



Cryptographic Issues and Vulnerabilities in Web Applications

H M P Kavinda Ranjan Kumara Herath

(Reg-No: MS20911058)

M.Sc. in IT

Specialized in Cyber Security

Supervisor: Dr. Lakmal / Ms. Chethana

December 2021

Secure Your World with Minimal Risk

Department of Information Technology

Faculty of Graduate & Reach

Sri Lanka Institute Of Information Technology

Dedicated
To
My Beloved Parents
And
Son Jalan RukmaHerath

DECLARATION

Here I declare my thesis of the research project work titled **“CRYPTOGRAPHIC ISSUES AND VULNERABILITIES IN WEB APPLICATIONS”** which was prepared submitted to the **Faculty Of Graduate & Research, Sri Lanka Institute Of Information Technology**, in the essential part of the requirements for the award of the **Master of Science in Information Technology Specializing in Cyber Security**, is a bonafide report of the work carried out me. The material contained in this Report has not been submitted to any University or Institution for the award of any degree.

H M P Kavinda Ranjan Kumara Herath Reg-No: MS20911058

.....

CERTIFICATE OF APPROVAL

The undersigned certify that the thesis entitled submitted to the Faculty Of Graduate & Research in partial fulfillment of requirement for the Master of Science in Information Technology Specializing in Cyber Security. The project was carried out under special supervision and within the time frame prescribed by the syllabus. As per the individual evaluation of the students that we pursued their dedication, hardworking, bonafide and ready to undertake any challenges appropriate commercial and industrial work related to their field of study and hence we recommend the award of Master of Science in Information Technology Specializing in Cyber Security.

1.

(Project Supervisor)

2.

(External Examiner)

3.

(Head)

Master of Science in Information Technology Specializing in Cyber Security

ACKNOWLEDGEMENT

The theme which was selected for my final project is quite challenging to narrate due to the subject matters quite new for the industry. I highly Appreciate Dr, Lakmal for being given his full support to select such a challenging topic to explore means of cyber security discipline indeed. Writing in this dissertation quit challenge due to the lack of previous attempts and new to the field of research. I highly appreciate each individual including with my colleges for your guideline which was given to me for accomplishment of the goal without further dragging.

This dissertation looks like a small portion of the single booklet however it could be contributed by several faculty members even though some of them are supported invisibly. My special appreciation goes to my mentor Dr. Lakmal and Ms. Chethana. With their contribution might not be carried out such work within the correct time. I will take this opportunity to express my gratitude to every individual who has given their fullest support for my achievement of this review report throughout our Master of Cyber Security program.

Finally, I would like to express my gratitude with a deep appreciation for my family members for their unforgettable involvement and perpetual encouragement throughout the Master program.

H M P Kavinda Ranjan Kumara Herath Reg-No: MS20911058

ABSTRACT:

Web application security is the most controversial and crucial factor to be concentrated on considering the security aspect of cyberspace. Cryptography takes critical parts of security by implementing encryption and decryption phenomena on data at rest, in moving, and in use to be protected the security breaches. Cryptographic concepts had developed over the last few decades as a result of well-known series of mathematical and logical functions. Weakness of poor programming techniques or leakiness of traditional software development life cycles is a crucial element of the security vulnerabilities that can be a huge impact on several web applications which are currently in existence.

The cryptographic vulnerabilities of the web application would depend on several factors such as lack of knowledge on particular subject matters of cryptography, least privilege and contribution of security techniques while coding, unable to proceed with proper standardized vulnerability assessment criteria, the improper adaptation of cryptographic concepts, unable to intended with high secure framework like DevSecOps, depend on the procedures rather than empirical approaches, etc. Sophisticated tools and techniques are necessary factors of driving through the rectification and mitigation of the security vulnerabilities that exist in the web applications whereas implementation process, testing and monitoring of the System Development Life Cycle. This dissertation emphasized indeed a further illustration of cryptographic vulnerability assessment in several specimens collected from different domains from enterprise web applications and related APIs (Application Protocol Interface) currently established. The tools are the critical elements used to evaluate errors on the codes whereas statistical or dynamic analysis. Static tools are given in high percentage of accuracy of the results whereas automated tools are well suited for mega scripting projects such as millions of code evaluated for errors. Java-based code scripting has been dominated still among the huge percentage of the web sources. Python will be established gradually due to the high inbuilt security system on it. Java and Python are the programming languages still being dominated of existence to discuss in the cryptographic vulnerabilities on the process of web application developments. The ultimate goal of this dissertation could be retain valuable sources of documents enriched with sophisticated technics to be used a reference guide for the developers and the security engineers to fulfilled their gaps between code and security requirements.

Keywords: Application Protocol Interface, Cryptographic Vulnerability, DevSecOps, Dynamic Analysis, Statistical Analysis System Development Life Cycle

TABLE OF CONTENTS

Cover Page	i
Dedication	ii
Declaration	iii
Certificate of approval	iv
Acknowledgement	v
Abstract	vi
Table of Contents	vii
List of Figures	viii
List of Tables	xii
Chapter 1: Introduction	1
1.1 Background Introduction	3
1.2 Other related topics or terms	4
1.2.1 Cryptography	4
1.2.2 New Definition of Cryptography	5
1.2.3 Encryption technology	5
1.2.4 Algorithm	7
1.2.5 Cryptographic Vulnerabilities	8
1.2.6 Cryptographic Domain	8
1.3 Motivation	9
1.4 Problem Definition and Research Questions	9
1.4.1 Problem Definition	9
1.4.2 Research Questions	10
1.5 Goals and Objectives	11
1.6 Scope and Applications	11
Chapter 2: Literature Review	12
2.1 Abstract	12
2.2 Introduction	13
2.3 Objectives	14
2.4 Method	14
2.5 Standard Framework	15
2.5.1 Investigation Frameworks	15
2.5.2 Bayesian Framework	16
2.5.3 Monotone Framework	16
2.5.4 Javascript Application Frameworks	16
2.6 Elements Related to Cryptographic in Web Applications	17
2.6.1 W3c Web Cryptography APIs	17
2.6.2 TLS/SSL Security	18

2.7	Tools & Techniques	18
2.7.1	CrySL	19
2.7.2	CryptoGuard	21
2.7.3	FixDroid	21
2.7.4	CogniCrypt	23
2.7.5	CryptoLint	24
2.7.6	Slicing Techniques	24
2.8	Protocols Related To Web Application	25
2.9	Security Breaches	27
2.9.1	Internet Protocol Security	27
2.9.2	IPSec Protocol	27
2.9.3	IPSec VPN	28
2.9.4	IPSec Protocol	28
2.10	Security Breaches	29
2.11	Penetration Testing for Cryptographic Issues and Vulnerabilities	29
2.12	Introduction to Bug Bounty	30
2.13	Standard to follow	31
2.14	Conclusion	31
Chapter 3:	Requirement Analysis	33
3.1	Activates of requirement Analysis	33
3.2	Requirement Analysis Techniques	34
3.2.1	Software and Hardware Requirement Specifications	34
3.2.2	Technical Feasibility	34
3.2.3	CryptoGuard	35
3.3	Further More on Feasibility study	35
3.4	Required Facilities	36
3.4.1	Budget	36
3.5	Known about the attack vectors and the countermeasures	37
Chapter 4:	System Design and Architecture	39
4.1	Type of study	39
4.2	Data collection method and research instruments	39
4.3	Sampling Technique	40
4.3.1	Intended data analysis techniques	40
Chapter 5:	Methodology	42
5.1	Penetration Testing	42
5.1.1	Penetration Testing Fundermentals	43
5.2	Penetration Testing Stages	44
5.2.1	Pre-Engagement Phase	45
5.2.2	Information-Gathering Phase	45

- 5.2.3 Vulnerability Analysis Phase45
- 5.2.4 Reporting Phase45
- 5.3 Vulnerability Assessment And Penetration Testing Tool47
 - 5.3.1 Testing Tools & Common Vulnerabilities47
 - 5.3.2 Cryptographic Vulnerability Testing Tools48
- 5.4 Bug Bounty Practices48
 - 5.4.1 Bug Bounty Process50
 - 5.4.2 Rewards51
 - 5.4.3 Kali Linux for bug bounty52
- 5.5 Common Cryptographic Vulnerabilities53
 - 5.5.1 Brute Force Attack53
- Chapter 6: Implementation Details50
 - 6.1 Finding Vulnerabilities from CryptoGuard55
 - 6.2 Java Build Tools56
 - 6.3 Scanning Enhancement57
 - 6.4 Java File54
 - 6.5 Java Class file.....60
 - 6.6 Schema Design61
 - 6.7 Python Cryptographic Vulnerabilities62
 - 6.8 Vulnerability Detection Methods63
 - 6.8.1 Cryptographic Vulnerability63
 - 6.8.2 Detection Methods63
- Chapter 7: Result and Analysis66
 - 7.1 Insufficiently protected credentials66
 - 7.2 Recommendations68
 - 7.3 Weak Keys vulnerability69
 - 7.4 FPGA70
 - 7.4.1 Field-Programmable Gate Array71
 - 7.5 Cryptographic common vulnerabilities72
 - 7.6 False positives74
- Chapter 8: Conclusion and Future Work76
 - 8.1 Conclusion76
 - 8.2 Limitation77
 - 8.3 Future enhancement77
 - 8.4 Future Implementation Suggestion79
- References81
- Appendix85

LIST OF FIGURES

Figure 1.1 Security and Vulnerability Risk3

Figure 1.2 Proposed security architecture of Security Operation Center.....3

Figure 2.1 Public key signature17

Figure 2.2 Vulnerability in coding20

Figure 2.3 (a) encryption of AES (b) Graph of data dependency for key Bytes20

Figure 2.4 the cryptographic key with in a hard-coded string22

Figure 2.5 insecure code detection from FixDroid22

Figure 2.6 suggestions to figure out the issue22

Figure 2.7 HTTPS upgrade as per the suggestion23

Figure 2.8 Architecture of FixDroid23

Figure 2.9 the process of code generation of CogniCript24

Figure 2.10 SSL and TLS development process26

Figure 3.1 security in depth for general for whole criteria37

Figure 5.1 Vulnerability Assessment and Penetration Testing Life cycle46

Figure 5.2 Penetration testing lab requirements46

Figure 5.3 Bug Bounty Variants50

Figure 5.4 Bug Bounty Cycle51

Figure 5.5 CFRF attack54

Figure 6.1 Cryptographic misuse detection from CriptoGusrd55

Figure 6.2 the flow of Scanning56

Figure 6.3 Expansion of source scanning Limited usage of Soot57

Figure 6.4 Methodologies of Java preparation to Soot58

Figure 6.5 Limited usage of Soot58

Figure 6.6 Method of Java Class Loading55

Figure 6.7 Retrieving qualified path for Java file59

Figure 6.8 Create soot environment in Java Class File60

Figure 6.9 From the Java class file retrieving fully qualified path61

Figure 6.10 Vulnerabilities and packages over years62

Figure 6.11 Vulnerabilities over Years in high, medium and low63

Figure 6.12 Type of Vulnerabilities according to Industrial Control System64

Figure 7.1 Users password change66

Figure 7.2 Read the value of password in config.properties67

Figure 7.3 Read the value of password in registry key67

Figure 7.4 verify the password from compress version67

Figure 7.5 RSA Key vulnerability69

Figure 7.6 Secure system of FPGA – model one71

Figure 7.7 Secure system of FPGA – model two71

Figure 7.8 Secure system of FPGA – model three72

Figure 7.9 Secure & Insecure source72

Figure 7.10 API URL73

Figure 7.11 Predictable number generation73

Figure 7.12 Insecure algorithms73

Figure 7.13 Insecure cipher algorithms73

Figure 7.14 Source of false positives74

Figure 7.15 Programming idioms74

Figure 8.1 most wanted languages78

Figure 8.2 Web Frameworks78

Figure 8.3 Proposed KSOC FPGA Security Cloud based solution80

LIST OF TABLES

Table 1.1 encryption algorithms6

Table 1.2 Algorithm Comparison7

Table 2.1 High, Medium, and Low level of risk of cryptographic vulnerabilities25

Table 2.2 different approaches of network security protocols26

Table 2.3 Secure and insecure threat models29

Table 2.4 ITIL, COBIT and ISO/IEC2700231

Table 3.1 Required Budget for the entire project36

Table 5.1 Testing Tools and Common Vulnerabilities47

Table 5.2 Cryptographic Vulnerability Testing Tools48

Table 6.1 Common vulnerabilities in Python61

Table 6.2 Vulnerabilities, attack type and cryptographic properties.....64

Table 6.3 CWE report on cryptographic vulnerabilities test cases64

Table 6.4 CRYPTOAPI-BENCH Comparison65

Table 7.1 NIST Cryptographic Algorithm recommendation68

Table 7.2 Strong and weak keys in some countries69

Table 7.3 50 most repeated SSH Keys70

Table 7.4 Diplomat Great Accuracy Value75