



**A FRAMEWORK FOR CLASSIFICATION  
SOFTWARE SECURITY USING COMMON  
VULNERABILITIES AND EXPOSURES**

**NOR HAFEIZAH BINTI HASSAN**

**DOCTOR OF PHILOSOPHY**

**2018**



**Faculty of Information and Communication Technology**

**A FRAMEWORK FOR CLASSIFICATION SOFTWARE SECURITY  
USING COMMON VULNERABILITIES AND EXPOSURES**

**Nor Hafeizah binti Hassan**

**Doctor of Philosophy**

**2018**

**A FRAMEWORK FOR CLASSIFICATION SOFTWARE SECURITY USING  
COMMON VULNERABILITIES AND EXPOSURES**

**NOR HAFEIZAH BINTI HASSAN**

**A thesis submitted  
in fulfillment of the requirements for the degree of  
Doctor of Philosophy**

**Faculty of Information and Communication Technology**

**UNIVERSITI TEKNIKAL MALAYSIA MELAKA**

**2018**

## DECLARATION

I declare that this thesis entitle “A Framework For Classification Software Security Using Common Vulnerabilities And Exposures ” is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : .....

Name : .....

Date : .....

## **APPROVAL**

I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of Doctor of Philosophy.

Signature : .....

Supervisor Name : .....

Date : .....

## **DEDICATION**

Hassan Aman

and

Maimun Yusof

## ABSTRACT

The main research aim is to investigate what information is necessary to make a formal vulnerability pattern representation. This is done through the usage of formal Backus-Naur-Form syntax for the execution and presented with newly created vulnerability flow diagram. Some future works were also proposed to further enhance the elements in the secured software process framework. This thesis focuses on the research and development of the design, formalization and translation of the vulnerability classification pattern through a framework using common vulnerabilities and exposures data. To achieve this aim, the following work was carried out. First step is to create and conceptualized necessary meta-process. Second step is to specify the relationship between the classifiers and vulnerability classification patterns. This inclusive of the investigation of vulnerability classification objectives, processes, classifiers and focus domains among prominent framework. Final step is to construct the framework by establishing the formal presentation of the vulnerability classification algorithm. The validation process was conducted empirically using statistical method to assess the accuracy and consistency by using the precision and recall rate of the algorithm on five data sets each with 500 samples. The findings show a significant result with precision's error rate or  $p$  value is between 0.01 and 0.02 with error rate for recall's error rate is between 0.02 and 0.04. Another validation was conducted to verify the correctness of the classification by using expert opinions, and the results showed that the ambiguity of several cases were subdue. Formal-based classification framework with notation may increase accuracy and visualization compared with hierarchy-tree only, but the conclusion remains tentative because of methodological limitation in the studies.



## ABSTRAK

Tujuan utama penyelidikan ini adalah untuk menyiasat perincian yang diperlukan untuk membuat perwakilan formal corak kerentanan. Ini dilakukan melalui penggunaan sintaks Backus-Naur-Form untuk pelaksanaan dan diwasilahkan dengan pengenalan kepada rajah aliran rentan yang baru. Beberapa titipan kerja untuk masa depan juga dicadangkan untuk menambahbaik elemen-elemen dalam rangka kerja perisian jamin-selamat. Tesis ini memberi tumpuan kepada penyelidikan dan pembangunan reka bentuk, formalisasi dan terjemahan corak klasifikasi kerentanan melalui rangka kerja menggunakan data kerentanan umum dan kededahan lazim. Untuk mencapai matlamat ini, kerja-kerja berikut telah dijalankan. Langkah pertama adalah mewujudkan dan memberi konsep kepada meta-proses. Langkah kedua ialah menentukan hubungan antara pengelas dan corak pengelas kerentanan. Ini termasuklah kenalpasti objektif klasifikasi kerentanan, proses, klasifikasi dan fokus domain di antara rangka kerja-rangka kerja yang ada. Langkah terakhir ialah membina rangka kerja dengan menghasilkan paparan algoritma klasifikasi kerentanan formal. Proses pengesahan dijalankan secara empirikal menggunakan kaedah statistik untuk menilai ketepatan dan ketekalan algoritma berdasarkan pada kadar ketepatan dan panggil-balik ke atas lima set data, setiap satunya dengan 500 sampel. Hasil penemuan menunjukkan dapatan yang signifikan dengan kadar ralat ketepatan atau nilai  $p$  adalah antara 0.01 dan 0.02 dan kadar ralat untuk kadar ralat panggil-balik adalah antara 0.02 dan 0.04. Satu lagi pengesahan telah dijalankan untuk menentusahkan jenis klasifikasi dengan menggunakan pendapat pakar, dan hasilnya menunjukkan bahawa ketidaktentuan beberapa kes telah dikurangkan. Justeru, rangka klasifikasi berasaskan formal dengan notasi boleh meningkatkan ketepatan dan visualisasi berbanding dengan secara hiraki sahaja, tetapi kesimpulannya adalah tentatif kerana batasan metodologi dalam kajian.

## ACKNOWLEDGEMENTS

Alhamdulillah. I would like to gratefully acknowledge the Ministry of Higher Education Malaysia for sponsoring this research. Also, to the Center for Graduate Studies for their commitment in assisting this work.

The success of this research was made possible with the encouragement, motivation, assistance by many individuals. First and foremost is Professor Datuk TS. Dr. Shahrin Sahib, who made my dream to pursue and accomplish the work on Secured Software field comes true with his continual study guidance despite his challenging busy schedule. My thanks to Professor Dr. Burairah Hussin for his determination in encouraging the publication and my deepest gratitude to colleagues of Faculty of Information Communication and Technology.

This thesis accomplishment was also endured by my dearest, Azman Awang Teh, Irfan, Anis, Alia, Imran, Amni and Ilyas.

Such a challenging writing companion, yet enjoyable!

## TABLE OF CONTENTS

	<b>PAGE</b>
<b>DECLARATION</b>	
<b>APPROVAL</b>	
<b>DEDICATION</b>	
<b>ABSTRACT</b>	<b>i</b>
<b>ABSTRAK</b>	<b>ii</b>
<b>ACKNOWLEDGEMENTS</b>	<b>iii</b>
<b>TABLE OF CONTENTS</b>	<b>iv</b>
<b>LIST OF TABLES</b>	<b>viii</b>
<b>LIST OF FIGURES</b>	<b>xi</b>
<b>LIST OF APPENDICES</b>	<b>xiii</b>
<b>LIST OF ABBREVIATIONS</b>	<b>xiv</b>
<b>LIST OF PUBLICATIONS</b>	<b>xv</b>
<b>CHAPTER</b>	
<b>1. INTRODUCTION</b>	<b>1</b>
1.1 Background	1
1.2 Problem Statement	2
1.3 Research Questions	3
1.4 Research Aim and Objectives	4
1.5 Research Scope	6
1.6 Significance of the Study	6
1.7 Organization of the Study	7
1.8 Summary	7
<b>2. LITERATURE REVIEW</b>	<b>8</b>
2.1 Introduction	8
2.2 An Overview	8
2.2.1 The Gap	10
2.2.2 Vulnerability Classification	16
2.2.3 Vulnerability Classification: The Generic Process	20
2.2.4 Vulnerability Classification: The Classifiers	21
2.2.5 Vulnerability Classification: The Vulnerabilities Patterns	22
2.3 Secured Software Framework	25
2.3.1 Measurement	29
2.4 Theoretical background	29
2.4.1 Automata theory	30
2.4.2 Formal language theory	30
2.4.3 Parsing	31

2.4.4	Model abstraction analysis	31
2.5	Conclusion	32
<b>3.</b>	<b>RESEARCH METHODOLOGY</b>	<b>33</b>
3.1	Introduction	33
3.2	Research Strategy	34
3.2.1	The Qualitative Approach	34
3.2.2	The Quantitative Approach	36
3.2.3	The Selected Research Approach: Hybrid Approach	37
3.3	Research Framework	38
3.3.1	Meta-Process Model	41
3.3.2	Vulnerability Classification Pattern	42
3.3.3	Accepting Affirmation	43
3.4	Data collection	44
3.5	Selection of data	46
3.6	Criteria for The Data	48
3.7	Data Sets	49
3.7.1	Data Selection	50
3.7.2	Obtaining the data	50
3.7.3	Choosing the data	51
3.7.4	Data analysis	53
3.8	Experts View	54
3.9	Threats Validity	55
3.10	Summary	55
<b>4.</b>	<b>META-PROCESS FOR VULNERABILITY CLASSIFICATION</b>	<b>56</b>
4.1	Introduction	56
4.2	Investigation of Meta-process for Vulnerability Classification	57
4.2.1	Selection of a reference model	57
4.2.2	Analyzing the process, activities and output	58
4.2.3	Conceptualization of Vulnerability Classification Meta-process	60
4.2.4	New Vulnerability Classification Framework (VulClaF)	61
4.3	Establishing the Vulnerability Classifiers	63
4.3.1	Identification of vulnerability classification reference model	63
4.3.2	The Label of Characterization as the Classifiers	67
4.3.3	Establishing the relationship using the proposed labeling characterization	69
4.4	Data Mapping and Representation	70
4.4.1	Domain Specific Marker (DSMarker)	70
4.4.2	Domain Specific Wordlist (DSWordlist)	72
4.4.3	Domain Specific Schema Grammar (DSSchema)	74
4.5	The generative grammar for VulClaP	75
4.6	Context-Free Grammar for Domain Specific Schema (DSSchema)	78

4.7	Relationship Notation	81
4.8	Summary	86
<b>5.</b>	<b>THE VULNERABILITY CLASSIFICATION PATTERN</b>	<b>87</b>
5.1	Introduction	87
5.2	The Execution of Schema	88
5.2.1	The Preprocessing	88
5.2.2	The Syntax-Semantic Analysis	90
5.2.3	The Pattern Matching	91
5.3	The VulClaP Algorithm	92
5.4	Summary	95
<b>6.</b>	<b>RESULT AND DISCUSSION</b>	<b>96</b>
6.1	Introduction	96
6.2	New Vulnerability Classification Framework	97
6.2.1	Identification	99
6.2.2	Analysis	99
6.2.3	Confirmation	100
6.2.4	Elimination	101
6.3	Execution on Datasets	102
6.3.1	CVE Threat Type vs. STRIDE	102
6.3.2	The Mix-based Data sets	103
6.3.3	The Vendor-based Datasets	108
6.4	Analysis on Experimental Result	111
6.5	Validation by Experts	113
6.5.1	Section A: Demographic Information	114
6.5.2	Section B: Understanding the STRIDE model	115
6.5.3	Overview of the Questions	116
6.5.4	Rational of Questions	124
6.5.5	Expertise feedback	130
6.6	Analysis on Validation Result	135
6.7	Final Result	145
6.8	Conclusion	148
<b>7.</b>	<b>CONCLUSION AND FUTURE WORK</b>	<b>149</b>
7.1	Introduction	149
7.2	Summary of Contribution	149
7.2.1	Research contributions	150
7.3	Future Work	151

<b>REFERENCES</b>	<b>153</b>
<b>APPENDICES</b>	<b>163</b>

## LIST OF TABLES

<b>TABLE</b>	<b>TITLE</b>	<b>PAGE</b>
1.1	Summary of research problems	2
1.2	Summary of research questions	4
1.3	Summary of research objectives	5
2.1	Vulnerabilities classification based on respective research goal	18
2.2	Vulnerabilities Classification based on Respective Research Goal (..continued)	19
2.3	Vulnerabilities classification based on processes	20
2.4	The stages focused in software security process framework	27
3.1	Data characteristics	46
3.2	Criteria for data selection	48
3.3	Selected Data Sets	49
3.4	Data characteristics	50
3.5	Data characteristics	52
3.6	The sample of CVE data	53
4.1	The meta-classification of a vulnerability modeling system	60
4.2	An analysis on classification model	65
4.3	The simplified reference model for vulnerability classification	66
4.4	The new label of characterization and relationship for secured software framework	68
4.5	Example of items for classifiers	72
4.6	Words, Definition and Classifiers	73
4.7	Domain Specific Word list with Semantic	74
4.8	Definition of source and target	81

4.9	A sample of semantic understanding of relationship notation for vulnerability flow	83
4.10	A sample of semantic understanding of relationship notation for vulnerability flow (cont.)	84
6.1	Identification - Sinopsis of Process, Activity and Output	99
6.2	Analysis - Sinopsis of Process, Activity and Output	100
6.3	Confirmation - Sinopsis of Process, Activity and Output	101
6.4	Elimination - Sinopsis of Process, Activity and Output	101
6.5	Mapping CVE Type to STRIDE Model	103
6.6	Number of Classification	104
6.7	DS1 - Precision and Recall rate	105
6.8	DS2 - Precision and Recall rate	107
6.9	DS3 - Precision and Recall rate	107
6.10	Basic criteria of data selection	108
6.11	Validation sets and Details	109
6.12	Results: Domain-based Dataset DS4	110
6.13	Results: Domain-based Dataset DS5	110
6.14	Summary of standard deviation and p value for DS1, DS2 and DS3	111
6.15	Summary of standard deviation and p value for DS4 and DS5	112
6.16	Expert Selection	113
6.17	Overview of STRIDE	115
6.18	Definition of STRIDE ( <i>source: <a href="https://msdn.microsoft.com">https://msdn.microsoft.com</a></i> )	117
6.19	Part II Section A	119
6.20	Part II Section B	120
6.21	Part II Section C	121
6.22	Part II Section D	122



6.23	Section B: Question 1	124
6.24	Section B: Question 2	125
6.25	Section B: Question 3	126
6.26	Section B: Question 4,5 and 6	127
6.27	Section B: Question 7 and 8	128
6.28	Section B: Question 9,10 and 11	129
6.29	Section B: Question 12,13 and 14	130
6.30	Summary of the feedback	132
6.31	Feedback vs Result	135
6.32	Feedback-Part II Section A	140
6.33	Feedback - Part II Section B	141
6.34	Feedback - Part II Section C	142
6.35	Feedback - Part II Section D	142
6.36	Feedback - Part II Section E Question 1	143
A.1	Relationship	164
A.2	Summary of The Entity	166
A.3	Semantic: Sample of Results	167
A.4	Sample of the semantic occurrences	168

## LIST OF FIGURES

FIGURE	TITLE	PAGE
2.1	Cyber and Non-cyber Incidents reported by Federal Agencies in fiscal year 2006-2015 (Source: GAO analysis of US-CERT)	9
2.2	The research on origin of errors introduced	11
2.3	Error, Fault and Failures	12
2.4	Relationship of the terms-Error, Fault, Failures and Vulnerabilities	13
2.5	Another Definition of Vulnerability by CVE, 2015	14
2.6	Vulnerabilities - negative consequences of faults - from one or more errors	15
3.1	Research Approach Used: Hybrid Approach	38
3.2	Research Framework	40
3.3	Data for Preliminary, Training and Testing	45
4.1	Structure for Chapter 4	56
4.2	The Formulation of Vulnerability Classification Meta-process	61
4.3	The New Vulnerability Classification Framework	62
4.4	The Summary of the labels	66
4.5	The proposed labels and relationship	69
4.6	The markers in incidents structure	71
4.7	Tree for Context-free grammar DSSchema	79
4.8	Example of Relationship Notation using Vulnerability Flow Diagram	85
5.1	Structure for Chapter 5	87
5.2	Flowchart for producing the words	91
6.1	Structure for Chapter 6	97
6.2	The Enhance Vulnerability Classification Framework(VulCaF)	98

6.3	CVE Vulnerability Type - An Example from 1999-2008 (source: CVE website)	102
6.4	Sample of question for categorization of an incident	116
6.5	Part II Section E Question 1	122
6.6	Part II Section E Question 2	123
6.7	Part II Section E Question 3 and 4	123
6.8	The New Enhance Vulnerability Classification Framework(VulCaF)	147
A.1	The Sample of CVE data	163
A.2	The Vulnerability Point	165
A.3	The Vulnerability Exploit	165
A.4	The Vulnerability Impact	166
A.5	Sample of Semantic Network Diagram	169

## LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	VulnerabilityPatterns	163
B	Sample Listings	170
C	Sample Dataset	182
D	Validation Expert 1	183
E	Validation Expert 2	191
F	Validation Expert 3	199
G	Validation Expert 4	205
H	Validation Expert 5	211
I	Validation Expert 6	217
J	Validation Expert 7	222
K	Validation Expert 8	227

## LIST OF ABBREVIATIONS

BNF	-	Backus-Naur-Form
CFG	-	Context-free grammar
CVE	-	Common Vulnerability and Exposures
DSMarker	-	Domain Specific Marker
DSSchema	-	Domain Specific Schema
DSWordlist	-	Domain Specific Wordlist
NVD	-	National Vulnerability Database
OSVDB	-	Open Source Vulnerability Database
OWASP	-	Open Web Application Security Project
PA	-	Protection Analysis
RISOS	-	Research Into Secure Operating Systems
VulClaF	-	Vulnerability Classification Framework
VulClaP	-	Vulnerability Classification Pattern

## LIST OF PUBLICATIONS

1. Nor Hafeizah Hassan, Nazrulazhar Bahaman, Burairah Hussin, Shahrin Sahib, (2018), Enhancing the Secured Software Framework using Vulnerability Patterns and Flow Diagrams, *International Journal of Advanced Computer Science and Applications (IJACSA)*.
2. Nor Hafeizah Hassan, Shahrin Sahib, (2015), Assessing The Mapping Process Using Evaluation Criteria to Validate Case Study Results, *Recent Advances in Computer Sciences, WSEAS*.
3. Hassan, N.H. and Selamat,S.R. and Sahib,S., (2014), Establishing the Relationship in Vulnerability Classification for a Secure Software Testing, *Atlantis-Press*.
4. Hassan, N.H. and Selamat, S.R. and Sahib,S. and Hussin,B., (2014), Incorporating Evaluation Criteria in Meta-process of Classification to Increase the Acceptance Level, *13th International Conference on Applied Computer, ACACOS 2014*.
5. Hassan, N.H. and Selamat, S.R. and Sahib S. and Hussin, B., (2011), Towards Incorporation of Software Security Testing Framework in Software Development, *Springer*.

# CHAPTER 1

## INTRODUCTION

### 1.1 Background

In software application, it is observed that there are negative consequences when security is compromised. Security can be compromised when there is lack of understanding of the in hand situation. Various terms used for security and it's family, huge numbers of models and framework to refer to, had created confusions to the software practitioner to classify vulnerability that is accurate, consistence and correct.

It is observed that there is a challenge in forming a vulnerability classification scheme due to type of data used. For example, some vulnerability database like Common Vulnerabilities Exposures or CVE is very much using natural language structure but without proper English grammar as given in its web page of (*Common Vulnerabilities and Exposures: The Standard for Information Security Vulnerability Names*, 2015). One way to extract the information is by using semantic analysis (Rebolloa et al., 2015). However, in security domain, some terms are used differently. For instance, the meaning of buffer overflow is to overwrite the adjacent memory by overrun buffer and is not simply means that buffer is more than full. Therefore, it is learned that the terms must be specified with related to predefined rules of information security. Another challenge was to formally translate the domain terms into a schema that can be translated to a workable engine to extract the vulnerability given a historical database as debated in (Shaikh and Sasikumar, 2015). Therefore, this study is to focus on this scenario.

## 1.2 Problem Statement

The current vulnerability classification suffered from multiple dimensions of classifiers. They are either too specific or too complex (Ruohonen et al., 2017; Tripathi and Singh, 2011). Or they were only for dedicated cases. This lead to disability to perform a detection or protection from next attack of vulnerability. The understanding of the taxonomy which also various, requires a formal classification that can be used for generic cases regardless of applications, mobiles, networks or other devices (Burger et al., 2014).

The above research statement is divided into three research problem (RP) and the summary of the above statements are illustrated in Table 1.1.

Table 1.1: Summary of research problems

RP	Research Problems (RP)
RP1	The current vulnerability classification use multiple dimensions of classifiers are the issues needed to be addressed (Carl et al., 1994; Aslam et al., 1996b; Tripathi and Singh, 2011; Du and Mathur, 1998)
RP2	Lack of generic and systematic process to describe the vulnerability classification process , which disable to be performed on other classes. (Jiwnani and Zelkowitz, 2002; Katrina et al., 2005; S et al., 2005; Eagle et al., 2006; Bazaz and Arthur, 2007)
RP3	There is an absent of formal application to translate the vulnerability classification into solutions. (Eagle et al., 2006; Bazaz and Arthur, 2007; Lowis and Accorsi, 2011; Leitner and Rinderle-Ma, 2014). Therefore, the vulnerability classification requires a comprehensive and viable process