell to what extent the servers have been up, what their uptime is, the last time they were reboot, etc. Netcraft is exceptionally simple to utilize. Clients can visit netcraft.com and put the ideal space data to get all the subtleties of their objectives [31].

5. METHODS/TECHNIQUES

Data Mining: Data mining is used for the decision making with the different techniques to find out required information or footprints from big datasets analysis. The analysis of these big datasets is too much time taking task as there is a need for quick extraction of required information for data. At this point, the old method of data analysis is not enough but there is a need to find new methods for analyzing these big datasets [32]. So, these are integrated with different types of fields like databases, machine learning, statistical data analysis, and artificial intelligence [33]. The main objective of the data mining method is to build the evocative model or proactive model [34]. With the help of the static method in the dataset, the evocative model will be used for defining the basic properties of evocative. On another end, the proactive model is used to analyze the datasets for new data future actions. The proactive model in data mining contains these features: rule binding, categorization, relapse analysis, and tendency analysis [35]. Data mining is more efficient and powerful to be used indifferent types of applications. Due to its automated process of searching and to extract the footprints from big datasets. After that these big datasets are analyzed to find out the logical relations and also transfer that data into new methods that can be used for further analysis.

Machine Learning: Machine learning is a computer program that developed on statistical theory and mathematical models to get more optimized results from data or old results [36]. The main functionality of machine learning is to solve issues and tasks automatically [37]. For achieving these required functions there are two types of techniques are used: first is managed learning (grouping, support vector machine, neural networks) and second is known as unmanaged learning (collecting, dimensionality lessening, recommender systems, remoteness, and stabilization). As per practical use, these learning methods are the same, but they are different at the level of their usage. Such as if we do not know about the dataset then the unmanaged learning method will be applied. And if we know something about the dataset then the managed learning method will be applied [38].

Deception: Interestingly, deception-based identification gives a powerful option in contrast to the detection of an anomaly in applications. Any part in a venture system, for example, a PC framework, a help, an accreditation, an information thing,etc which can be utilized for deception-based recognition. The original deception technologies are a honeypot, shown the viability of misleading as a component of a layered security methodology. A few deception strategies have been proposed

to be incorporated into the application layer of web applications, for example, utilizing two instruments dependent on parameter infusion into web application traffic [39], and conveying a deception-based intermediary to recognize any conceivable bogus caution [40]. Subsequently, there is as yet a hole to fill with regards to the advancement of new extra methods to different sorts of web application attacks. Cyber Security is like military circumstances in that the aggressor has an immediate favorable position of being the initiator. In a digital domain, digital aggressors have the additional points of interest to get obscure with their malevolent goal and probably won't be perceptible to the protector until they succeed [41].

The security professionals can be one step ahead of hackers if the deception method is applied in the right way. It can be achieved with the traps of system services by which attackers attack methods can be detected at an early stage. To stop the hacker's attack or to break the chain of attack as soon as possible is the priority of security professionals. So, there are too many methods to setup the deception method in cyber security systems like as at network, operating system, web applications, and information layer [42]. The deception method deployed at the host level is known as the system layer. The deception method deployed for the application functions like for websites or database servers is known as the application layer.

Different deception methods proposed by the research, just as practices by the business. Whaley characterizes and arranges deception into two principal classes: suppression and recreation [43]. Suppression includes the procedure of concealing genuine data through camouflaged and deception of data towards conceivable unapproved testing. Recreation then again includes a procedure of making a procedure or administration unknown to segregate it from being attached to a particular procedure. There exists a likelihood that the strategy could be actualized in blend or progressive to add to its adequacy as countermeasures to potential attacks.

6. WEBSITE SECURITY FIREWALLS

For the detection and prevention of website attacks, there are too many methods and firewalls recommended by researchers. So, to prevent the web application from these new security issues different types of methods are developed. But as per current research two techniques are in place [44]:

- ***Behavioral-Base:*** For the detection of unknown attacks this behavioral-based technique is used. But the big issue with this technique is its performance and ratio of false-positive for perfection.
- ***Signature-Base:*** This is another method in use to overcome the accuracy issues and performance of the behavioral-based method. In this method already known attacks rules are defined to detect known attacks and get results on the form of a less false-positive ratio. But in this technique, new attacks cannot be detected.

Both types of techniques are used for security problem solutions at different places and a variety of applications. The overview as per the OSI layer model for these firewalls is depicted in Fig. 4.

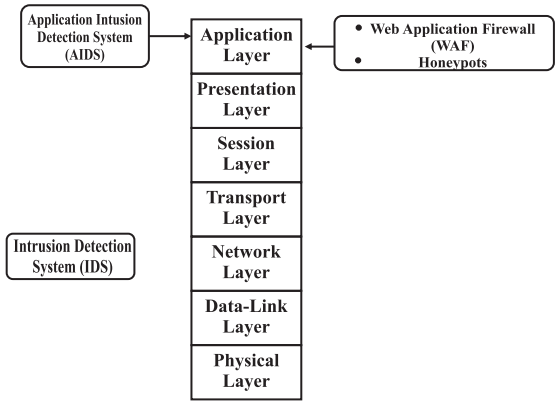

FIG. 4. OSI LAYER BASED WEB APPLICATION FIREWALLS

Web Application Firewall: The more common firewall is used for the detection and prevention of website attacks is a WAF (Web Application Firewall). It is considered a good firewall by security professional because operate at the application layer for the monitoring website traffic. This is also used for the detection of malicious traffic of web servers. The researchers are believing that WAF is a powerful tool for preventing data losses and mitigate attacks [45]. But it was still some threatening shortcomings.

There are numerous impediments to the web application firewall which makes it is not a good solution. These are high false negatives and high false positives rates, low exactness, and powerlessness to distinguish obscure attacks, notwithstanding expanding operational expense and manual endeavors. As of late, these web application firewall impediments have been tended to by analysts in numerous ways. For instance, applying automated systems, for example, AI and information mining calculations as in being that as it may. It will bring up the issue of execution as the web applications work continuously with high traffic. Incomprehensibly, applying methods to upgrade the exhibition of information mining, what's more, AI calculations will likewise bring about a decrease of precision. In like manner, sending an equipment web application firewall to withstand the weight of the presentation can bring about a significant expense.

Application Intrusion Detection System: The scholars have

recommended another type of firewall which is known as an AIDS (Application Intrusion Detection System) to fulfill the shortcomings of WAF [46]. AIDS has fulfilled the gap of IDS (Intrusion Detection System) issues at the network. These AIDS can be deployed along the side of existing firewalls to increase another security layer for web applications. As per the current implementation of IDSs are using behavioral-based or signature-based detection methods, but some security professionals are using both methods at the same time. So that the signature-based firewall performance is better as compared with a behavioral-based firewall for the known attacks. Due to this advantage, the signature-based is more implemented by paid firewall devices and other side behavioral-based are less in use by them. The behavioral-based is mostly used by scholars, due to its functionality for detection or prevention from unknown attacks.

Web Application Honeypot: The technique used by the honeypot is different from WAF and AIDS. As the attack is performed on any system or application then honeypot can be used as the proof of that attack. The honeypot is used as distraction and deception methods for the monitoring of malicious traffic to get information which is not possible with other methods [47]. It is divided into two parts: research base honeypot and honeypots in the production environment. The objective of honeypot in production to protect the companies directly from attacks. While the research base honeypot is used to collect the information regarding an attacker's behavior or method of attack and give security passively. The honeypots are further divided as per their usage for applications or systems which are known as high interaction honeypots and low interaction honeypots [48].

The issue of false-positive faces in WAF and AIDS is decreased with the honeypots. But the honeypots are still collecting a smaller number of results for false-positive instead of that more useful information can be collected. The companies will remain safe from the direct attack with the deployment of honeypot for a web application or network. From the last decade, more work has been done for the web application honeypots [49-50] all are focusing on the decoy and extending the suppression. From the last few years' scholars are developing honeypots for auto-generation of signatures for IDS [51] monitoring network traffic and website attacks observations. With this can get information regarding attack patterns and the intent of attackers. With the development of these types of honeypots, it will support us to defeat the cybercriminal, trace out attackers, attack detection, and the website attack detection will be easier.

Web Application Forensics: Another technique has been recommended by scholars, web application forensic can be considered as part of digital forensic for gathering and analyze the attacker's actions. This can be used for finding the footprints or any other security breach on web applications [52]. Such as, it is used for the investigation of website downtime issues or how this application is hacked by an attacker. The forensic investigation of any web application attack can be done within an organization or by any external part will do it. Both types of investigations are performed with the same process for checking law violations or for any weakness in websites. As for the technical level of the web, application forensics are given below.

- The detection of the attack process.
- To find the fingerprints of attack, investigation, and intent of the attacker.
- The more valuable information can be collected as compared with other detection methods.

The skills and expertise of a forensic person are depending on his knowledge regarding the latest tools and methods. As the number of attacks is growing day by day and the amount of data is larger due to this it is very tough to find out the evidence with old forensic tools and processes. The current best practice for finding evidence with the help of log analysis from servers and firewalls.

A complete study of web application forensic tools is defined. As per the study, the greater part of the apparatus's centers around the packed information, the relationship of the different sources, and details. Notwithstanding, a monstrous measure of information created from overwhelming web traffic is driving conventional techniques and apparatuses to get incapable; joined by increments in time, cost, and endeavors. Subsequently, analysts began to scan for progressively viable arrangements. Secure web application development suggestion is given in Table 1.

TABLE 1. SECURE WEB APPLICATION DEVELOPMENT SUGGESTIONS

No.	Suggestion for Security of Web Applications
1.	Input Fields Validation
2.	Latest version of Frameworks, Modules, Webserver should be used
3.	Check the third-party API's for hidden vulnerabilities
4.	Application Response Headers should be sanitized
5.	Application errors should be handled properly
6.	Application sessions should be managed as per requirements
7.	Access control should handle properly as user's role
8.	Time to time web application should be scanned for vulnerabilities
9.	Test password for admin users should be changed
10.	Use strong passwords to avoid brute force attacks
11.	Web application should be deployed with standard security configurations

7. CONCLUSIONS

In this research have taken a brief look into a different type of web application attacks, tools, and techniques with a detailed description. As the last decade or more research regarding website attacks, there are two types of top attacks SQL injection and XSS injection. These two attacks are more dangerous in the form of DoS attacks, data breaches, user privacy, access control, and forging the data for legitimate users. Along with these attacks are discussed other attacks also which are threatening to steal information without user permissions way such as, broken authentication, using third party APIs, or frameworks with vulnerabilities, security misconfiguration. The different types of tools for detection and prevention from these attacks are described in this paper. Some of them are open-source and paid also. For the detection of SQL injection, XSS attacks, SSL certificate weaknesses, and gathering information regarding the target website by using automated tools. Different type of techniques is also recommended by scholars such as WAF, AIDS, Data mining, and web application forensics. The WAF has a limitation it can detect or block known attacks only because it works on a signature base. At other ends, AIDS can work on a signature base or behavioral base but as it works on behavioral base then issue of performance with this firewall. For data mining and web application, there is too much time required to analyze the big data sets for tracing the attackers or his intent. So, recommendations in this paper are the application should properly code, input fields should be validated and at the time of application design, any security standard should be added as a primary part of web application development.

ACKNOWLEDGMENT

Thankful to unknown Reviewers/Experts for valuable comments/suggestions to improve the research papers.

REFERENCES

[1] Halfond, W.G., Viegas, J., and Orso, A., "A Classification of SQL-Injection Attacks and Countermeasures", Proceedings of IEEE International Symposium on Secure Software Engineering, Volume 1, pp. 13-15, March, 2006.

[2] Tajpour, A., Heydari, M.Z., Masrom, M., and Ibrahim, S., "SQL Injection Detection and Prevention Tools Assessment", IEEE 3rd International Conference on Computer Science and Information Technology, Volume 9, pp. 518-522, July, 2010.

[3] Hou, B., Qian, K., Li, L., Shi, Y., Tao, L., and Liu, J., "MongoDB NoSQL Injection Analysis and Detection", IEEE 3rd International Conference on Cyber Security and Cloud Computing, pp. 75-78, June 2016.

[4] "OWASP Attack Category",https://www.owasp.org/ index.php/ Category:Injection, (Accessed on 10, October 2019).

[5] Nagpal, N.B., and Nagpal, B., "Preventive Measures for Securing Web Applications Using Broken Authentication and Session Management Attacks: A Study", International Conference on Advances in Computer Engineering and Applications, 2014.

[6] Hassan, M.M., Ali, M.A., Bhuiyan, T., Sharif, M.H., and Biswas, S., "Quantitative Assessment on Broken Access Control Vulnerability in Web Applications", International Conference on Cyber Security and Computer Science, Safranbolu, Turkey, 18-20 October, 2018.

[7] Wang, M., Lu, Y., and Qin, J., "A Dynamic MLP-Based DDoS Attack Detection Method Using Feature Selection and Feedback", Computers & Security, Volume 88, pp. 101645, 2020.

[8] Mirea, M., Wang, V., and Jung, J., "The Not so Dark Side of the Darknet: A Qualitative Study", Security Journal, Volume 32, No. 2, pp. 102-118, 2019.

[9] Kaur, S., and Randhawa, S., "Dark Web: A Web of Crimes", Wireless Personal Communications, pp. 1-28, 2020.

[10] Chaudhari, G.R., and Vaidya, M.V., "A Survey on Security and Vulnerabilities of Web Applications", International Journal of Computer Science and Information Technologies, Volume 5, No. 2, pp. 1856-1860, 2014.

[11] Srivani, P., Ramachandram, S., and Sridevi, R., "A Survey on Client-Side and Server-Side Approaches to Secure Web Applications", IEEE International Conference of Electronics, Communication, and Aerospace Technology, Volume 1, pp. 22-27, April, 2017.

[12] Makanadar, S., and Solankurkar, V., "An Approach to Detect and Prevent SQL Injection Attacks using Web Service", International Journal of Science and Research, Volume 2, No. 4, pp. 242-245, 2013.

[13] Choudhary, A.S., and Dhore, M.L., "CIDT: Detection of Malicious Code Injection Attacks on Web Applications", International Journal of Computer Applications, Volume 52, No. 2, 2012.

[14] Halfond, W.G., Viegas, J., and Orso, A., "A Classification of SQL-Injection Attacks and Countermeasures", Proceedings of IEEE International Symposium on Secure Software Engineering, Volume 1, pp. 13-15, March, 2006.

[15] Hou, B., Shi, Y., Qian, K., and Tao, L., "Towards Analyzing MongoDBNoSQL Security and Designing Injection Defense Solution", IEEE 3rd International Conference on Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing, and IEEE International Conference on Intelligent Data and Security, pp. 90-95, May, 2017.

[16] Mittal, N., "MongoDB Security – Injection Attacks with PHP", https://blog.securelayer7.net/mongodb-security-injection-attacks-with-php/, (Accessed on 27, January 2020).

[17] Okman, L., Gal-Oz, N., Gonen, Y., Gudes, E., and Abramov, J., "Security Issues in NoSQL Databases", IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications, pp. 541-547, November, 2011.

[18] Johari, R., and Sharma, P., "A Survey on Web Application

Vulnerabilities (SQLIA, XSS) Exploitation and Security Engine for SQL Injection", IEEE International Conference on Communication Systems and Network Technologies, pp. 453-458, May, 2012.

[19] Gupta, S., and Gupta, B.B., "Cross-Site Scripting (XSS) Attacks and Defense Mechanisms: Classification and State-of-the-Art", International Journal of System Assurance Engineering and Management, Volume 8, No. 1, pp. 512-530, 2017.

[20] Putthacharoen, R., and Bunyatnoparat, P., "Protecting Cookies from Cross-Site Script Attacks using Dynamic Cookies Rewriting Techniques", IEEE 13th International Conference on Advanced Communication Technology, pp. 1090-1094, February, 2011.

[21] El-Moussaid, N.E., and Toumanari, A., "Web Application Attacks Detection: A Survey and Classification", International Journal of Computer Applications, Volume 103, No. 12, 2014.

[22] Kuchiwale, S.L., and Lolge, S., "A Survey on Website Attack Detection and Prevention", International Journal of Advanced Engineering and Global Technology, Volume 03, No. 01, 2015.

[23] Singh, A., "Instant Kali Linux", Packt Publishing Ltd, 2013.

[24] Pundlik, S., Kumar, R., Gaikwad, B., Aadhale, A., and Waghmare, V., "SQLijhs: SQL Injection Attack Handling System", International Journal of Engineering Research and Technology Volume 2, No. 6, ESRSA Publications, 2013.

[25] Nagpal, B., Singh, N., Chauhan, N., and Panesar, A., "Tool Based Implementation of SQL Injection for Penetration Testing", IEEE International Conference on Computing, Communication & Automation, pp. 746-749, May, 2015.

[26] Alzahrani, A., Alqazzaz, A., Zhu, Y., Fu, H., and Almashfi, N., "Web Application Security Tools Analysis", IEEE 3rd International Conference on Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing, and IEEE International Conference on Intelligent Data and Security, pp. 237-242, May, 2017.

[27] Bau, J., Bursztein, E., Gupta, D., and Mitchell, J., "State-of-the-Art: Automated Black-Box Web Application Vulnerability Testing", IEEE Symposium on Security and Privacy, pp. 332-345, May, 2010.

[28] "Q. Inc. SSL Server Rating Guide", [Online]. Available: https://www.ssllabs.com/, (Accessed on 20, January 2020).

[29] "Web Server Survey",https://news.netcraft.com/archives2014/04/02/april-2014-web-server-survey.html, (Accessed on 20, January 2020).

[30] Viega, J., Messier, M., and Chandra, P., "Network Security with OpenSSL: Cryptography for Secure Communications", O'Reilly Media, Inc., 2002.

[31] Bruns, A., "From Presumption to Produsage", Handbook on the Digital Creative Economy, Edward Elgar Publishing, 2013.

[32] Roiger, R.J., and Geatz, M.W., "Data Mining: A Tutorial-BPrimer", Pearson Education. Inc: USA, 2003.

[33] Mirza, N., Patil, B., Mirza, T., and Auti, R., "Evaluating the Efficiency of the Classifier for Email Spam Detector Using Hybrid Feature Selection Approaches", IEEE International Conference on Intelligent Computing and Control Systems, pp. 735-740, June, 2017.

[34] Ks, D., and Kamath, A., "Survey on Techniques of Data Mining and Its Applications", International Journal of Emerging Research in Management &Technology, Volume 6, No. 2, pp. 198-201, 2017.

[35] Han, J., Kamber, M., and Pei, J., "Data Mining: Concepts and Techniques", San Francisco: Morgan Kauffman, 2006.

[36] Alpaydin, E., "Introduction to Machine learning", MIT Press, 2014.

[37] Chellapilla, K., and Simard, P.Y., "Using Machine Learning to Break Visual Human Interaction Proofs (HIPs)", Advances in Neural Information Processing Systems, pp. 265-272, 2005.

[38] Hotho, A., Nürnberger, A., and Paaß, G., "A Brief Survey of Text Mining", Ldv Forum, Volume 20, No. 1, pp. 19-62, May, 2005.

[39] Pan, G., Seow, P.S., Chan, C., and Lim, C.Y., "Analytics and Cybersecurity: The Shape of Things to Come", Institutional Knowledge, Singapore Management University, 2015.

[40] Han, X., Kheir, N., and Balzarotti, D., "Evaluation of Deception-Based Web Attack Detection", Proceedings of Workshop on Moving Target Defense, pp. 65-73, October, 2017.

[41] Gupta, B., and Jyoti, K., "Big Data Analytics with Hadoop to Analyze Targeted Attacks on Enterprise Data", International Journal of Computer Science and Information Technologies, Volume 5, No. 3, pp. 3867-3870, 2014.

[42] Han, X., Kheir, N., and Balzarotti, D., "Deception Techniques in Computer Security: A Research Perspective", ACM Computing Surveys, Volume 51, No. 4,pp. 1-36, 2018.

[43] Whaley, B., "Toward a General Theory of Deception", Military Deception and Strategic Surprise, pp. 186-204. Routledge, 2012.

[44] Kokkonen, T., "An Anomaly-Based Online Intrusion Detection System as a Sensor for Cybersecurity Situational Awareness System", Jyväskylä Studies in Computing, pp. 251, 2016.

[45] Olayemi, O., Antti, V., Keijo, H., and Pekka, T., "Security Issues in Smart Homes and Mobile Health System: Threat Analysis, Possible Countermeasures, and Lessons Learned",International Journal onInformation Technologies & Security, Volume 9, No. 1, pp. 31-50, 2017.

[46] Park, Y., and Park, J., "Web Application Intrusion Detection System for Input Validation Attack", IEEE 3rd International Conference on Convergence and Hybrid Information Technology, Volume 2, pp. 498-504, November, 2008.

[47] Tahir, S., and Iqbal, W., "Big Data - An Evolving Concern for Forensic Investigators", IEEE 1st International Conference on Anti-Cybercrime, pp. 1-6, November, 2015.

[48] Gupta, M.K., Govil, M.C., and Singh, G., "An Approach to Minimize False Positive in SQLI Vulnerabilities Detection Techniques Through Data Mining", IEEE International Conference on Signal Propagation and Computer Technology, pp. 407-410, July, 2014.

[49] Watson, D., and Riden, J., "The Honeynet Project: Data Collection Tools, Infrastructure, Archives, and Analysis", IEEE Workshop on Information Security Threats Data Collection and Sharing, pp. 24-30, April, 2008.

[50] Müter, M., Freiling, F., Holz, T., and Matthews, J., "A Generic Toolkit for Converting Web Applications into High-Interaction Honeypots", University of Mannheim, Volume 280, pp. 6-1, 2008.

[51] Djanali, S., Arunanto, F.X., Pratomo, B.A., Baihaqi, A., Studiawan, H., and Shiddiqi, A.M., "Aggressive Web Application Honeypot for Exposing the Attacker's Identity", IEEE 1st International Conference on Information Technology, Computer, and Electrical Engineering, pp. 212-216, November, 2014.

[52] Kyaw, A.K., Sioquim, F., and Joseph, J., "The Dictionary Attack on WordPress: Security and Forensic Analysis", IEEE 2nd International Conference on Information Security and Cyber Forensics, pp. 158-164, November 2015.

Information for Authors

BEFORE SUBMISSION

- Before Submission of an article, authors should carefully read the Publication Ethics, Open Access policy and Plagiarism Policy of the Journal.
- Authors should ensure that the work has not been published previously and it is not under consideration for publication elsewhere.
- Publication must be approved by all authors of the paper.
- Authors should carefully list and order their names on the first page of the article. An author's name will not be added or deleted after submission. One of the author should be nominated as Correspoding Author.
- An article should be written in good English.

SUBMISSION

- Authors may submit an article through our on line submission system or through an email as a MS Word file PDF file is not acceptable. In case of on line submission, author will be required to Sign up. On successful registration, author can upload the article. Alternatively, an artic le may be sent through email to the Editor-in-Chief, or Focal/Coordinating Person.
- Authors should note that all submitted articles are referred through a double blind peer review process (National and International) which means that author's identities remain unknown to reviewers and vice versa throughout the reviewing process.
- Articles should be formatted in single or double spacing, preferably in Times New Roman size 12 font. Accepted articles will be correctly formatted by our publication team.
- The first page should contain the full title of the article and full names and affiliations of all authors including email address of the corresponding author.
- The format of the manuscript is Abstract of up to 200-300 words, Keywords, Introduction then main body of the paper ending with conclusion followed by acknowledgement and list of references.
- References should be cited in the text by number within square brackets. At the end of the paper, they should be listed in the order in which they appear in the text.

AFTER ACCEPTANCE

- Authors will be notified about the acceptance/rejection of the paper through email.
- Upon acceptance of an article, authors will sign copy right and Authorship form.
- Accepted manuscript will be sent to the corresponding author for proof reading. No editing or proofreading will be allowed after publication of an article.
- Authors to give Undertaking/Copy Right that this Research Paper is their original work/contribution and not published before in any Journal or in any Conference and contribution of individual authors regarding submitted research paper.