Broken Authentication and Session Management Vulnerability: A Case Study Of Web Application

Md. Maruf Hassan*[1,2], Shamima Sultana Nipa[1], Marjan Akter[1], Rafita Haque[2], Fabiha Nawar Deepa[2], Mostafijur Rahman[1,2], Md. Asif Siddiqui[1], Md. Hasan Sharif[1]

[1]Cyber Security Centre, Daffodil International University, Bangladesh
[2]Software Engineering Department, Daffodil International University, Bangladesh
Email: maruf.swe@diu.edu.bd, shamimanipa743@gmail.com, marjanshifat95@gmail.com, rafitahaque93@gmail.com, fabiha.deepa@gmail.com, mostafijur.swe@diu.edu.bd, sharif.swe@diu.edu.bd

***Abstract* — Web applications have extensively taken over the roles of atomization and enhancement of prevailing solutions. It also provides different services to the multiple users of the application. In the recent time, performance of the web services are measured through two important properties such as authentication and session management. However, user authentication appears to be crucial when a valid user of the web application inappropriately discontinues their communication while the session remains active and an unauthorized user pick the same session to get access into the system. Broken Authentication and Session Management vulnerability exploitation risk is becoming enormously higher due to attackers creative skills, system's weak design and improper implementation of web applications. The consequence of the above exploitation may result not only identity theft but also removal/tamper confidential information. This paper has analyzed the authentication vulnerability attack i.e. Broken Authentication and Session Management, its exploitation types and their impact upon investigating on 267 websites of public and private sectors in Bangladesh. 56% websites of our samples were found vulnerable with the given weaknesses by conducting the examination using manual penetration testing method following double blind testing strategy. The result shows the impact and percentage of this vulnerability attacks.**

Keywords* — *Cyber Security; Web Application Vulnerabilities; Web Application Exploitation Techniques; Broken Authentication; Session Management.

I. INTRODUCTION

More than three billions of people around the world are using internet as well as web applications via a variety of different devices because of the friendly usability and easy accessibility to anywhere at any time[1]. Currently, web application is the first step to automate the basic activities of the day to day life by upgrading the existing solutions. Due to the above reason, most of the organizations or service providers e.g. industry, bank, government, educational, medical, and other sectors like to provide their services to their service holders through web application. On the contrary, risks of exploitation of these web applications are increasing every day through different cyber attackers. A survey reveals that more than 82.8% of web service providers are using PHP platform to build their web applications for the easier code practicing [2].

There are some common types of vulnerability available in web application such as Structured Query Language Injection (SQLi) [3], Cross Site Scripting (XSS) [4], Cross Site Request Forgery (CSRF) [5], Local File Inclusion (LFI) [6], Remote File Inclusion (RFI) [7], Local File Disclosure (LFD) [8], Broken Authentication, Session Management, etc. Broken Authentication and Session Management vulnerabilities are often found due to improper implementation of user authentication and management of active session which is one of the top two risks according to OWASP [33]. Although different frameworks and functions provide proper authentication and session management; however, customized authentication and Session Management are built often by developers which may lead to exploit Broken Authentication and Session Management vulnerabilities. In 2015, Bangladesh faced a cyber-war against Pakistan where the Pakistani hackers defaced more than 180 web sites of Bangladesh for having issues of the Broken Authentication. Percentages of vulnerability exploitations by the Pakistani hackers were 63% of Broken Authentication vulnerability, SQL injection in 26% sites, and other exploitations conducted on 11% of the web applicant [9].

An assessment and analysis on Broken Authentication and Session Management vulnerability and its five exploitation types are discussed in this paper. Those techniques have been implemented on the different organization's web application of public and private sector in Bangladesh. This paper is organized in seven sections. Introduction and Literature Review are discussed in section I and II respectively. Overview of Broken Authentication and Session Management vulnerability are explained in section III. Section IV describes data collection procedure. After performing data analysis, result and statistics are furnished in section V and prevention techniques from Broken Authentication and Session Management vulnerabilities are described in section VI. The paper is concluded with the outcome of the study, limitation, and future work in section VII.

II. Literature Review

There have been a number of researches conducted on web application vulnerabilities. About five web application vulnerabilities i.e. SQLi, Inclusion Attack [10], XSS, Brute Forcing Attack [11] and Insecure Cryptographic Storage [12] discussed in the article where the authors suggested some recommendation to get rid of those vulnerabilities. A survey conducted on internet usages in which the author figured out that 60% of resources in the internet were not in safe position due to the existence of the application level vulnerabilities [10]. Another survey performed on the various types of SQLi and XSS vulnerabilities in web application where the author of the article suggested some countermeasures to defend those attacks [13]. Review on the most prevailing vulnerabilities on web applications were exploited using different hacking tools and preventive guidelines were also provided through a solution of those attacks [14]. An examination performed to detect the existence of various web vulnerabilities. The survey performed on 110 web sites and revealed the cause of application layer vulnerabilities [15]. Three main factors were also identified for the above reason that includes lack of experience, lack of knowledge in web security programming, and neglecting of using the encryption methods [16].

Few investigations on Broken Authentication and Session Management vulnerabilities are also found in recent years. A study conducted on SQLi, Broken Authentication, Session Management, and XSS web application vulnerability. The author discussed the code level problem analysis of those application layer weaknesses and recommended a guideline for the developers to secure the web application [17]. A study performed on root cause analysis to detect the Session Management and Broken Authentication vulnerabilities and prescribed solutions have been given to reduce the recurring attack of the web application [18]. Process of identifying the Broken Authentication vulnerability, attack procedure, and prescribed guidelines were discussed to protect the web-based system from the intruder [19]. A technique Nemesis, used for preventing access control vulnerabilities and Exploiting Authentication problem on web application are presented in the paper. The author implemented Nemesis through a tool by which the developer can control the given vulnerabilities in a small amount of time [20]. The study explaining the types of Broken Exploiting Authentication problem and Session Management attacks of web applications. Precautionary measures of the given problems were also illustrated at the end of the study [21].

Examination on exploitation techniques of different web vulnerabilities were also conducted through a case study in various domains of Bangladesh. Reason, code review and three exploitation techniques of SQLi were discussed in the article and impact after exploiting on .bd domain, educational websites, and financial web applications of Bangladesh were assessed. The authors also prescribed some precautions to mitigate the above risk [24, 25]. Study on XSS and CSRF vulnerability in the web applications in Bangladesh was evaluated through different exploitation techniques. The author also drew an attention for the developers about the coding flaws that are frequently be made during the development period. [26]. Detailed description on LFI vulnerability exploitation techniques based on RFI and SQLi were discussed in the paper and showed the impact after conducting the given exploitation on 153 web application of Bangladesh . LFD exploitation techniques and impact after exploitation on educational websites in Bangladesh were presented in the article [25].

After reviewing the above literature, it is found that insignificant research work on Broken Authentication and Session Management have been done by this time and lack of study on the above vulnerabilities is also present in Bangladeshi domain.

III. BROKEN AUTHENTICATION & SESSION MANAGEMENT

Broken Authentication is a kind of web vulnerability which occurs due to the misconfiguration of session management. After an authentication process completed, a session will be created which will be activated for data communication between the server and a particular user. Fig. 1 represents the problem of Broken Authentication by exploiting session mismanagement problem. If any intruder can get access in the active session of any specific user bypassing the authentication process, the scenario is treated as broken Exploiting Authentication problem of the given application. Fig. 1 represents the overall process of user authentication and session management.

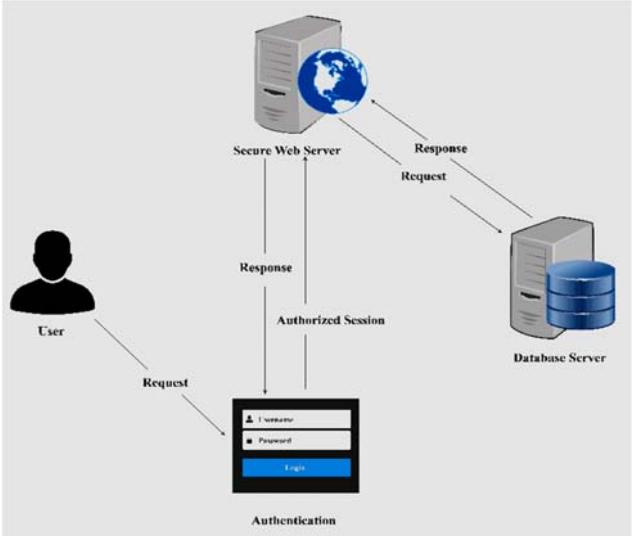

Figure 1. Authentication and Session Management Process

A session request is raised by a user of a web application through the login page where the user credential has been provided. Once the given request has been sent from the client side to server side, the server initiates a query to the database for checking whether the user provided credential is matched

with the record of the database or not. As soon as the validation process is successful, a session with a specific ID will be allocated for the user to communicate the application. A user then can access the system with a given privileges provided by the administrator of the system for getting different services. A valid session works for a certain duration which is predefined by the system designer. Browsers stores the user credential in the authentication cookie [27] so that the session will remain continue once the session is expired its period by sending the authentication information to the server side. This process is performed automatically behind the user interface which will reduce the effort of the user to authenticate [28] they repeated. However, the intruder can catch and get access into other's active session by using different applications like, cookie manager, eat my cookie, advanced cookie manager+, etc., in case the user missed to close the session as directed by the application designer.

There are some exploitation techniques used exploit Broken Authentication & Session Management. Types of the above are given below:

A. General Broken Authentication & Session Management Exploitation

Broken Authentication and Session Management have different types of exploitation techniques that are discussed in this paper. Manual penetration testing method [29] has been used to check the above vulnerability of web applications in public and private sector of Bangladesh. Fig. 2 represents the general Broken Authentication & Session Management exploitation technique.

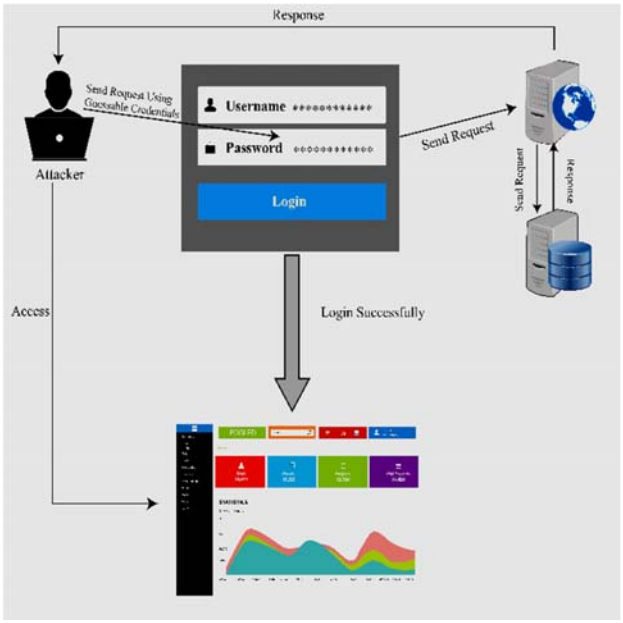

Figure 2. General Broken Authentication & Session Management Exploitation.

It continuously sends the request of produced user credential until the system finds it correct. As soon as the guessable credentials are matched with database, the system sends a response to the attacker with the access in the account or admin panel. It is mentionable here that many systems are easily exploitable due to use the weak passwords like admin admin, admin123, etc.

B. Exploitation Techniques

The five types of Broken Authentication & Session Management exploitation techniques are discussed below.

a) *Session Misconfiguration Attack:* Session duration is one of the major facts in maintaining a secure authentication process of the web applications. As soon as the user credential is validated from a system, it assigns a session for the particular user with a session ID for a limited period of time. In case the developer of the web application sets the session duration parameter with a large value, the session will remain active for that specific period if the user not logged off their account as directed by the designer of the application.

Therefore, that session can be reestablished to re-using by an intruder which leads to Broken Authentication. Session misconfiguration is one of the most critical areas for Broken Authentication and Session Management vulnerability. Attacker uses the browser, Google dork, and no-redirection add-ons for bypassing admin panel in session misconfiguration exploitation process. Session misconfiguration exploitation processes are described in four steps.

Step 01: Attacker uses Google dork to search vulnerable web sites e.g. inurl:apanel/admin/;
Step 02: Google returns the list of possible vulnerable web site list (in figure 3);

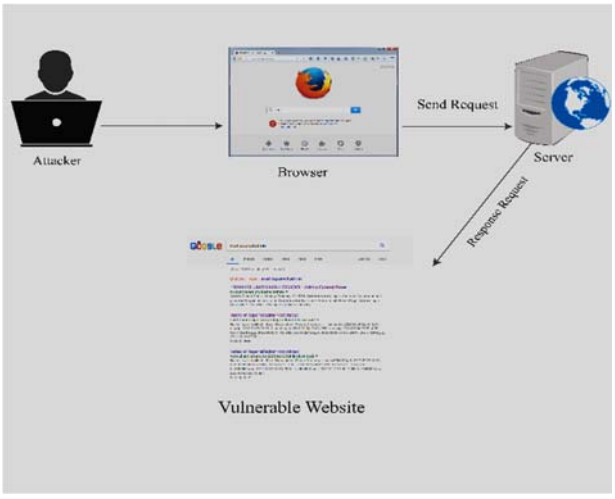

Figure 3. Search vulnerable web site using Google dork.

Step 03: The possible vulnerable web site lists have been observed (in Fig. 4) and attacker selects the specific URLs with index file like, *inurl:apanel/admin/index.php.*

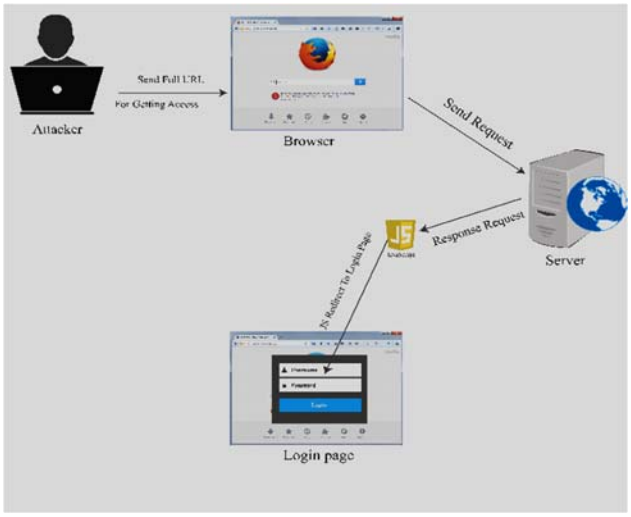

Figure 4. Session Misconfiguration Attack Exploitation process.

Browser sends a request to the server to get access to the user admin panel directly without using username and password. Java script redirects the request and send attacker to the login page.

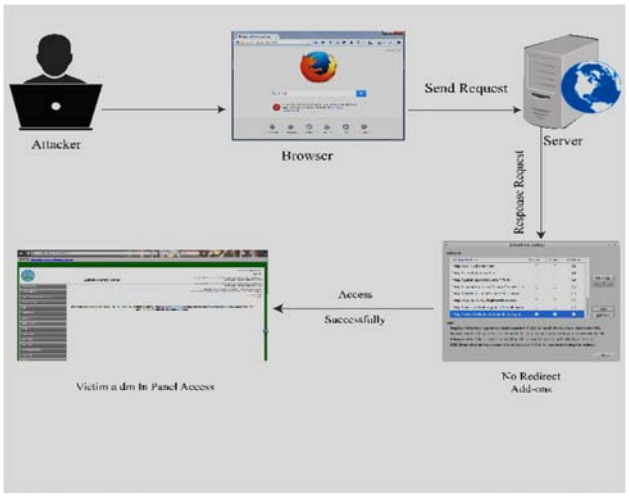

Figure 5. Bypassing admin panel using no-redirect add-ons in Firefox browser

Step 04: Fig. 5 shows the installation process of no-redirect add-ons in Firefox browser where the target website URL are added in the add-ons for getting access in admin panel successfully by preventing re-direction of the request. No-redirect add-ons helps attacker to bypass the admin panel successfully, and intruder gets privilege the access into the system.

ii) Using Cracking/ Guessing Weak Password Exploitation

Due to lack of awareness about password management, some non-technical users keep their password in a generalize form like admin, password, mypassword, password123, admin1997 etc. and also in some cases, user remains the default password for their access into the system which will be easy to guess for an attacker to get access in the system. It is an automated process of cracking/ guessing user's weak passwords. Attacker gives user login link in Hydra in which it checks predefined dataset for trying to find username and password. Fig. 6 represents Cracking/ Guessing Weak Password exploitation using multiprotocol brute force tool Hydra.

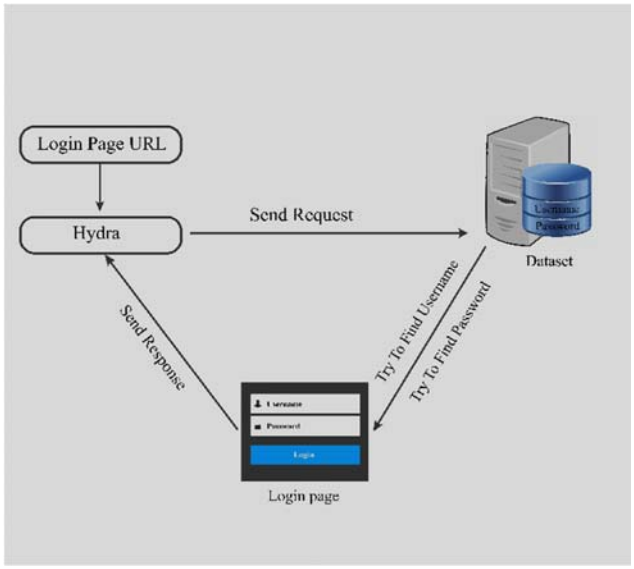

Figure 6. Cracking/ Guessing Weak Password exploitation using multiprotocol brute force tool Hydra.

Finally, Hydra shows whether username and password is found or not.

iii) Exploiting Authentication problem

Web applications authentication systems are handled by using conditional quires to check username and password against one user for authentication. If these conditional queries get infected or not properly handled, it could easily compromised by an intruder to get access into the system without proper authentication.

iv) Decoding Inadequate Encryption

In some web applications privacy measures are not properly handled by the developers. Therefore, an attacker can steal the session ID against one user by exploiting the security flaws of disclosing the session ID in the URL of the system, e.g.

http://www.demosite.com/transactions/saleitems?sessionid=7892384838&dest=demouser

The example shows the general transaction's session id of demouser has been disclosed publicly in the URL. As such, it is not very critical for an attacker to steal some other user's

session id just only changing the session ID value into the URL. The attack process is feasible for the inadequate encryption in the value of session ID. After changing the value in session ID, it will look like as below:

http://www.demosite.com/transactions/saleitems?sessionid=7892384839&dest=attackername.

v) Other Vulnerabilities

Web application vulnerabilities allow users to disclose users/ systems sensitive information. It also causes major harm to other circumstance e.g. it allows users to execute malicious quires in the system if the system is vulnerable to XSS vulnerability, it also allows attackers to post malicious links for phishing to steal session of the victim, etc. Forgotten password functionally, relying on IP address for session, emailing user credentials, not authenticating a user before changing password, and not having adequate timeouts for inactive session are also reason for Broken Authentication.

IV. DATA COLLECTION PROCEDURE

Manual penetration testing method [29] has been used following the double blinded strategy [30] to perform the study. The small sample techniques [31] has also selected to determine the sample size for this study. Different exploitation techniques of Broken Authentication in the application where Session Management problem existed have been used in this research. To identify the vulnerable (i.e. Broken Authentication and Session Management vulnerability) web application of public and private sectors of Bangladesh, it was searched in http://www.google.com/ using several operators. The most used operators in our examinations are i.e.

inurl:apanel/admin/index.php, inurl:news.php?id=, inurl:gallery.php?id=, inurl:article.php?ID=, and inurl:event.php?id=.

After finding our desired vulnerable web applications, it then be exploited using different Broken Authentication and Session Management exploitation technique described above and find out the types of given vulnerability existed in the application. Level of access after exploitation has been identified in this study.

Environment & tools

Hydra tool [37] has been used for brute force attack to cracking the weak password. Mozilla Firefox version 54.0 has been used to stop java script re-direction. No redirection plugins version 1.3.2.13.1 and JS switch 0.2.10.1 add-ons have also been used.

V. RESULT AND STATISTICS

Small sample technique has been selected as sampling method for this study. The above technique has been constructed using the Equation (1) [32]:

$$s = X^2 + NP(1-P) \div d^2(N-1) + X^2P(1-P) \qquad (1)$$

In the above formula, required sample size is denoted as '*s*', '*N*' *is* the population size, '*P*' *is* the population proportion, '*d*' the degree of accuracy expressed as a proportion, and 'X^2' is the table value of chi-square for 1 degree of freedom at the desired confidence level (3.841). A statistical tool, G*Power 3.1.9.2, has been used to identify the sample size of our examination applying the Eq.1. Linear multiple regression test has been conducted under F tests family where number of predictors is selected as 5 in our case since the maximum predictors of the testing model is the types of exploitation. The value of α err prob was set as 0.05 and Power (1-β err prob) is selected as 0.95 in the tool. As per the result from the tool, minimum 138 valid samples was required. Figure 7 shows the graph of result for sample size of five predictors using small sample technique [31].

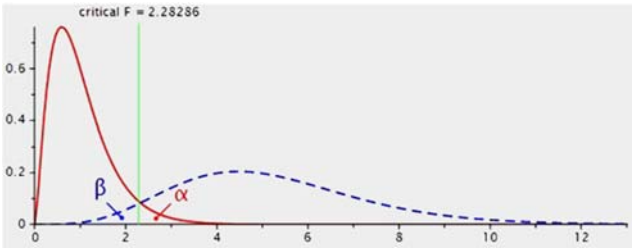

Figure 7. G*Power result for sample size of five predictors using small sample technique.

Finally, 150 Broken Authentication and Session Management vulnerable websites have been decided for our review. 267 web applications were examined to get 150 valid sample i.e. Broken Authentication and Session Management vulnerable websites. Fig. 8 represents the percentage of sample between secure and vulnerable website.

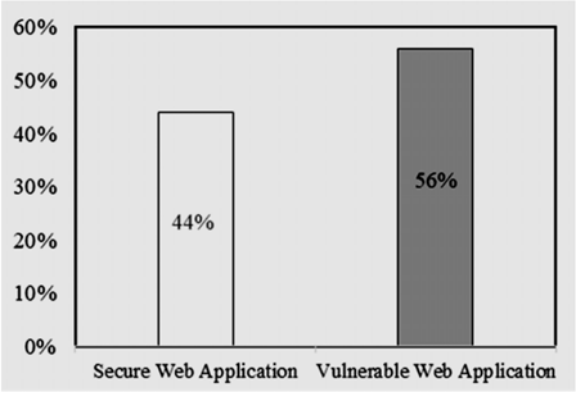

Figure 8. Percentage of sample between secure and Broken Authentication and Session Management vulnerable websites

Among the sample, 56% websites were found with Broken Authentication and Session Management vulnerability. Presence of five exploitation types of Broken Authentication and Session Management vulnerability was existed in those websites. Those vulnerable web applications have been examined for our study. Manual penetration testing method [29] using double blinded strategy [30] was chosen to collect data for this study. This dataset has been analyzed, based on Broken Authentication and Session Management exploitation type and domain based exploitation in private and public sector web applications of Bangladesh. The analysis is discussed below.

4. Analysis on Broken Authentication and Session Management exploitation type

Table I represents the frequency analysis of Broken Authentication and Session Management problem exploitation types. It is observed from the given table that 44% of Broken Authentication and Session Management vulnerable sites of our sample were exploitable through Session Misconfiguration Attack. Decoding the Decoding Inadequate Encryption was the least used technique to get the access.

TABLE I. FREQUENCY ANALYSIS OF BROKEN AUTHENTICATION AND SESSION MANAGEMENT PROBLEM EXPLOITATION TYPES

Exploitation Type	Frequency	Percent	Cumulative Percent
Session Misconfiguration Attack	66	44%	44%
Cracking/ Guessing Weak Password	33	22%	66%
Exploiting Authentication Problem	27	18%	84%
Decoding Inadequate Encryption	6	4%	88%
Exploitation on other vulnerabilities	18	12%	100%
Total	**150**	**100%**	

Cracking/ guessing weak password and exploitation through authentication problem were successful to get the access in 22% and 18% web application respectively. 12% of sample applications were exploited through other vulnerabilities. After analyzing the above statistics, it is observed that a good number of web applications in Bangladesh have been built with the problem of Session Misconfiguration attack. On the other hand, the designer and developer of the applications are not properly aware to enforce the password complexity policy in their application.

A. Analysis on Sector Wise Exploitation

This study has categorized the sector into two groups i.e. public and private. Frequency analysis of sector wise exploitation is shown in Table II.

TABLE II. FREQUENCY ANALYSIS OF SECTOR WISE EXPLOITATION

Sector	Frequency	Percent	Cumulative Percent
Public Sector	108	72%	72%
Private Sector	42	28%	100%
Total	**150**	**100%**	

In this table, it shows that Broken Authentication and Session Management vulnerability exist 72% web applications in Public sector whereas the remaining 28% of the applications were found with same vulnerabilities in private sector. It is assume from the above data that web applications of the public sector are more concerned about the features and services of their hosted application rather than concentrating on adequate security testing and security features enforcement before hosting. On the contrary, web applications of private sector are more structured than public sector web applications.

B. Analysis on Domain Wise Exploitation

Education Institution, E-Commerce, Medical Institute, Online Portal, and Government Counterpart Websites are selected domain for our study. Fig. 9 represents the percentage of domain wise exploitation. The figure specifically shows the impact on the above five domains both in public and private sector. It shows that Broken Authentication and Session Management vulnerability exists mostly on Education Institutions and Government counterparts web applications in Bangladesh with the percentage of 37% and 33% respectively.

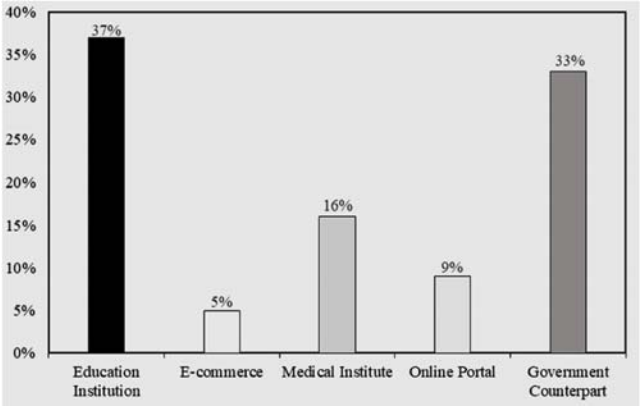

Figure 9. Percentage of the domain wise exploitation

Same weakness was found present at Online Portal and Medical Sector with the percentage of 9% and 16% correspondingly. It is also to be noticeable that only 5% of E-commerce sites got the above weakness.

C. Analysis on Exploitation Type wise Domain

Session Misconfiguration Attack, cracking Cracking/ Guessing Weak Password, Exploiting Authentication problem, Decoding Inadequate Encryption, and Exploitation

through Other Vulnerabilities have been identified to conduct the exploitation among the five domains' sample web applications. Impact of particular exploitation type on those five domains are furnish below.

i) Session Misconfiguration Attack

Table III indicates the frequency analysis of Session Misconfiguration Attack among five domains. 66 web applications in all sector has been exploited by Session Misconfiguration Attack.

TABLE III. FREQUENCY ANALYSIS OF SESSION MISCONFIGURATION ATTACK AMONG FIVE DOMAINS

Domain	Frequency	Valid Percent	Cumulative Percent
Education Institution	30	45.5%	45.5%
E-commerce	3	4.5%	50.0%
Medical Institute	9	13.6%	63.6%
Online Portal	6	9.1%	72.7%
Government Counterpart	18	27.3%	100.0
Total	**66**	**100%**	

It is visible in the table that the web applications of educational institutions are mostly affected by the Session Misconfiguration Attack with the percentage of 45.5% to compromise their admin access whereas e-commerce sites are the least affected domain with only 4.5% for the given type of exploitation. Medical Institutes, Online Portal, and Government counterpart sites were affected with Session Misconfiguration Attack with the percentage of 13.6%, 9.1%, and 27.3% respectively.

ii) Cracking/ Guessing Weak Password

Table 04 defines the frequency analysis of Cracking/ Guessing Weak Password among five domains. Total number of 33 web applications in all sectors is exploited by Cracking/ Guessing Weak Password. It is understood from the table that the most vulnerable position to be affected by the Cracking/ Guessing Weak Password with the percentage of 45.5% and 33.3% respectively among the sample is the web applications of educational institution and government counterpart domain.

TABLE IV. FREQUENCY ANALYSIS OF CRACKING/ GUESSING WEAK PASSWORD AMONG FIVE DOMAINS

Domain	Frequency	Valid Percent	Cumulative Percent
Education Institution	15	45.5%	45.5%
E-commerce	0	0%	45.5%
Medical Institute	5	15.2%	60.6%
Online Portal	2	6.1%	66.7%
Government Counterpart	11	33.3%	100%
Total	**33**	**100%**	

However, E-commerce and Online portal domain have a safe position with the above exploitation with the percentage of 0% and 6.1% consecutively. 15.2% attack has been faced with the above exploitation in medical institution's websites.

iii) Exploiting Authentication problem

The frequency analysis of exploiting authentication problem among five domains is explained in Table V. Total number of 27 web applications in all sectors is exploited by authentication problem.

TABLE V. FREQUENCY ANALYSIS OF EXPLOITING AUTHENTICATION PROBLEM AMONG FIVE DOMAINS

Domain	Frequency	Valid Percent	Cumulative Percent
Education Institution	8	29.6%	29.6%
E-commerce	1	3.7%	33.3%
Medical Institute	4	14.8%	48.1%
Online Portal	3	11.1%	59.3%
Government Counterpart	11	40.7%	100%
Total	**27**	**100%**	

Exploitation through authentication problem in web application was successful at 40.7% in government counterpart site, 29.6% in education site, 14.8% in medical institution's sites, 11.1% in online portal, and 3.7% in E-commerce site respectively.

iv) Decoding Inadequate Encryption

Frequency analysis of decoding inadequate encryption among five domains has been illustrated in Table IV.

TABLE VI: FREQUENCY ANALYSIS OF DECODING INADEQUATE ENCRYPTION AMONG FIVE DOMAINS

Domain	Frequency	Valid Percent	Cumulative Percent
Education Institution	1	16.7%	16.7%
E-commerce	1	16.7%	33.3%
Medical Institute	1	16.7%	50.0%
Online Portal	1	16.7%	66.7%
Government Counterpart	2	33.3%	100%
Total	**6**	**100%**	

Only 6 web applications were exploited through authentication problem. The table represents that government counterpart sites were compromised by decoding inadequate encryption process with the percentage of 33.3%. The remaining four domains have been affected by the same the exploitation type with a percentage of 16.7%.

v) Exploitation through Other Vulnerabilities

Table VII describes the frequency analysis of exploitation through other vulnerabilities among five domains. 18 web applications in all sectors are exploited by other vulnerabilities exploitations.

TABLE VII. FREQUENCY ANALYSIS OF EXPLOITATION THROUGH OTHER VULNERABILITIES AMONG FIVE DOMAINS

Domain	Frequency	Valid Percent	Cumulative Percent
Education Institution	2	11.1%	11.1%
E-commerce	2	11.1%	22.2%
Medical Institute	5	27.8%	50%
Online Portal	2	11.1%	61.1%
Government Counterpart	7	38.9%	100%
Total	**18**	**100%**	

It is evident from the table that the websites of government counterparts and medical institutions have been exploited through other vulnerabilities with the percentage of 38.9% and 27.8% respectively. The remaining domains i.e. education institution, e-commerce, and online portals are affected by the above exploitation with the percentage of 11.1%.

D. *Cross Tabulation of Exploitation Types vs. Domain*

Cross tabulation of Exploitation Types vs. Domain of Broken Authentication and Session Management problem exploitation types are presented in Table VIII.

TABLE VIII. CROSS TABULATION OF EXPLOITATION TYPES VS. DOMAIN

Exploitation Type / Type of Institutions	Session Misconfig. Attack	Cracking/ Guessing Weak Password	Exploiting Authentication Problem	Decoding Inadequate Encryption	Exploit other Vuln.	Total
Education Institution	30	15	8	1	2	56 (37.33%)
E-Commerce	3	0	1	1	2	7 (4.67%)
Medical Institute	9	5	4	1	5	24 (16%)
Online Portal	6	2	3	1	2	14 (9.33%)
Government Counterpart Website	18	11	11	2	7	49 (32.67%)
Total	66	33	27	6	18	150 (100%)

It is understood form the table that educational institutions and government counterpart's websites were mostly vulnerable on all five types of the above described exploitation techniques with a percentage of 37.33% and 32.67% respectively. Only 7% of those exploitations were able to affect the e-commerce sites. Web applications of Medical Institutions and Online portals were compromised by the given exploitation types with the percentage of 16% and 9.33% correspondingly.

VI. PREVENTION TECHNIQUES FROM BROKEN AUTHENTICATION AND SESSION MANAGEMENT VULNERABILITY

Basic guidelines to manage the session are provided below for preventing the given types of exploitation. It is to be noted that all the solution examples are given in PHP code.

A. Session ID Life Cycle

Session IDs can be generated in two types i.e. permissive and strict [37]. The permissive mechanism initially accepts any session ID value set by a user to create a new session. On the contrary, strict mechanism enforces the web application to accept session ID values generated by the system. If web applications do not validate and filter out the invalid session ID values before processing, it can potentially be used by an attacker to exploit other web vulnerabilities as well. The session ID must be renewed or regenerated even if the same user upgrades/degreases their user privilege level.

B. Session Reset

A defined session is build-up during an authenticated user logged in. The system preserves authentication cookies [38] for validating the user during their active session. These session cookies should be reset after that user is logged out from the system to ensure confidentiality [39]. After logging out of a user, the module should end up with the following types of code.

```
if (ISSET($_REQUEST['LOGOUT']))
{
UNSET($_SESSION[LOGOUT])
}
```

C. Session Expiration

Sessions hijacking [40] [41] is one of the types of attack by which an attacker can exploit over an active session. Therefore, it is necessary for the developer of the web application to set expiration timeouts for every session to prevent sessions hijacking attacks[40] [41]. The developer should also ensure the mechanism to keep the session active as long as the valid user remains in work. Irregular session expiration increases different types of session-based attacks as the attacker could reuse the valid session IDs and also can hijack the active associated sessions. Example of cookie expiration is shown as below:

Set-Cookie: id=; Expires=Friday,-15- July-17 18:45:00 GMT.

D. Cookies

Cookies based session ID exchange mechanism ensures numerous security properties in the form of cookie attributes which it can be used to safeguard the exchange of the session ID.

E. Session Attacks Detection

When an attacker tries to guess/ brute force a valid session ID or analyze the predictability of the session ID using statistical analysis, multiple sequential requests against the

target web application has to be launched using different session IDs from a single or multiple IP addresses. Web application's firewall has to have the capability to detect the above scenario based on the number of attempts that the system observed from different session IDs. Alert to the administrator has to be ensured and block those offending IP addresses by analyzing the payload.

F. Client-Side Defenses for Session Management

Web Application's session maintenance technique using java-script validation for client site protection is a regular way to make it safe from general users. Although it is not enough for defending any skilled intruder, but it may generate another layer of security. Attacker can bypass this client site protection using some advance tool (e.g. burp suite [42] and techniques. Therefore, the server side security needs to be address properly.

The confidential pages must use the defined system session strictly for being secure from unauthorized access. Proper session maintenance is the main key point of reducing Broken Authentication vulnerability. Insecure sessions are generally compromised by the attackers for interrupting in general session mechanism. In this case, developers need to meet some initiatives which are described below for proper management of sessions.

i) Predefined Session Period:

Session should be started with the proper validation of user's credentials i.e. username and password. The session cookie will be used to authenticate the user continually as long as the user stays active in the system. If the user found without any activity for a certain period, the session will be destroyed automatically by the system. Sample of the automatic session destroy code is given below:

```
if (isset($_SESSION['ACTIVITY']) && (time() - $_SESSION['ACTIVITY'] > 1200)) {
 // here previous request was 20 min ago
   session_unset();    //
   session_destroy();  // destroying active sessions
}
$_SESSION['ACTIVITY'] = time(); // now updating last activity
```

From the above code, it is observed that the system will destroyed the active session once it finds the user inactive for 1200 seconds.

ii) Destroy Old Sessions:

The system should not allow long duration session without proper authentication for ensuring users validity. The following types of code may help the developer to prevent session based attack.

```
if(!empty($_SESSION['deleted_time'])&&
$_SESSION['deleted_time'] < time() - 180) {
session_destroy(); // delete the old sessions
```

iii) Set Cache Limitation as Private:

The cache expiration is reset to the default value of 180 stored in the function of session.cache_expire during request startup time. Thus, the developer should ensure to call the function, session_cache_expire() for every request to define every cache limit as private. Example of sample solution code is given below:

```
/* set the cache limiter to 'private' */
session_cache_limiter('private');
$cache_limiter = session_cache_limiter();
```

G. Generating an Access Token:

Use of access token for entering into any active session is now very popular for web applications. When a user requests for creating a new session after completing the authentication process, the system generates an access token randomly to validate the user. Users have to enter the given token code with their credential to get access into their session. Since the token code are generated randomly for a limited time period, an attacker cannot hijack the user's sessions using brute-force technique even if the attacker discovers the correct user credential.

VII. CONCLUSION

Almost all web applications are maintaining the users' profile separately to ensure the quality services and communications to its user. Broken Authentication and Session Management problem are one of the major impediment to confirm the confidentiality of the web application. Therefore, the above two weaknesses have been listed as the most critical web application vulnerability since 2007 and now it is ranked as 2nd in Open Web Application Security Project (OWASP) [33]. It has been observed after conducting this examination that the existence of Broken Authentication and Session Management problem were found mostly in educational institutions and government counterpart's websites of Bangladesh with the percentage of 37.33% and 32.67% respectively. It is also revealed from this study that Session Misconfiguration attack and Cracking/ Guessing Weak Password are the most effective ways to exploit the Broken Authentication and Session Management vulnerabilities of the web application in those domains. This study has demonstrated five exploitation techniques and evaluated on websites of Bangladesh. It is our observation that the risk of discussed exploitation will be reduced once the developer follow the prevention techniques described in this paper. In future, we are intended to work on other exploitation

techniques and investigate other websites regarding Broken Authentication and Session Management vulnerability.

ACKNOWLEDGMENT

We acknowledge the authorities of the organizations who have given us the permission to conduct our examination on their websites.

REFERENCES

[1] "World Internet Users Statistics and 2017 World Population Stats", *Internetworldstats.com*,2017.[Online].Available:http://www.internetworldstats.com/stats.htm. [Accessed: 31- Oct- 2017].

[2] "Usage Statistics and Market Share of Server-side Programming Languages for Websites, November 2017", *W3techs.com*, 2017. [Online].Available:https://w3techs.com/technologies/overview/programming_language/all. [Accessed: 17- July- 2017].

[3] J. Thome, L. K. Shar, D. Bianculli, and L. Briand, "Security Slicing for Auditing Common Injection Vulnerabilities," *2017, Journal of Systems and Software.,to be published.*

[4] I. Hydara, A. B. M. Sultan, H. Zulzalil, and N. Admodisastro, " Current state of research on cross-site scripting (XSS)–A systematic literature review, " *2015 Information and Software Technology*, pp. 170-186.

[5] A. Z. M. Saleh, N. A. Rozali, A. G. Buja, K. A. Jalil, F. H. M. Ali and T. F. A. Rahman, "A Method for Web Application Vulnerabilities Detection By Using Boyer-Moore String Matching Algorithm," *2015 Procedia Computer Science, pp.*112-121.8

[6] A. Begum, M. M. Hassan, T. Bhuiyan and M. H. Sharif, "RFI and SQLi based local file inclusion vulnerabilities in web applications of Bangladesh," *2016 International Workshop on Computational Intelligence (IWCI)*, Dhaka, 2016, pp. 21-25.

[7] N. Nikiforakis, L. Invernizzi, A. Kapravelos, S. Van Acker, W. Joosen, C. Kruegel, F. Piessens & G. Vigna, "You are what you include: large-scale evaluation of remote javascript inclusions, " *2012 In Proc. of ACM conf. on Computer and communications security.*, pp. 736-747

[8] M. I. Ahmed, M. M. Hassan, T. Bhuyian, "Local File Disclosure Vulnerability: A Case Study on the Web Applications of Public Sector, 10th International Conference on Computer and Electrical Engineering (ICCEE 2017) ", Edmonton, Canada, October 2017,11-13,

[9] *Zone-h.org*, 2017. [Online]. Available: http://zone-h.org/?zh=1. [Accessed: 11- Aug- 2017].

[10] P. V. Ami and S. C. Malavy, "Top Five Dangerous Security Risks over Web Application" *2013 International Journal of Emerging Trends & Technology in Computer Science, 2(1), 41-43*.

[11] T.Petsios, V. P.Kemerlis, M.Polychronakis and A. D.Keromytis, "Dynaguard: Armoring canary-based protections against brute-force attacks, " *In Proc. 31st Annu. Computer Security Applications Conference, 2015,* pp. 351-360.

[12] N.Kaaniche and M.Laurent, "Data security and privacy preservation in cloud storage environments based on cryptographic mechanisms," *2017 Computer Communications*, pp.120-141.

[13] R. Johari and P. Sharma, "A Survey on Web Application Vulnerabilities (SQLIA, XSS) Exploitation and Security Engine for SQL Injection," *2012 International Conference on Communication Systems and Network Technologies*, 2012, Rajkot, pp. 453-458.H.

[14] A.TorkamanAtashzar, M. Bahrololum and M. H. Tadayon, "A survey on web application vulnerabilities and countermeasures,"*6th International Conf. on Computer Sciences and Convergence Information Technology (ICCIT),*Seogwipo, 2011, pp. 647-652.

[15] G.Deepa and P. S.Thilagam, "Securing web applications from injection and logic vulnerabilities: Approaches and challenges, " *2016 Information and Software Technology*, *74*, 160-180.

[16] B. Rexha, A. Halili, K. Rrmoku and D. Imeraj, "Impact of secure programming on web application vulnerabilities," *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*, Bhubaneswar, 2015, pp. 61-66.

[17] O. B. Al-Khurafi and M. A. Al-Ahmad, "Survey of Web Application Vulnerability Attacks," *4th International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, Kuala Lumpur, 2015, pp. 154-158.

[18] D. Huluka and O. Popov, "Root cause analysis of Session Management and Broken Authentication vulnerabilities," *World Congress on Internet Security (WorldCIS-2012)*, Guelph, 2012, pp. 82-86.

[19] L. Murphey, " Secure Session Management: Preventing Security Voids in Web Applications," *2005, The SANS Institute, 29.*

[20] D. Michael, K. Christos and Z. Nickolai, "Preventing Authentication & Access Control Vulnerabilities in Web Applications, "*2009, 267-282.*

[21] B. Nagpal and B. Nagpal, "Preventive measures for securing web applications using Broken Authentication and Session Management attacks: *A study.," 2014 In International Conf. on Advances in Computer Engineering and Applications (ICACEA), 2014.*

[22] D. Alam, T. Bhuiyan, M. A. Kabir and T. Farah, "SQLivulnerabilty in education sector websites of Bangladesh,"*Second International Conf. on Information Security and Cyber Forensics (InfoSec)*, Cape Town, 2015, pp. 152-157.

[23] D. Alam, M. A. Kabir, T. Bhuiyan and T. Farah, "A Case Study of SQL Injection Vulnerabilities Assessment of .bd Domain Web Applications," *2015 Fourth International Conf. on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec)*, Jakarta, 2015, pp. 73-77.

[24] T. Farah, D. Alam, M. A. Kabir and T. Bhuiyan, "SQLi Penetration Testing of Financial Web applications: Investigation of Bangladesh region," *2015 World Congress on Internet Security (WorldCIS)*, Dublin, 2015, pp. 146-151.

[25] M. M. Hassan, T. Bhuiyan and S. Biswas, "An Investigation of Educational Web Applications in Bangladesh: A Case Study on Local File Disclosure Vulnerability,"4th International Conf. on "Engineering & Technology, Computer, Basic & Applied Sciences" (ECBA- 2016), Sydney, 2016, Australia, pp-11

[26] T. Farah, M. Shojol, M. Hassan and D. Alam, "Assessment of vulnerabilities of web applications of Bangladesh: A case study of XSS & CSRF,"*Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, Konya, 2016, pp. 74-78.

[27] A. Alabrah and M. Bassiouni, "Robust and fast authentication of session cookies in collaborative and social media using position-indexed hashing,"*9th IEEE International Conf. on Collaborative Computing: Networking, Applications and Worksharing, Austin, TX, 2013*, pp. 241-249.

[28] Xiang-Wen Huang, Chin-Yun Hsieh, Cheng Hao Wu and Yu Chin Cheng, "A Token-Based User Authentication Mechanism for Data Exchange in RESTful API,"*18th International Conf. on Network-Based Information Systems, Taipei, 2015*, pp. 601-606.

[29] Y. Stefinko, A. Piskozub and R. Banakh, "Manual and automated penetration testing. Benefits and drawbacks. Modern tendency,"*13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET),* Lviv, 2016, pp. 488-491.

[30] B. H. Kang "About Effective Penetration Testing Methodology," *2008, Journal of Security Engineering*, JSE Vol. 5, No.5,

[31] V.K.Robert, W.M.Daryle, "Morgandeter Mining sample size for research activities,"*Educational and Psychological Measurement, The NEA Research Bulletin*, December, 1960, Vol. 38 , p. 99.

[32] "Top 10 2017-Top 10 - OWASP", *Owasp.org*, 2017. [Online]. Available: https://www.owasp.org/index.php/Top_10_2017-Top_10. [Accessed: 15- Sep- 2017].

[33] P. G.Kelley, S. Komanduri, M. L. Mazurek, R. Shay, T. Vidas, L. Bauer, J. Lopez, " Guess Again (and Again and Again): Measuring Password Strength by Simulating Password-Cracking Algorithms". *2012 IEEE Symposium on Security and Privacy, pp. 523-537.*

[34] S. Gold, "Cracking passwords", *Network Security*, vol. 2010, no. 8, pp. 4-7,

[35] S. Cho, Yeo and S. Kim, "Securing against brute-force attack: A hash-based RFID mutual authentication protocol using a secret value", *Computer Communications*, 2011, vol. 34, no. 3, pp. 391-397.

[36] C. Sera, S. Matlock, Y. Watashiba, K. Ichikawa and J. Haga, "Hydra: A High-throughput Virtual Screening Data Visualization and Analysis Tool", *Procedia Computer Science*, 2016, vol. 80, pp. 2312-2316.

[37] "Session Management Cheat Sheet - OWASP", *Owasp.org*, 2017. [Online].Available:https://www.owasp.org/index.php/Session_Management_Cheat_Sheet#Session_ID_Generation_and_Verification:_Permissive_and_Strict_Session_Management. [Accessed: 04- Dec- 2017].

[38] A. Liu, J. Kovacs and M. Gouda, "A secure cookie scheme", Computer Networks,2012, vol. 56, no. 6, pp. 1723-1730.

[39] Y. Wang and J. Wei, "Toward protecting control flow confidentiality in cloud-based computation", *Computers & Security*, 2012, vol. 52, pp. 106-127.

[40] D. Thomsen, "IP spoofing and session hijacking", *Network Security*,1995, vol. 1995, no. 3, pp. 6-11.

[41] A. V, P. Amritha and M. Sethumadhavan, "Sum Chain Based Approach against Session Hijacking in MPTCP", *Procedia Computer Science*,2017, vol. 115, pp. 794-803.

[42] J. Andress, "Application Security", *The Basics of Information Security*, 2011, pp. 147-166.