Detection of Malicious OOXML Documents Using Domain Specific Features

by

Priyansh Singh

Roll. No.: 2017IS-17

विश्वजीवनामृतं ज्ञानम्

ABV–INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT GWALIOR (M.P.), INDIA

Outline

- Motivation
- Key Research
- Objectives
- Research Gaps
- Hypothesises
- Proposed Method
- Research Outcome
- Results

Motivation

- With the advent of automation in malware creation, the generation of malicious programs is relatively rudimentary. (eg. Metasploit, Microsoft Word Intruder)
- In the SophosLabs 2019 Threat Report [1], the authors have noted an increase in targeted attacks.
- Malicious documents are specifically designed to store user provided content which may or may not be executed, making it harder to identify. Many standard signatures based detection and Antivirus software often fail when encountered with malicious documents.

Motivation

- The Cisco Annual Cybersecurity Report [2], states the Office Suite extensions contributed 38% of the share, highest amongst all file extensions during January through September 2017.
- Kaspersky Labs reported an increase in attacks using Office documents. In a presentation they reported as of Q4 2018, 70% of all attacks now target Office vulnerabilities [3].
- Malicious documents are being used to drop ransomware payload. Sophos Labs 2018 Malware Forecast [4] allude the use of MSWord files to spread ransomware of the Locky family.
- Western Ukraine power supply in 2016 [5, 6] was attacked using a known vulnerability MSWord to bring down three power distribution centres and nearly 60 substations and to leave more than 230,000 residents without power.

Key Related Research

There has been little published work on detection of OOXML document detection which considers the entire file:

- Nissim et al. in [7], defines a machine learning model supported by active learning, which uses structure file paths as features. They report an accuracy of 99.6% true positive rate of 93.34% and false positive rate of 0.19%.
- Rudd et al. in [8], uses Deep Learning and Boosted Trees on office documents using 4-byte level features. They report AUC values of 0.99 or greater with Deep Learning.

Objectives

- **Detection of malicious documents:** To develop and implement features and feature extraction strategy based on domain knowledge of Office document formats. These features are paired with numerous machine learning models to identify and categorise documents in malicious and benign classes seamlessly and efficiently.
- **Evaluation Against Novel Attacks:** In a realistic scenario, not all the possible malware configurations are known. Furthermore, new vulnerabilities, malware mutate, and new malware campaigns are discovered every week. The model developed needs to be tested against attacks which are unknown to the system at the time of training.

Research Gaps

- **R1:** There is limited peer-reviewed research on the topic of detection of malicious Office documents. There has been no research conducted which takes into account the domain knowledge of the OOXML while performing the detection task.
- **R2:** No research has been pursued, which focuses on making the detector customisable at the time of deployment.

Research Gaps

- **R3:** Temporal evaluation, where the dataset is split according to some time metric, provides a more realistic evaluation. This method of evaluation hasn't been pursued with the exception of a few research articles.

- **R4:** The datasets used in some of the research articles discussed use smaller or simulated datasets. The use of larger datasets leads to more statistically stable results.

Hypothesises

- **H1:** It is hypothesised forensic identifiers and metadata present in an OOXML document are ideal candidates for customisable and domain-specific feature representation. Additional representation in the form of structural paths and entropy of selected files, as well as byte histograms entropy and byte histograms, provide robustness to the feature domain.

- **H2:** It is hypothesized that metadata timestamps collected from XML tags like *created*, *lastPrinted*, and *modified* can be effectively used to split the dataset for temporal analysis.

Proposed Method I

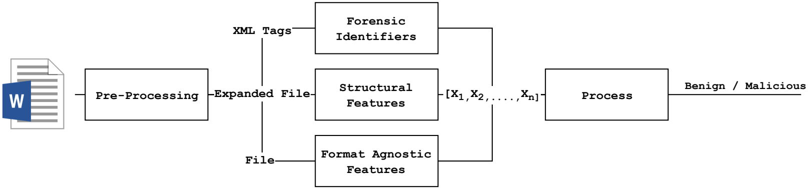


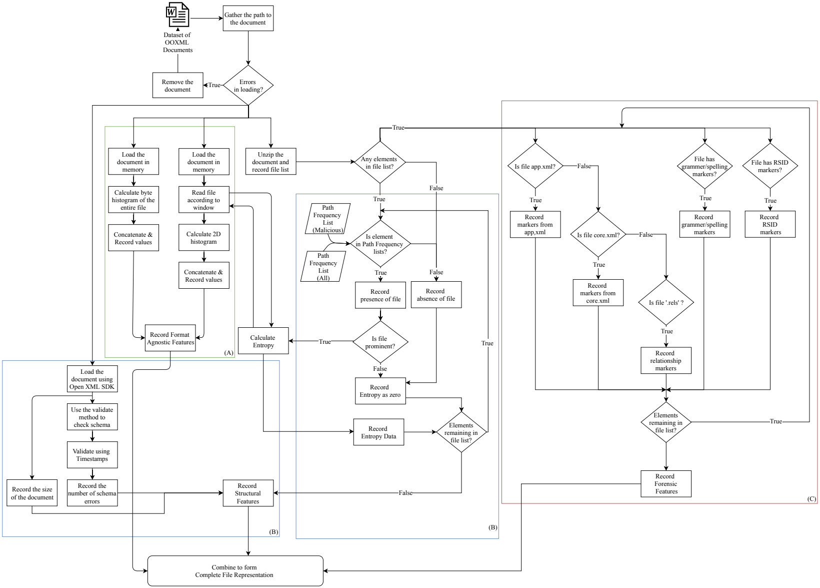
Figure 1: Architecture of the proposed method

Proposed Method II

Table 1: Summary of Format Agnostic & Structural Features

Feature	Description	Data Type
Byte-entropy histogram	2D byte entropy histogram of the file	256 floats
Byte histogram	Byte histogram of the file	32 floats
Size	Stores the size of the file in number of bytes	integer
Error Number	Number of errors found by Open XML SDK	integer
Prominent files	Presence or absence of file based of factor	boolean
Entropy of prominent files	Entropy of prominent Files	float

Proposed Method III



Dataset of OOXML Documents
Gather the path to the document
Errors in loading?
True
Remove the document
Load the document in memory
Calculate byte histogram of the entire file
Concatenate & Record values
Load the document in memory
Read file according to window
Calculate 2D histogram
Concatenate & Record values
Record Format Agnostic Features
(A)
Unzip the document and record file list
Calculate Entropy
Load the document using Open XML SDK
Use the validate method to check schema
Validate using Timestamps
Record the size of the document
Record the number of schema errors
Record Structural Features
(B)
Any elements in file list?
True
False
True
Path Frequency List (Malicious)
Path Frequency List (All)
Is element in Path Frequency lists?
True
False
Record presence of file
Record absence of file
Is file prominent?
True
False
Record Entropy as zero
Record Entropy Data
Elements remaining in file list?
True
False
(B)
Is file app.xml?
False
True
Record markers from app,xml
Is file core.xml?
False
True
Record markers from core.xml
Is file '.rels' ?
True
Record relationship markers
File has grammer/spelling markers?
True
Record grammer/spelling markers
File has RSID markers?
True
Record RSID markers
Elements remaining in file list?
True
Record Forensic Features
(C)
Combine to form Complete File Representation

Proposed Method IV

Table 2: Summary of Forensic Identifiers

Feature	Description	Data Type
Revision Identifiers	Number of types of revisions	18 integers
Relationship Identifiers	Count of various relationships	5 integers
Grammar and Spelling	Count of grammar and spelling errors	3 integers
app.xml indicators	Identifiers from app.xml	24 integers
core.xml indicators	Identifiers from core.xml	14 integers
timeSincePrinted	Time since last print(in Sec)	integer
timeSinceCreation	Time since creation (in Sec)	integer
timeSinceModified	Time since last modification (in Sec)	integer
timeSpentIn-Document	Time spent in document (in Sec)	integer

Proposed Method V

Table 3: Summary of Feature Selection and Classifiers used to test the proposed method

Feature Selection	Classifiers	FL- All	FL- Forensic
Chi-Square	Linear Support Vector Machines	10	10
Information Gain	RBF Support Vector Machines	50	15
Feature Importance	Random Forests	75	20
	XGB Classifier	100	35
		150	50
		200	68
		400	
		800	

Research Outcome I

- First application of forensic identifiers and metadata in modern office documents for malware detection. With just these markers the method performed at an accuracy of 99.83%

- A static detection methodology based on forensic identifiers, structural features, byte histograms and byte histograms to detect modern office malware.

- Proposed detection methodology was tested against the standard, and temporal evaluation and results were presented.

Research Outcome II

- As it pertains to *R1, R2 and H1*, standard evaluation with forensic feature class and all three feature classes prove these features can be used to detect malicious documents and at the same time they can be used to customise if need be.

- *R3* identify the generalisability testing and temporal analysis of the detector a research gap. *H2* suggest the use of metadata stored in the document as a method to perform this. The proposed technique under temporal analysis leads to the best accuracy of 96.45%, thereby crediting the *created* tag is a good metric to split the Dataset.

Research Outcome III

- To cover *R4*, the files were collected from sources of repute like VirusTotal and CommonCrawl. A total of 705 malicious and 35,000 benign files were downloaded.

Table 4: The composition of Datasets for standard evaluation.

Name	Malicious	*.docm* and *.dotm*	*.docx* and *.dotx*	Malicious : Benign
Set I	664	552	444	40 - 60
Set II	664	552	997	30 - 70
Set III	664	552	2,104	20 - 80
Set IV	664	552	12,076	05 - 95

Result I

Table 5: Results from testing 705 malicious documents against Windows Defender and Kaspersky Total Security

Document Format	Total	Windows Defender	Kaspersky Total Security
OOXML Document	705	4.25% (30 files)	16.45% (109 files)

Result II

Table 6: Detection accuracy (ACC %), False Positive Rate (FPR), Precision (PREC), Recall, F-Score (F-1) and Area Under Curve (AUC) of malware detection system for four different Datasets (DS) using all the feature classes

DS	CL	FS	FL	ACC	FPR	PREC	RECALL	F-1	AUC
SET I	XGB	FI	200	99.16	0.015	0.990	0.9960	0.993	0.9991
SET II	XGB	IG	150	99.37	0.014	0.994	0.9968	0.995	0.9992
SET III	XGB	IG	200	99.58	0.014	0.997	0.9981	0.997	0.9992
SET IV	XGB	IG	100	99.82	0.018	0.999	0.9990	0.999	0.9995

Result II

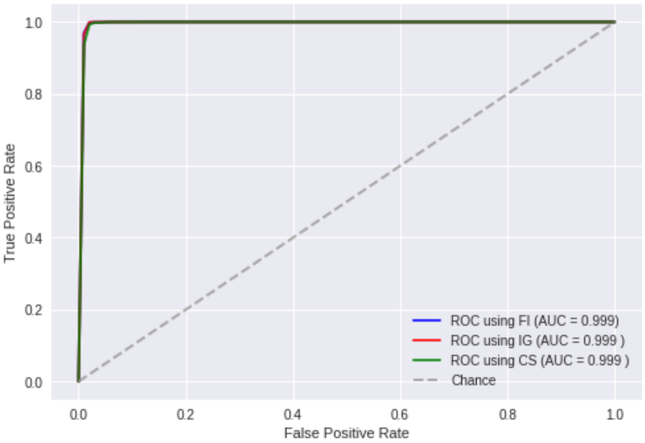


Figure 2: ROC curves for feature selection techniques used by different classifiers for 100 features on Datasets-IV and XGB classifier

Result III

Table 7: Detection accuracy (ACC %), False Positive Rate (FPR), Precision (PREC), Recall, F-Score (F-1) and Area Under Curve (AUC) of malware detection system for four different Datasets (DS) using forensic feature class

DS	CL	FS	FL	ACC	FPR	PREC	RECALL	F-1	AUC
SET I	RF	FI	50	99.28	0.014	0.991	0.9970	0.994	0.9981
SET II	XGB	FI	50	99.32	0.012	0.995	0.9955	0.995	0.9989
SET III	XGB	CS	10	99.55	0.015	0.996	0.9981	0.997	0.9986
SET IV	XGB	IG	15	99.83	0.020	0.999	0.9993	0.999	0.9988

Result IV

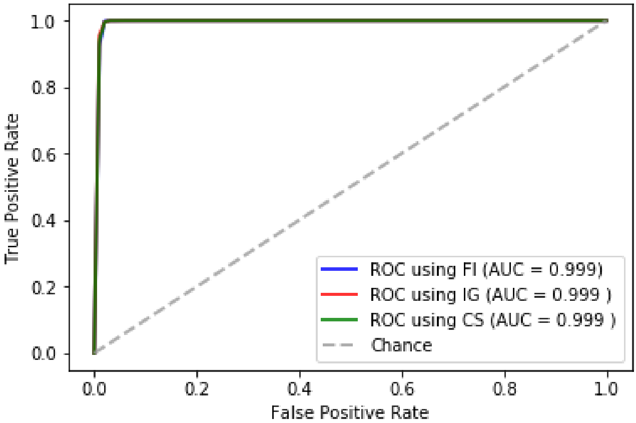


Figure 3: ROC curves for feature selection techniques used by different classifiers for 100 features on Datasets-IV, XGB classifier and all feature classes

Result V

Table 8: Detection accuracy (%) of malware classifier using all features reduced to eight feature length sets using information gain and feature importance for the temporal dataset

Dataset	FL	Information Gain		Feature Importance	
		RF-Accuracy	XGB-Accuracy	RF-Accuracy	XGB-Accuracy
Temporal	50	93.79	94.01	95.12	94.01
Temporal	75	93.79	94.01	94.24	95.57
Temporal	100	94.46	94.68	94.68	95.12
Temporal	150	96.01	94.46	94.68	94.90
Temporal	200	**96.45**	94.46	94.46	**96.01**

Result Comparison

Table 9: Comparison of Accuracy (%), True Positive Rate (%), False Positive Rate (%) and F-Score between [7] and proposed method.

Proposed Method				[7]			
ACC	TPR	FPR	F-Score	ACC	TPR	FPR	F-Score
99.82	99.93	2.0	0.999	99.67	93.34	0.19	0.957

- Both the proposed method and [8] achieve Area Under the Curve of an ROC of over 0.99 albeit in different configurations.

Research Outcome

During the course of this thesis, two research articles were communicated to international journals. The titles of these articles are as follows:

1. Singh P. and Tapaswi S., ‘Detection of Malicious Office Documents Employing Forensic Identifiers’ (Communicated)
2. Singh P., Tapaswi S., and Gupta, S., ‘Malware Detection in PDF and Office Documents: A Survey’ (Communicated)

References I

Sophos.
Sophoslabs 2019 threat report, 04 2018.
Accessed: 04 Feburary 2019.

Cisco.
Cisco 2018 annual cyber security report, 02 2018.
Accessed: 2018-09-19.

Catalin Cimpanu.
Kaspersky: 70 percent of attacks now target office vulnerabilities.

Sophos.
Sophoslabs 2018 malware forecast, 11 2017.

Kim Zetter.
Inside the cunning, unprecedented hack of ukraine's power grid, 03 2016.

References II

Kelly Jackson Higgins.
Macros, network sniffers, but still no 'smoking gun' in ukraine blackout, 01 2016.

Nir Nissim, Aviad Cohen, and Yuval Elovici.
Aldocx: detection of unknown malicious microsoft office documents using designated active learning methods based on new structural feature extraction methodology.
IEEE Transactions on Information Forensics and Security, 12(3):631–646, 2017.

Ethan M Rudd, Richard Harang, and Joshua Saxe.
Meade: Towards a malicious email attachment detection engine.
arXiv preprint arXiv:1804.08162, 2018.