Prevalence of Security Risks in e-Governance Applications and its Remediation - A Case Study

B.S. Kumar[#1], V. Sridhar[#2] , K. R. Sudhindra[#3]
[#1]*Department of Electronics and Communication Engineering, PES College of Engineering, Mandya (India)*
[#2]*Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology (India*)
[#3]*Department of Electronics and Communication Engineering, B.M.S College of Engineering, Bengaluru (India*)

Abstract

Over last few years, e governance in India has made rapid progress and adopted global best practices in terms of citizen-centricity, reach, connectivity, efficiency, transparency, accountability and availability. Accordingly, as e Governance software are becoming increasingly critical, complex, and connected, the difficulty of achieving application security has increased exponentially. The security threat landscape for applications constantly changes and new type of vulnerabilities keep manifesting. In today's race to build complex and cutting-edge e Governance business solutions, web applications are being developed and deployed with minimum attention to security threats. Proven threat design techniques and known patterns are being invariably used by the attackers to exploit the commonly found security loopholes in web applications. Government can no longer afford to tolerate relatively simple and widespread security issues which could hinder delivery of services and impact the confidentiality, integrity and availability of information. In this context, an attempt is made to pool together application security issues in e Governance applications to gain a better understanding of application vulnerability landscape and its prevalence. Based on analysis, this paper outlines the vulnerability distribution pattern generally found in e Government applications and determines the prevalence and probability of different vulnerability security issues. Recommended remediation process and security controls to mitigate prevalent security issues are also discussed.

Keywords: *e Governance application security, prevalent security issues, Defect density, injection attack, Security misconfiguration, Sensitive data exposure*

I. INTRODUCTION

The Service maturity level of e Governance has seen exponential rise from being mere web-presence to interactional, transactional, transformational, and connected cum integrated services with guaranteed service levels. Security has emerged as critical compliance parameter under SLA for the successful implementation of e-Governance projects providing integrated and transaction based services. There are number of security threats at application layer which could potentially pose significant risks to an organization's information if not handled properly. As good practice, it is required to carry out

periodic analysis of these risks from e Governance perspective to determine the prevalence and its severity. This will not only save time and effort but ultimately facilitates to take an informed decision on initiating timely countermeasures. The aim is to ensure that the services does not get disrupted by prevalent known risks and at the same time cannot ignore the consequences due to less understood critical risks.

II. COMPILATION AND ANALYSIS OF APPLICATION SECURITY AUDIT REPORTS

Government departments generally get their applications audited for application security before hosting in production environment[3,4]. Most of the state data centre have made it mandatory to get 'free to host' clearance certificate from 'CERT IN' empaneled audit agencies before go live. The security audit is generally carried out in staging environment similar to production environment. The security audit is generally performed for presence of top 10 vulnerabilities/security risks as per OWASP[1]. OWASP is widely accepted as the de facto application security standard which lists the most critical web application security flaws in a document entitled "The Ten Most Critical Web Application Security Vulnerabilities". Based on independent application security audit reports, the application security vulnerabilities for compliance with OWASP top ten issues commonly found in e Governance applications are compiled and analysed for 31 tested applications. The security issues were analysed and the vulnerability distribution pattern generally found in e Governance applications is established[2]. The following sequence of activities is carried out to determine the prevalence factor associated with each identified security issues:

- Compilation of e-Governance Application Security Test reports
- Analysis of Security defects/issues: Distribution pattern and prevalence
- Determining prevalence risk factor for each security issues which can be taken as one of likelihood parameter for estimating overall risk score

III. ANALYSIS OF SECURITY ISSUES AS FOUND IN E GOVERNANCE SECURITY AUDIT REPORTS

The security issues as reported in 31 Application Test reports are analyzed for OWASP Top 10 2013 standard since the test reports issued pertains to the period 2016-17. The two issues of OWASP Top 10 2013 standard viz. Insecure direct object reference and Missing Function Level Access Control are combined and is represented as Broken Access control issue in line with OWASP Top 10 2017. Merging is done since both issues represent access control issue pertaining to data and Functionality separately. The observations resulting from analysis of Application security issues as found and reported in e Governance application audit are given below[8]:

A. Prevalence of security issues in e Governance applications

The following inferences are made based on the analysis:

- E Governance applications are affected by all types of Top ten OWASP security risks.
- The most widespread vulnerabilities are Security misconfiguration and Sensitive data exposure issues with prevalence of 94% and 78% respectively. According to OWASP Top 10 2017 Risk factor summary, these two issues are also rated as widespread.
- From the risk rating estimate [7], both Security misconfiguration and Sensi-

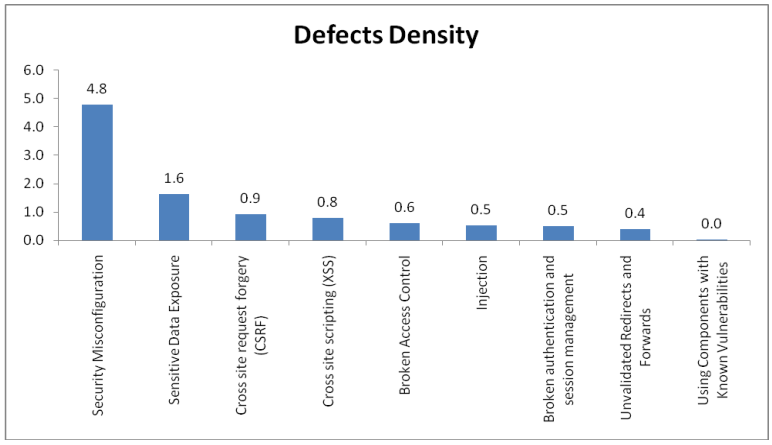

Fig. 1Details of Defects Density

tive data exposure have high impact risk value. The impact score of Security misconfiguration issue is highest with 7.2 and that of Sensitive data exposure is fourth highest with impact score of 6.

- The prevalence of injection (47%) and Broken access control (44%) issues are substantial and have considerably high impact score of 6.8 and 6.6 respectively.
- The Cross site request forgery is spread across 69% of web sites and its impact value is also moderately high with 5.2 score. However, as per WASC web application security statistics [8], it is prevalent in 5% of the applications and it is reducing since many frameworks are providing CSRF defenses.
- It is found that the following 5 security issues which are widespread are also having significant impact risk score.

table i **IMPACE SCORE FOR PREVALENCE PERCENTAGE**

Security Issue	Prevalence	Impact score
Security Misconfiguration	94%	7.2
Sensitive Data Exposure	78%	6
Broken Access Control	44%	6.6
Injection	47%	6.8
Cross site scripting (XSS)	69%	4.8

B. Security defect density analysis: Number of defects detected in each e-Governance application

The following inferences are made based on the analysis:

- On an average more than four (4.8) Security misconfiguration issues are found in each website. Since its impact is also very high, the presence of multiple defects per application could be very harmful and damaging.
- On an average, one issue pertaining to Sensitive data exposure (1.6) are present per application which also needs to be mitigated since its impact is significantly high

IV. DETAIL ANALYSIS OF TOP THREE MOST PREVALENT SECURITY ISSUES

The prevalence and defect density of security issues like Security misconfiguration, Sensitive data exposure and Injection is significant. These three security issues also have high impact risk score [7]. Further analysis of these three prevalent risks is carried out to know in detail the actual contribution and defect distribution at attack vector level.

A. Types of Security misconfigurations threat vectors and its distribution

The following inferences are made based on the analysis of distribution of attack vectors[5]:
One major finding is that HTTP security headers related vulnerabilities such as Content Security Policy, X-Content-Type-Options, X-XSS-Protection and HTTP Strict Transport Security comprises major share of 60% of detected risks under security misconfiguration vulnerability.
These HTTP response header categories require configuration settings at web server and application level to mitigate attacks and provide much needed protection. These five Security Headers are crucial to a web browser to initiate security checks. This is very handy and easy to implement and prevents attacks to a large extent.
These web security policy mechanisms determines which content to be trusted, whitelists specific origins using nonce or hash, protects against MIME sniffing, provides XSS filtering, ensures secure HTTPS connections etc. basically to prevent from code injection, Cross site scripting, etc.
Clickjacking is another major vulnerability which uses iframes with hidden malicious code in an application. It opens a door for resource sharing from other web pages but also leads to leakage of information or script injection from malicious sites. This can be prevented using Content security policy by instructing browser not to allow framing from other domains.

Another issue is autocomplete feature which allows a browser to cache whatever the user types in an input field. If an attacker gets hold of this in user's browser, credentials could be compromised and sensitive details is leaked. The autocomplete attribute should be disabled.

Security issue due to Missing HTTPOnly Attribute in session Cookie may allow adversary to steal sensitive information stored in the cookie (e.g., a session ID) and assume the identity of the user. Setting the HttpOnly flag directs compatible browsers to prevent client-side script from accessing cookies. This will mitigate the risk associated with certain client-side attacks such as Cross-site scripting.

Sensitive Cookie in HTTPS Session without `Secure' Attribute can expose cookies in plain text over an HTTP session leading to man in the middle attack. Setting the secure flag on the cookie will prevent browser submitting the cookie in any request that use an unencrypted HTTP connection.

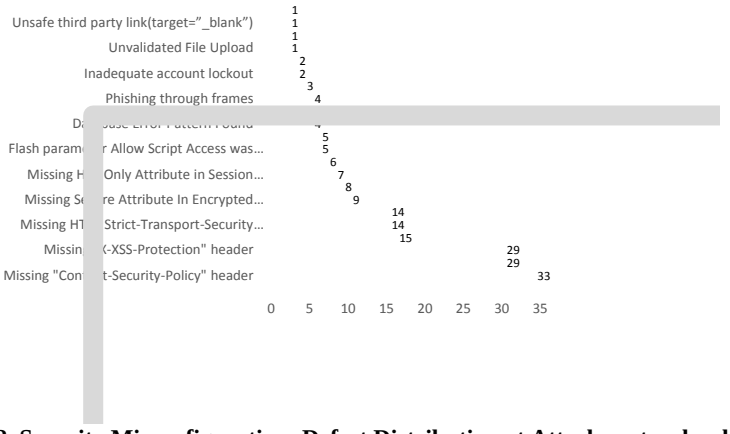

Fig. 2 Security Misconfiguration: Defect Distribution at Attack vector level

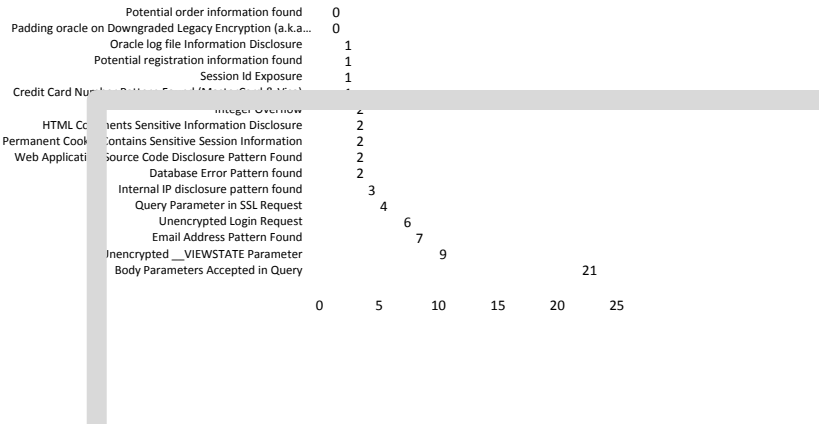

Fig. 3 Threat vector and its Defect distribution

B Types of Sensitive data exposure threat vectors and its distributions

The following inferences are made based on the analysis:

From the above findings, the risk due to "Body Parameters accepted in query" is significant contributing around 33% of the total risks of sensitive data exposure category. An attacker changes the type of request to inject some malicious content in the query string of the web application request to initiate attacks like directory traversal etc. If exploited sensitive details could be revealed to the outside world. This problem is due to developers rely on client-side validation and neglect the server-side validation all together.

In ASP.NET View-state parameter is another vulnerability issue which is generally used to save the state of the web page and lessen the load on server. If the configuration is not set properly to encrypt the Viewstate parameters in web.config file, an attacker could exploit it.

Sometimes developers leave sensitive details like email address, IP, phone numbers etc. in the HTML pages which is visible to the outside world.

C. Types of Injection threat vectors and its distributions

Injection attacks refer to a broad class of attack vectors with SQL injection (SQLi) being the most prevalent. Other types of injection attacks which are commonly exploited are OS command injection, Carriage Return and Line Feed injection, LDAP injection, etc. SQL Injection involves tampering of user provided data with SQL commands that can read or modify data from a database. Potential impact could lead to Information disclosure, Data loss, Data theft, Loss of data integrity, Denial of service and Full system compromise. This can be prevented by using Prepared statement with parameterized queries, whitelisting input validation or escaping all user supplied data.

V. DETERMINING PREVALENCE RISK FACTOR RATING

Based on the analysis of distribution of security issues across different e Governance applications, the prevalence value for all issues as listed in OWASP Top 10 2013 is derived and is given below in table 1. Prevalence risk factor can be taken as one of the factor under likelihood to estimate overall risk value [7] for security issues.

Table 2
PREVALENCE VALUE FOR VARIOUS SECURITY ISSUES

Security Issues	Prevalence value
Injection	5
Broken Authentication	5
Sensitive Data Exposure	8
Broken Access Control	5
Security Misconfiguration	9
Cross Site Scripting (XSS)	4
Using Components with Known Vulnerabilities	1
Cross-Site Request Forgery	7
Unvalidated Redirects and Forwards	4

VI. CONCLUSION

The security threat landscape for applications constantly changes and known techniques are being frequently adopted by attackers to probe the e-Governance sites. To ensure that Government applications are hack resistant from widespread security threats, it is required to monitor and identify widespread security vulnerabilities with similar exercise of analysis on regular basis by reputed National agencies. The Prevalence risk factor rating determined periodically will facilitate organizations and developers to mitigate the prevalent risk issues to acceptable level in a cost-effective manner. Mitigation plan should ensure identifying and initiating remediation process and appropriate security controls spanning the life cycle.These will not only produce secure e-Governance web applications but it also enables the stakeholders to dynamically develop preventive mitigation strategies for prevalent and potential vulnerabilities.

REFERENCES

[1] OWASP Top 10 – The Ten most critical Application Security issues, 2013 and 2017

[2] OWASP Risk Rating Methodology, https://www.owasp.org/index.php/

[3] CWE. Common Weakness Enumeration CWE/SANS Top 25 retreieved from https://CWE.mitre.org/

[4] Robert A. Mar Sean Barnum "A Status Update: The Common Weaknesses Enumeration"retrieved from https://www.researchgate.net/publication/234812149

[5] WASC (2010) WASC Threat Classification, version 2.00 retrieved from http://projects.webappsec.org/f/WASC-TC-v2_0.pdf

[6] WASC Web application security statistics 2008, http://www.webappsec.org/.

[7] Karishma Pooj , Sonali Patil. "Understanding File Upload Security for Web Applications" International Journal of Engineering Trends and Technology (IJETT) – Volume-42 Number-7 - December 2016

[8] Deven C. Pandya, Dr. Narendra J. Patel. Study and analysis of E-Governance Information Security (InfoSec) in Indian Context, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 1, Ver. IV (Jan.-Feb. 2017), PP 04-07