



Detecting malicious Java Scripts embedded in PDF files through machine learning classifiers

W T N Perera

(Reg. No.: MS19814452)

M.Sc. in IT

Specialized in Cyber Security

Supervisor: **Dr. Lakmal Rupasinghe**

December 23, 2021

**Department of Information Technology
Faculty of Graduate Studies and Research
Sri Lanka Institute of Information Technology**

Table of Contents

Table of Contents	2
List of Figures	4
List of Tables	5
Chapter 1 Introduction	6
1.1 Overview	6
1.2 Background and Motivation	9
1.3 Problem Definition	12
1.4 Research Question	12
1.5 Research Objectives	12
1.6 Outline of the Thesis	13
Chapter 2 Literature Review	14
2.1 Portable Document Format (PDF).....	14
2.1.1 Introduction	14
2.1.2 History behind PDF	14
2.1.3 PDF File Structure.....	15
2.1.4 PDF Data Types	18
2.1.5 Documents Structure	19
2.1.6 Malicious PDF.....	28
2.2 Machine Learning.....	41
2.2.1 Introduction	41
2.2.2 Machine Learning Methods	43
2.2.3 Classification Methods.....	44
2.3 Malware Analysis	49
2.4 Related Work	50
2.5 Existing tools for feature extraction	55
2.6 Conclusion.....	60
Chapter 3 Methodology.....	61
3.1 Introduction	61
3.2 Proposed System Architecture	61
3.3 Identification and Collection of Data samples	64
3.4 Feature Extraction.....	64
3.5 Classification algorithm.....	69
Chapter 4 Implementation.....	70
Chapter 5 Conclusion and Future Dimensions.....	73

List of Figures

Figure 1: Components of a PDF file.....	14
Figure 2: Example of a cross reference table.....	16
Figure 3: PDF file Trailer structure	17
Figure 4 : PDF format after the updates	17
Figure 5: Document structure of a PDF file.....	19
Figure 6: Basic document structure of a 2-page PDF file.....	20
Figure 7: Sample of document catalog	23
Figure 8: Sample page tree	25
Figure 9: Simple document information dictionary.....	26
Figure 10 : Example of trailer dictionary.....	27
Figure 11: Simple document in .tex file	27
Figure 12: Resulting PDF file	28
Figure 13 : Adding indirect reference to JavaScript object.....	37
Figure 14 : Adding second object.....	37
Figure 15 : Adding an object	37
Figure 16: xref updating.....	38
Figure 17 : Modify the catalog object	38
Figure 18 : JavaScript output	38
Figure 19 : Vulnerabilities of Adobe Acrobat Reader PDF reader software	40
Figure 20: Classification of features described in traditional Machine Learning. Approaches	42
Figure 21 : Decision tree algorithm sample	48
Figure 22: Random Forest example with two decision trees	48
Figure 23: PDF Scrutinizer	51
Figure 24: Architecture of PDF Scrutinizer.....	52
Figure 25: Hidost System Design.....	52
Figure 26: Performance Comparison Chart Of Machine Learning Models Using Hyper Parameter Tuning	53
Figure 27: Types of Features.....	54
Figure 28: Parser of the PDF Tool toolkit.....	56
Figure 29 : Malicious PDF File analyze PDF Dumper tool	56
Figure 30 : Extract PDF file using Jsunpack-n.....	57
Figure 31 : extracting elements of malicious PDF file.....	58
Figure 32: High level architecture of the proposed solution	62
Figure 33: Sample identification of embedded JavaScript in PDF file	66
Figure 34: Search For option in PDF Stream Dump	67
Figure 35: Sample JavaScript in JavaScript_UI	67
Figure 36: Sample shellcode generated via JavaScript_UI.....	67
Figure 37: scSigs command execution	67
Figure 38: Output of scLog.....	68
Figure 39: Import scikit dataset in the beginning of the code.....	70
Figure 40: Setting the configuration options.....	70
Figure 41: Functions - Data generation and split.....	71
Figure 42 : Building the classifier function.....	72

List of Tables

Table 1: Document catalog entries	22
Table 2: Entries of the dictionary Page	24
Table 3 :entries of the document information dictionary	26
Table 4: Entries of the trailer dictionary	27