Malware Prediction Techniques: Selection and Implementation for Integrated Cyber Evidence

Suriayati Chuprat[1], Mohd Naz'ri Mahrin1, Syahid Anuar[1], Aswami Ariffin[2], Fakhrul Afiq Abd Aziz[2], Muhammad Zaharudin Ahmad Darus[2], Mohd Zabri Adil Talib[2]
[1]Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia, Malaysia
[2]CyberSecurity Malaysia, Malaysia.

Summary
Recent studies have shown the negatives impact of malware attacks are increasing. To prevent malware attack in more proactive way, predictions of such attacks are needed. However, the quality and the accuracy of these predictions are determined by the applied techniques. In this paper, we report our findings on selecting and implementing such techniques in predicting malware attacks. For the selection process, we conducted a systematic review and searched over 5 major databases. 89 articles on malware predictions were finally included and prediction techniques are classified. As part of our on-going development project known as Integrated Cyber Evidence (ICE), we evaluated the selected technique using actual data of malware attacks. The results of evaluation had helped us to decide the final technique to be implemented in prediction module of ICE systems.

Key words:
Algorithms, Malware, Machine Learning, Predictions.

1. Introduction

The threat (and the effects thereof) of malware will expand considerably in the coming years, mainly due to the improvements in techniques and goals (Crimeware, APTs, etc.) There is struggle against malware spins off from different areas which ranging from the awareness among users to adopt security measures to the development of anti-malware software by specialized companies [68, 36]. This struggle also develops through the setting up of adequate security policies in different agencies and companies. Over the past decade, there has been an increase in the number of types of malware created and this eventually leads to the existence of their effects. According to a study reported by PandaLabs Annual Report 2017, the mean number of computers infected by malware is currently 31.88%, the countries with the highest infection rates are China (52.26%), Turkey (43.59%), Peru (42.14%), and Bolivia (41.67%). On the other hand, the countries least affected are Sweden (21.03%), Nor-way (21.14%), and Germany (24.18%).

The economic losses caused by malware in its different scenarios (government agencies, companies and individuals) are huge and have been estimated at thousands of millions of dollars per year. The 2016 McAfee Labs Report mentioned that mal-ware is still at large with significant new changes to the kinds of threats such as file-less attacks, exploitation of remote shell and remote control protocols, encrypted infiltrations, and credential theft which are harder to detect. In addition, this report claimed that Stuxnet and supporting Duqu, Flame, and Gauss malware have been developed tosecretly target specific devices and make minor configuration changes that would result in a major impact, for example to a nuclear program. The intent was not to destroy a computer or harvest massive amounts of data. Instead, it was to achieve the attackers' goals by carefully selecting the modified working systems.

In December 2016, Kaspersky Lab detected over 1,966,324 registered notifications on attempted malware infections that aimed to steal money via online access to bank accounts. Ransomware programs were detected on 753,684 computers of unique users; 179,209 computers were targeted by encryption ransomware. Kaspersky antivirus solution also detected 121,262,075 unique malicious objects: scripts, exploits, executable files, etc. and this could be one of the reasons why 34.2% of computer users were subjected to at least one web attack over the year.

These dramatically increased threat had given us the significant reason to strengthen the national security in more proactive way. Thus, in our on-going research project, we are developing an integrated malware analytics framework that will expose the future threats of malware attacks. However, in order to predict the future threats, we need to select an efficient prediction technique and apply them in the malware analytics framework. In this paper, we will present the selection process using systematic literature review over 5 major databases and 89 articles on malware prediction were finally included. These 89 articles on malware prediction has been reviewed, analyzed, and then classified. We then perform an experimental evaluation using actual malware attack data and report the performance of the selected technique.

The remaining of this paper is structured as follow: In Section 2, we briefly describe the project methodology and brief overview of ICE project. Section 3 describes the methodology of the conducted systematic review. In section 4 describes the result of the systematic review and the selection of malware prediction technique. We described the implementation and evaluation of the selected techniques in Section 5. Finally, in Section 6 we conclude

Manuscript received August 5, 2018
Manuscript revised August 20, 2018

this paper by summarizing the results, and highlighting some ideas on future work.

2. ICE Overview and Project Methodology

In this section we briefly describe the overview of the on-going project known as Integrated Cyber-Evidence (ICE) systems and part of the project methodology. ICE is a system in which aimed to have the ability to learn the trend of malware attacks and predict the future attacks. This system consists of three main modules which include: Data Warehouse, Data Analytics and Visualization. Data Warehouse is a component that will be the central repository for the storing the collected data. Any unstructured data will be transformed into a structured data and some will need to be enriched to become more meaningful for further analysis.

In Data Analytics component, data will be further analyzed, correlated and uncovered any possible hidden patterns or connections using collected historical data. Besides, descriptive analytics, this component will also be performing predictive analytics to discover the future or unknown malware attacks. Finally, in Data Visualization component, the data patterns, connections, and prediction will be presented in a pictorial or graphical format to enable decision makers to view the analytics graphically. These reports are crucially important as it will be used as evidence in forensic investigations and can be used to distribute warning to the targeted organization. However, in this paper, we only present part of the on-going project that related to data analytics. Figure 1 depicted the project methodology presented in this paper, consists of four main phases.

In phase 1, we conducted an extensive Systematic Literature Review (SLR) in order to investigate the current techniques used for malware prediction. This task will be further described in Section 3. The results of the SLR provided us with a list of techniques. In phase 2, we rank the list and short-listed the most used techniques. We then analyze each technique and match the suitability with collected malware attacks data. In phase 3, we evaluated the selected technique, by benchmarking it with other commonly used technique and confirm the performance. Finally, in Phase 4 we identified the best technique and implemented in predicting malware attacks in Data Analytics module.

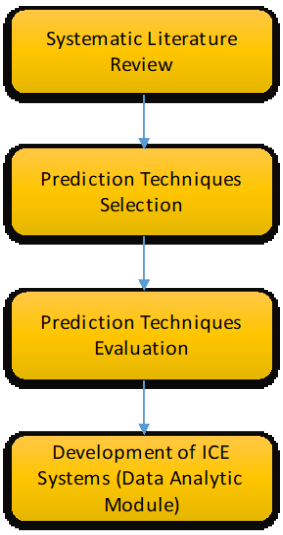

Fig. 1 The Project Methodology

3. Systematic Literature Review

In this section we present the methodology used in conducting this systematic review. This methodology was built based on the well-known guideline by Kitchenham and Brereton [45].

3.1 Formulating the Research Questions

The first step conducted in this SR methodology is to derive the research question. Based on our early investigation on the problem background, we derived the following research question: What are the existing prediction techniques for malware threats/attacks?

3.2 Identify the Search String

By considering the identified research questions, we outlined the research keywords which include: Malware OR Malicious OR Attacks OR Threat, Prediction, Technique. Then, by using the outlined research keywords, we identify the search string and used it in searching the related literature. The identified search string is:

<< ***((Malware OR Malicious OR Attacks OR Threat) AND Prediction AND Techniques)***>>

3.3 Search Strategy

The third step in our SR is to execute the literature search using the identified search string. We execute the search for the period of Jan 2010 to October 2016, using the following search strategy: (1) Automatic search in 5 major databases (IEEE Explore Digital Library, Science Direct, ACM Digital Library, Springer Link and Wiley Online Library); (2) Manual search in conferences proceedings and journals;

(3) Snowballing for a complete set of primary Malware papers. The result of the initial search is shown in Table 1.

Table 1: Search Result

Num	Databases	Number of Papers
1.	IEEE Explore Digital Library	11
2.	Science Direct	280
3.	ACM Digital Library	25
4.	Springer Link	254
5.	Wiley Online Library	100

3.4 Applying the Inclusion and Exclusion Criteria

During the initial selection we apply a set of inclusion and exclusion criteria based on guideline proposed by Kitchenham and Brereton [45] and Khanian and Mahrin [41], to ensure only relevant works on malware prediction were accepted into the SR. The inclusion and exclusion criteria were applied in 6 phases and the results are presented in Table 2.

Table 2: Inclusion and Exclusion Criteria

Phase (P)	Inclusion/exclusion criteria	Number of paper after applying Inclusion/exclusion
P1	Searching literature via the search string on electronic databases to cover journal articles, workshops and conference papers	670
P2	Excluding numbers of literature that is a short paper, a poster presentation, prefaces, editorials, slides presentation, non-English papers	310
P3	Removing duplicate literatures that emerge in different databases	280
P4	The literature must be a peer-reviewed	150
P5	Read the full paper (the introduction, method section and conclusion)	110
P6	Excluding literatures that were not related to malware prediction	89

4. Prediction Techniques Selection

In order to select the prediction techniques, we reviewed and classified the 89 articles according to the proposed techniques for malware prediction. The effectiveness of these techniques are based on features extracted using dynamic or static analysis that has been presented in the domain of malware detection and the field of malicious document detection. The prediction techniques proposed by researches in the 89 articles are listed in Table 3(a) and Table 3(b). These prediction techniques provide the relevance of the features for identifying the searched malwares, and on the quality of training data for being unbiased and representative of malwares. Some articles [77,12,4] have proposed the structural feature extraction methodology for the detection of unknown malwares using machine learning algorithms. The same result was also proposed by [22] who apply classification algorithms to classify unknown malicious in documents based on structural features.

Table 3(a): Malware Prediction Techniques

Techniques for malware prediction	References
Bipartite graph	[61]
API call graph	[29]
Graph structure + Clustering process	[47]
Control flow graph (CFG)	[1], [26]
Fuzzy	[48], [37], [38],[39]
Fuzzy + Association rules	[19], [20]
Fuzzy+ Clustering method	[4]
Network intrusion activity on computer network	[80]
Markov Model	[44], [74]
Markov Model + Entropy-based detection	[14]
Stochastic Model	[50]
Ensemble learning algorithms	[51], [67], [54],
	[12]
Ensemble Methods + Harmony search	[69]
Clustering algorithms	[31],[8], [58], [10]
Clustering + Genetic algorithm	[52]
Propagation model	[17]
Propagation model + File relation graph, Active learning method	[59]
Honeypot technique + Association rule mining	[34]
Honeypot technique	[60]
Decision tree classifiers (J48, Random Forest (RF))	[76], [81],
	[24], [5]
Decision tree + Feature selection algorithm	[77]
Decision trees + Adaboost	[42]
Decision trees + Support Vector Machines (SVMs)	[18]
Support Vector Machine (SVM)	[78], [36],[53], [71]
SVM + Interpretable string analysis	[88]
SVM + graph kernels	[11], [9]
Speculative execution	[84]
Forecasting modeling	[34], [42]

Table 3(b): Malware Prediction Techniques

Techniques for malware prediction	References
Multi Agent Systems	[56]
Neural Network	[65]
Application's network traffic patterns	[68]
Logistic Regression	[40]
Static analysis techniques + Classification algorithm	[82]
Static analysis techniques	[63], [73]
Static analysis + Dynamic analysis	[75]
Partial matching classification algorithm	[91]
AccessMiner (system-centric approach)	[32]
Collaborative decision fusion	[69]
Motivation Theory	[23]
Text mining + Information retrieval	[72]
Sequential association rule	[43]
Association algorithm + connectivity metric	[16]
Associative classification (Classification + Association rule)	[85]
Association rule + Learning-based method	[90]
Object oriented association mining + called API's	[25], [87]
Sequential pattern mining + Nearest Neighbor classifier	[30]
Pattern mining + Hooking	[6]
Frequent pattern mining	[26]
Nearest-Neighbor algorithm (KNN)	[2], [46]
Naive Bayes classifier	[27]
Naive Bayes classifier + Logistic regression + Threshold matching + Rank based	[66]
Naive Bayes + Dimensionality reduction with Markov Blanket	[62]
Classification algorithms (Decision trees, KNN, SVM, Artificial neural network, Logistic Regression, Hierarchical Clustering)	[55]
Classification algorithms (Decision trees, KNN, SVM, Naive Bayes)	[64]
Classification algorithms (Decision trees, SVM,	[81]
AdaBoost, logistic regression)	
Classification algorithms (AdaBoost, Decision trees,	[22]
Bayesian Network, Naive Bayes, Sequential Minimal	
Optimization, Logistic Regression, Bagging)	
Classification algorithms (Decision trees, Bayes network, KNN, multi-layer perceptron) + Anomaly-based	[57]
Classification algorithms (SVM, rule learning, Decision tree classifiers (J48, Random Forest))	[3]
Classification algorithms (Decision trees, SVM, KNN, logistic, Naive Bayes, Adaptive regularization of weights)	[83]
Lazy associative classification algorithm + Execution- based dynamic analysis	[89]
Positive selection classification algorithm	[33]
Behavior-based detection technique	[49]
N Gram-based attribution method	[21]
Header information technology	[79]
Swarm-based approach + Stigmergic communication	[15]
Hierarchical associative classification	[86]

The articles were classified by the most used techniques in malware detection as showed at Figure 1. Techniques that have been employed less than three times have been classified in "Others". It is apparent malware prediction researches increased the employing classification algorithms such as Decision trees (14 out of 89 papers) and SVM (12 out of 89 papers). Among data mining techniques, also Fuzzy, KNN, Clustering and association rule mining have been used the most often in malware prediction researches (7 out of 89 papers).

These techniques are able to predict the unknown, new malwares accurately, by feature selection process and feature extraction process. Researchers selected these techniques to categorize the features of malware into static features which are pertaining to installation files, dynamic features which are pertaining to the behavior of the application after installation or hybrid features which are combination of both dynamic and static features and features extracted from executable files include printable strings, byte code n-gram, system calls, instruction sequence and opcode n-gram. On the other hand, these classification techniques extracted the features (i.e., byte sequences, printable strings, and system resource information) from malware samples via dynamic analysis or static analysis and based on the extracted features identify the malware automatically.

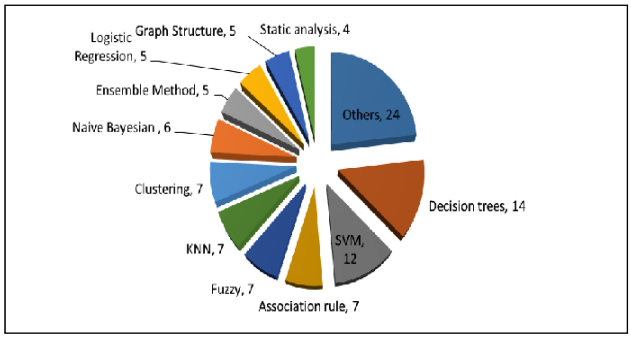

Fig. 2 Classified Prediction Techniques

5. Prediction Techniques Evaluation

As shown in Figure 2, decision tree and SVM was mainly used for malware prediction. However, we selected SVM over decision tree as our data set is not suitable to be used with decision tree in predicting the malware attack. Thus, we further compare SVM with four other techniques to evaluate the technique performance, before implementing it in the development of Integrated Cyber Evidence (ICE) systems. Figure 2 shows the results of predictive analytics of Botnet attacks trend forecasting for December 2016 using SVM, ARIMA (Auto-Regressive Integrated Moving Average) model, linear regression, random forest and ANN. The analytics was estimated on the training data from January 2016 to November 2016. The actual attack data for the period of December 2016 are also shown in blue line.

Sliding window method has been used to model our predictive analytics. The sliding window method performs in a way: in the training set, we use y(i) as input and y(i+1) as output, iteratively constructed the sample in this way to

form the training set, then train the model to predict one step ahead (or multi-steps). To evaluate the forecast accuracy, most commonly used scale-dependent measures are based on the absolute errors or squared errors, and scaled mean absolute. The forecast errors are the difference between the actual values in the test set and the forecasts produced using only the data in the training set. The forecast accuracy measures of botnet attack prediction model by using selected ML algorithms are computed in Table 4.

Table 4: Malware Prediction Techniques

	MSE (Mean Square Error)	RMSE (Root Mean Square Error)	MAE (Mean Absolute Error)
Linear Regression	0.0581	0.241	0.1994
Random Forest	0.1959	0.4426	0.3852
ANN	0.1083	0.3291	0.2872
SVM	0.0442	0.2102	0.1689
ARIMA	0.0454	0.213	0.1711

The results show the comparison of trend forecasting based on SVM, linear regression, random forest, ANN and ARIMA. Notice that the MAE values of forecasting using linear regression, SVM and ARIMA accuracy measures are less than 0.5. This means that the scaled error of the ML forecast algorithms is better than the average forecast computed on the training data. In our context, it is harder to predict the mal-ware attack more accurately could be because of smaller training dataset.

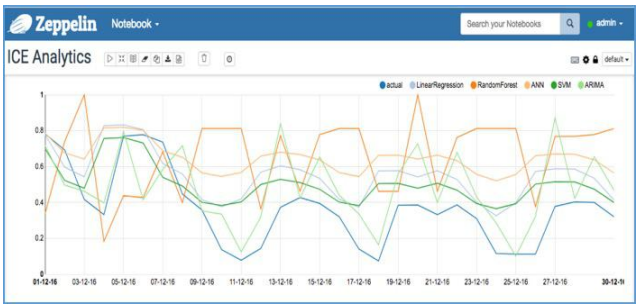

Fig. 3 Prediction Techniques Evaluation

As shown in Figure 3, the fluctuation trend of botnet attack could also have influenced on the accuracy of the prediction. Random Forest forecast algorithm, which is shown in dark orange line color, has the lowest accuracy with the highest estimation error of MAE at 0.385.

6. Conclusion and Future Work

Malware is the primary choice of weapon to carry out malicious intents in the cyberspace, either by exploitation into existing vulnerabilities or utilization of unique characteristics of emerging technologies. In this paper, we have presented an extensive systematic literature review on the malware prediction technique. Using 6 clearly predefined selection criteria, 89 malware prediction papers have been strictly selected, and then reviewed. From these primary malware prediction papers, we have extracted and synthesized the data to answer the research question. These 89 articles on malware prediction has been reviewed, and then classified by techniques proposed in detection of new malware. Among the classified techniques, we have implemented the chosen technique SVM using actual malware attack data and evaluated the technique with other common used prediction techniques. Based on the experimental evaluation, SVM was also proven to be the most accurate. Thus, we have proposed this prediction technique to be implemented in the Data Analytics module of the Integrated Cyber Evidence (ICE) systems.

Acknowledgment

This project was funded by the DSTIN grant of Malaysia Government granted to CyberSecurity Malaysia in collaboration with Universiti Teknologi Malaysia.

References

[1] Alam, S., Horspool, R. N., Traore, I., & Sogukpinar, I. (2015). A framework for metamor-phic malware analysis and real-time detection. computers & security, 48, 212-233.

[2] Alazab, M. (2015). Profiling and classifying the behavior of malicious codes. Journal of Systems and Software, 100, 91-102.

[3] Allix, K., Bissyandé, T. F., Jérome, Q., Klein, J., & Le Traon, Y. (2016). Empirical as-sessment of machine learning-based malware detectors for Android. Empirical Software Engineering, 21(1), 183-211.

[4] Altaher, A. An improved Android malware detection scheme based on an evolving hy-brid neuro-fuzzy classifier (EHNFC) and permission-based features. Neural Computing and Applications, 1-11.

[5] Al-Bataineh, A., & White, G. (2012, October). Analysis and detection of malicious data exfiltration in web traffic. In Malicious and Unwanted Software (MALWARE), 2012 7th International Conference on (pp. 26-31). IEEE.

[6] Ahmadi, M., Sami, A., Rahimi, H., & Yadegari, B. (2013). Malware detection by behav-ioural sequential patterns. Computer Fraud & Security, 2013(8), 11-19.

[7] Asmitha, K. A., & Vinod, P. (2014). Linux malware detection using non-parametric sta-tistical methods. In Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on (pp. 356-361). IEEE.

[8] Apel, M., Biskup, J., Flegel, U., & Meier, M. (2009, September). Towards early warning systems–challenges, technologies and architecture. In International Workshop on Critical Information Infrastructures Security (pp. 151-164). Springer Berlin Heidelberg.

[9] Anderson, B., Quist, D., Neil, J., Storlie, C., & Lane, T. (2011). Graph-based malware de-tection using dynamic analysis. Journal in computer virology, 7(4), 247-258.
[10] Aresu, M., Ariu, D., Ahmadi, M., Maiorca, D., & Giacinto, G. (2015, October). Cluster-ing android malware families by http traffic. In Malicious and Unwanted Software (MALWARE), 2015 10th International Conference on (pp. 128-135). IEEE.
[11] Ban, T., Isawa, R., Guo, S., Inoue, D., & Nakao, K. (2013, August). Application of string kernel based support vector machine for malware packer identification. In Neural Networks (IJCNN), The 2013 International Joint Conference on (pp. 1-8). IEEE.
[12] Bai, J., & Wang, J. (2016). Improving malware detection using multi-view ensemble learning. Security and Communication Networks, 9(17), 4227-4241.
[13] Bocchi, E., Grimaudo, L., Mellia, M., Baralis, E., Saha, S., Miskovic, S. & Lee, S. J. (2016). MAGMA network behavior classifier for malware traffic. Computer Networks, 109, 142-156.
[14] Canfora, G., Mercaldo, F., & Visaggio, C. A. (2016). An HMM and structural entropy based detector for Android malware: An empirical study. Computers & Security, 61, 1-18.
[15] Castiglione, A., De Prisco, R., De Santis, A., Fiore, U., & Palmieri, F. (2014). A botnet-based command and control approach relying on swarm intelligence. Journal of Network and Computer Applications, 38, 22-33.
[16] Caglayan, A., Toothaker, M., Drapeau, D., Burke, D., & Eaton, G. (2012). Behavioral analysis of botnets for threat intelligence. Information Systems and e-Business Management, 10(4), 491-519.
[17] Chen, Z., Wang, M., Xu, L., & Wu, W. (2015). Worm propagation model in mobile net-work. Concurrency and Computation: Practice and Experience.
[18] Chen, Z., Roussopoulos, M., Liang, Z., Zhang, Y., Chen, Z., & Delis, A. (2012). Malware characteristics and threats on the internet ecosystem. Journal of Systems and Software, 85(7), 1650-1672.
[19] Chan, G. Y., Lee, C. S., & Heng, S. H. (2013). Discovering fuzzy association rule pat-terns and increasing sensitivity analysis of XML-related attacks. Journal of Network and Computer Applications, 36(2), 829-842.
[20] Chan, G. Y., Lee, C. S., & Heng, S. H. (2014). Defending against XML-related attacks in e-commerce applications with predictive fuzzy associative rules. Applied Soft Computing, 24, 142-157.
[21] Chouchane, R., Stakhanova, N., Walenstein, A., & Lakhotia, A. (2013). Detecting ma-chine-morphed malware variants via engine attribution. Journal of Computer Virology and Hacking Techniques, 9(3), 137-157.
[22] Cohen, A., Nissim, N., Rokach, L., & Elovici, Y. (2016). SFEM: Structural feature ex-traction methodology for the detection of malicious office documents using machine learning methods. Expert Systems with Applications, 63, 324-343.
[23] Dang-Pham, D., & Pittayachawan, S. (2015). Comparing intention to avoid malware across contexts in a BYOD-enabled Australian university: A Protection Motivation Theory approach. Computers & Security, 48, 281-297.
[24] De Lille, D., Coppens, B., Raman, D., & De Sutter, B. (2015, October). Automatically combining static malware detection techniques. In Malicious and Unwanted Software (MALWARE), 2015 10th International Conference on (pp. 48-55). IEEE.
[25] Ding, Y., Yuan, X., Tang, K., Xiao, X., & Zhang, Y. (2013). A fast malware detection al-gorithm based on objective-oriented association mining. Computers & security, 39, 315-324.
[26] Eskandari, M., & Hashemi, S. (2012). A graph mining approach for detecting unknown malwares. Journal of Visual Languages & Computing, 23(3), 154-162.
[27] Eskandari, M., Khorshidpour, Z., & Hashemi, S. (2013). HDM-Analyser: a hybrid analysis approach based on data mining techniques for malware detection. Journal of Computer Virology and Hacking Techniques, 9(2), 77-93.
[28] Eskandari, M., & Raesi, H. (2014). Frequent sub-graph mining for intelligent malware detection. Security and Communication Networks, 7(11), 1872-1886.
[29] Elhadi, A. A. E., Maarof, M. A., Barry, B. I., & Hamza, H. (2014). Enhancing the detec-tion of metamorphic malware using call graphs. Computers & Security, 46, 62-78.
[30] Fan, Y., Ye, Y., & Chen, L. (2016). Malicious sequential pattern mining for automatic malware detection. Expert Systems with Applications, 52, 16-25.
[31] Fachkha, C., Bou-Harb, E., & Debbabi, M. (2015). On the inference and prediction of DDoS campaigns. Wireless Communications and Mobile Computing, 15(6), 1066-1078.
[32] Fattori, A., Lanzi, A., Balzarotti, D., & Kirda, E. (2015). Hypervisor-based malware pro-tection with accessminer. Computers & Security, 52, 33-50.
[33] Fuyong, Z., & Deyu, Q. (2011). Run-time malware detection based on positive selection. Journal in computer virology, 7(4), 267-277.
[34] Jiang, C. B., Liu, I., Chung, Y. N., & Li, J. S. (2016). Novel intrusion prediction mecha-nism based on honeypot log similarity. International Journal of Network Management.
[35] Jones, M., Kotsalis, G., & Shamma, J. S. (2013). Cyber-attack forecast modeling and complexity reduction using a game-theoretic framework. In Control of Cyber-Physical Systems (pp. 65-84). Springer International Publishing.
[36] Huda, S., Abawajy, J., Alazab, M., Abdollalihian, M., Islam, R., & Yearwood, J. (2016). Hybrids of support vector machine wrapper and filter based framework for malware detec-tion. Future Generation Computer Systems, 55, 376-390.
[37] Huang, H. D., Lee, C. S., Hagras, H., & Kao, H. Y. (2012, October). TWMAN+: A Type-2 fuzzy ontology model for malware behavior analysis. In Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on (pp. 2821-2826). IEEE.
[38] Huang, H. D., Acampora, G., Loia, V., Lee, C. S., Hagras, H., Wang, M. H., ... & Chang, J. G. (2013). Fuzzy markup language for malware behavioral analysis. In On the Power of Fuzzy Markup Language (pp. 113-132). Springer Berlin Heidelberg.
[39] Huang, H. D., Lee, C. S., Wang, M. H., & Kao, H. Y. (2014). IT2FS-based ontology with soft-computing mechanism for malware behavior analysis. Soft Computing, 18(2), 267-284.
[40] [40] Kantchelian, A., Tschantz, M. C., Afroz, S., Miller, B., Shankar, V., Bachwani, R., & Tygar, J. D. (2015, October).

Better malware ground truth: Techniques for weighting anti-virus vendor labels. In Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security (pp. 45-56). ACM.

[41] Najafabadi, M. K., and Mahrin, M. N. R. (2016). A systematic literature review on the state of research and practice of collaborative filtering technique and implicit feedback. The Artificial Intelligence Review, 45(2), 167.

[42] Kollias, S., Vlachos, V., Papanikolaou, A., Chatzimisios, P., Ilioudis, C., & Metaxiotis, K. (2014). A global-local approach for estimating the Internet's threat level. Journal of Communications and Networks, 16(4), 407-414.

[43] Kim, Y. H., & Park, W. H. (2014). A study on cyber threat prediction based on intru-sion detection event for APT attack detection. Multimedia tools and applications, 71(2), 685-698.

[44] Kim, D. H., Lee, T., Kang, J., Jeong, H., & In, H. P. (2012). Adaptive pattern mining model for early detection of botnet-propagation scale. Security and Communication Net-works, 5(8), 917-927.

[45] Kitchenham, B., and Brereton, P. (2013). A systematic review of systematic review pro-cess research in software engineering. Information and software technology, 55(12), 2049-2075.

[46] Lakhotia, A., Walenstein, A., Miles, C., & Singh, A. (2013). VILO: a rapid learning nearest-neighbor classifier for malware triage. Journal of Computer Virology and Hacking Techniques, 9(3), 109-123.

[47] Lee, J., & Lee, H. (2014). GMAD: Graph-based Malware Activity Detection by DNS traffic analysis. Computer Communications, 49, 33-47.

[48] Liu, X., Fang, C., & Xiao, D. (2011). Intrusion diagnosis and prediction with expert system. Security and Communica-tion Networks, 4(12), 1483-1494.

[49] Ma, W., Duan, P., Liu, S., Gu, G., & Liu, J. C. (2012). Shadow attacks: automatically evading system-call-behavior based malware detection. Journal in Computer Virology, 8(1), 1-13.

[50] Magkos, E., Avlonitis, M., Kotzanikolaou, P., & Stefanidakis, M. (2013). Toward early warning against Internet worms based on critical-sized networks. Security and Communi-cation Networks, 6(1), 78-88.

[51] Masud, M. M., Al-Khateeb, T. M., Hamlen, K. W., Gao, J., Khan, L., Han, J., & Thuraisingham, B. (2011). Cloud-based malware detection for evolving data streams. ACM Transactions on Management Information Systems (TMIS), 2(3), 16.

[52] Martín, A., Menéndez, H. D., & Camacho, D. (2016). MOCDroid: multi-objective evo-lutionary classifier for Android malware detection. Soft Computing, 1-11.

[53] Miao, Q., Liu, J., Cao, Y., & Song, J. (2016). Malware detection using bilayer behavior abstraction and improved one-class support vector machines. International Journal of Information Security, 15(4), 361-379.

[54] Menahem, E., Shabtai, A., Rokach, L., & Elovici, Y. (2009). Improving malware detec-tion by applying multi-inducer ensemble. Computational Statistics & Data Analysis, 53(4), 1483-1494.

[55] Mohaisen, A., Alrawi, O., & Mohaisen, M. (2015). Amal: High-fidelity, behavior-based automated malware analysis and classification. Computers & Security, 52, 251-266.

[56] Monga, R., & Karlapalem, K. (2008, May). MASFMMS: Multi Agent Systems Frame-work for Malware Modeling and Simulation. In International Workshop on Multi-Agent Systems and Agent-Based Simulation (pp. 97-109). Springer Berlin Heidelberg.

[57] Narudin, F. A., Feizollah, A., Anuar, N. B., & Gani, A. (2016). Evaluation of machine learning classifiers for mobile malware detection. Soft Computing, 20(1), 343-357.

[58] Neugschwandtner, M., Comparetti, P. M., Jacob, G., & Kruegel, C. (2011, December). Forecast: skimming off the malware cream. In Proceedings of the 27th Annual Computer Security Applications Conference (pp. 11-20). ACM.

[59] Ni, M., Li, T., Li, Q., Zhang, H., & Ye, Y. (2016). FindMal: A file-to-file social network basedmalware detection framework. Knowledge-Based Systems, 112, 142-151.

[60] Portokalidis, G., & Bos, H. (2007). SweetBait: Zero-hour worm detection and contain-ment using low-and high-interaction honeypots. Computer Networks, 51(5), 1256-1274.

[61] Rahbarinia, B., Perdisci, R., & Antonakakis, M. (2016). Efficient and Accurate Behavior-Based Tracking of Malware-Control Domains in Large ISP Networks. ACM Transactions on Privacy and Security (TOPS), 19(2), 4.

[62] Raphel, J., & Vinod, P. (2016). Heterogeneous Opcode Space for Metamorphic Malware Detection. Arabian Journal for Science and Engineering, 1-22.

[63] Seo, S. H., Gupta, A., Sallam, A. M., Bertino, E., & Yim, K. (2014). Detecting mobile malware threats to homeland security through static analysis. Journal of Network and Computer Applications, 38, 43-53.

[64] Santos, I., Brezo, F., Ugarte-Pedrero, X., & Bringas, P. G. (2013). Opcode sequences as representation of executables for data-mining-based unknown malware detection. Infor-mation Sciences, 231, 64-82.

[65] Saxe, J., & Berlin, K. (2015, October). Deep neural network based malware detection using two dimensional binary program features. In Malicious and Unwanted Software (MALWARE), 2015 10th International Conference on (pp. 11-20). IEEE.

[66] Sexton, J., Storlie, C., & Anderson, B. (2016). Subroutine based detection of APT mal-ware. Journal of Computer Virology and Hacking Techniques, 12(4), 225-233.

[67] Shahzad, R. K., & Lavesson, N. (2012, August). Veto-based malware detection. In Availability, Reliability and Security (ARES), 2012 Seventh International Conference on (pp. 47-54). IEEE.

[68] Shabtai, A., Tenenboim-Chekina, L., Mimran, D., Rokach, L., Shapira, B., & Elovici, Y. (2014). Mobile malware detection through analysis of deviations in application network behavior. Computers & Security, 43, 1-18.

[69] Sheen, S., Anitha, R., & Natarajan, V. (2015). Android based malware detection using a multifeature collaborative decision fusion approach. Neurocomputing, 151, 905-912.

[70] Sheen, S., Anitha, R., & Sirisha, P. (2013). Malware detection by pruning of parallel ensembles using harmony search. Pattern Recognition Letters, 34(14), 1679-1686.

[71] Singh, T., Di Troia, F., Corrado, V. A., Austin, T. H., & Stamp, M. (2016). Support vector machines and malware detection. Journal of Computer Virology and Hacking Techniques, 12(4), 203-212.

[72] Suarez-Tangil, G., Tapiador, J. E., Peris-Lopez, P., & Blasco, J. (2014). Dendroid: A text mining approach to analyzing and classifying code structures in android malware families. Expert Systems with Applications, 41(4), 1104-1117.
[73] Talha, K. A., Alper, D. I., & Aydin, C. (2015). APK Auditor: Permission-based Android malware detection system. Digital Investigation, 13, 1-14.
[74] Tafazzoli, T., & Sadeghiyan, B. (2015). A stochastic model for the size of worm origin. Security and Communication Networks.
[75] Tong, F., & Yan, Z. (2016). A hybrid approach of mobile malware detection in Android. Journal of Parallel and Distributed Computing.
[76] Truong, D. T., & Cheng, G. (2016). Detecting domain-flux botnet based on DNS traffic features in managed network. Security and Communication Networks, 9(14), 2338-2347.
[77] Vatamanu, C., Gavriluț, D., & Benchea, R. M. (2013). Building a practical and reliable classifier for malware detection. Journal of Computer Virology and Hacking Techniques, 9(4), 205-214.
[78] [Wang, P., & Wang, Y. S. (2015). Malware behavioural detection and vaccine develop-ment by using a support vector model classifier. Journal of Computer and System Scienc-es, 81(6), 1012-1026.
[79] Walenstein, A., Hefner, D. J., & Wichers, J. (2010). Header information in malware fam-ilies and impact on automated classifiers. In Malicious and Unwanted Software (MALWARE), 2010 5th International Conference on (pp. 15-22). IEEE.
[80] Wahid, A., Leckie, C., & Zhou, C. (2011). Self-similar characteristics of network intru-sion attempts and the implications for predictability. Concurrency and Computation: Practice and Experience, 23(12), 1367-1385.
[81] Webb, S., Caverlee, J., & Pu, C. (2008). Predicting web spam with HTTP session infor-mation. In Proceedings of the 17th ACM conference on Information and knowledge man-agement (pp. 339-348). ACM.
[82] Wu, S., Wang, P., Li, X., & Zhang, Y. (2016). Effective detection of android malware based on the usage of data flow APIs and machine learning. Information and Software Technology, 75, 17-25.
[83] Xiao, X., Xiao, X., Jiang, Y., Liu, X., & Ye, R. (2016). Identifying Android malware with system call co-occurrence matrices. Transactions on Emerging Telecommunications Technologies.
[84] Xu, Z., Zhang, J., Gu, G., & Lin, Z. (2014, September). GOLDENEYE: Efficiently and Effectively Unveiling Malware's Targeted Environment. In International Workshop on Recent Advances in Intrusion Detection (pp. 22-45). Springer International Publishing.
[85] Ye, Y., Li, T., Jiang, Q., & Wang, Y. (2010). CIMDS: adapting postprocessing techniques of associative classification for malware detection. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 40(3), 298-307.
[86] Ye, Y., Li, T., Huang, K., Jiang, Q., & Chen, Y. (2010). Hierarchical associative classifier (HAC) for malware detection from the large and imbalanced gray list. Journal of Intelligent Information Systems, 35(1), 1-20.
[87] Ye, Y., Wang, D., Li, T., Ye, D., & Jiang, Q. (2008). An intelligent PE-malware detection system based on association mining. Journal in computer virology, 4(4), 323-334.
[88] Ye, Y., Chen, L., Wang, D., Li, T., Jiang, Q., & Zhao, M. (2009). SBMDS: an interpretable string based malware detection system using SVM ensemble with bagging. Journal in computer virology, 5(4), 283-293.
[89] Yu, J., Huang, Q., & Yian, C. (2016). DroidScreening: a practical framework for re-al-world Android malware analysis. Security and Communication Networks.
[90] Zhang, H., Yao, D. D., Ramakrishnan, N., & Zhang, Z. (2016). Causality reasoning about network events for detecting stealthy malware activities. computers & security, 58, 180-198.
[91] Zhou, Y., & Inge, W. M. (2008, October). Malware detection using adaptive data com-pression. In Proceedings of the 1st ACM workshop on Workshop on AISec (pp. 53-60). ACM.

Suriayati Chuprat is Senior Lecturer at Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia. She received BSc and MSc degrees in Computer Science (Software Engineering) and PhD in Mathematics from Universiti Teknologi Malaysia. In part of her PhD research, she was attached to the University of North Carolina, USA. She did a postdoctoral program at the University of York, UK. Her research interest includes Software Engineering, Algorithms and Scheduling Theories, Real-time Systems and Parallel Computing. She currently active in research and development projects in the area of Big Data Analytics and Information Security. She is a member of the ACM Professional and IEEE Computer Society.

Mohd Naz'ri Mahrin is an Associate Professor at Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia. He received BSc and MSc degrees in Computer Science (Software Engineering) from Universiti Teknologi Malaysia, and the PhD degree in Software Engineering from University of Queensland, Australia. His research focuses on Software Engineering, including Software Measurement, Software Engineering Process, and Software Quality Assurance. He also actively involved in research and development projects in the area of Big Data Analytics and Information Security. He is a member of the ACM Professional and IEEE Computer Society.

Syahid Anuar is a senior lecturer at Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia. He received BSc, MSc and Phd degrees in Computer Science (Software Engineering) from Universiti Teknologi Malaysia. His research focuses on data analytic, including big data analytic, machine learning, artificial neural network and pattern recognition. He also actively involved in research and development projects in the area of Big Data Analytics and Information Security.

Dr. Aswami Ariffin (Dr.AA) is a digital forensics scientist with vast experience in big data, security assurance, threat intelligence, incident response and digital forensics investigation with various law enforcement agencies and provided expert testimonies in court. Due to his immense contribution in cyber security, Dr.AA was awarded ISLA (Information Security Leadership Award) in 2009 by (ISC)2 USA including commendation letter from the Attorney General's Chambers Malaysia and a certificate of appreciation from the Royal Malaysia Police in 2010. He had alsobeen appointed as an expert referral by the New South Wales Police, Australia and currently a member of Interpol Digital Forensics Expert Group. Dr.AA is active in research and one of his papers was accepted for publication in the Advances in Digital Forensics IX. He has secured several large research and development funds from the government and highly experience in software engineering to develop digital forensics, cyber security and big data threat intelligence analytics dashboard capabilities in Malaysia. Currently, Dr.AA is Senior Vice President of Cyber Security Responsive Services at CyberSecurity Malaysia. He provides input on strategic direction, technical leadership and marketing strategy for Malaysia Computer Emergency Response Team (MyCERT), Digital Forensics Department and Secure Technology Services. Dr.AA is regularly consulted the government, industries, universities, communities and media on cyber security issues, strategies and operation including invitation as keynote speaker in conferences.

Mohammad Zaharudin Ahmad Darus is currently working with Cybersecurity Malaysia, a Government Agency, that responsible to create and sustain a safer cyberspace nationally. He holds a position as Manager within the Cyber Forensic Value Innovation Unit under the Digital Forensics Department. The unit function is to bring up and to create new value to the current Digital Forensics services, products as well as conducting short research in Emerging Technologies. Zaharudin is passionate about Digital Forensics especially in CCTV and video technology. He has 9 years experiences in conducting cases mostly in CCTV and video Forensics. He also appeared in court as expert witness for more than 10 times. As to date, he possesses 2 professional certificates, which are EnCase Certified Examiner (EnCase) and Certified Biometric Professional (CBP). He also had 6 years experiences during his previous tenures with Telekom Research and Development, conducted research in the area of Biometric application, image processing, digital watermarking and Artificial Intelligent.

Fakhrul Afiq Abd Aziz been working with CyberSecurity Malaysia since 2016. He is experienced in Malware and Big Data Analytics. He also involves in embedded development to produce forensics tools and software. One of the tools, x-Forensik toolkit won Malaysia Innovation Product Award (MIPA) in 2017. Other than that, he serves as a member of project management team to manage research projects related to cyber security. He is certified in iVe Vehicle System Forencisc by Berla and also in Certified Ethical Hacker (CEH) by Ec-Council.

Mohd Zabri Adil Talib plays major role of ensuring the competent and efficient overall operations of CyberSecurity Malaysia (CSM) digital forensics services. As the Head of CSM Digital Forensics services, he has vast experiences in handling computer crimes, computer-related crimes forensics examination for various law enforcement agencies in Malaysia. This is also including e-discovery cases for civil claim. Zabri has completed the digital evidence case investigation circle by testifying in Malaysia Intellectual Property Court, Magistrate Court, Session Court, High Court and Royal Commission of Inquiry including many high-profile cases. He is the ISC2 ISLA 2011 Managerial Professional for Information Security Project Showcased Honoree for Exemplary Leadership in and Dedication to Enhancing the Information Security Workforce. He is also an ASCLD/LAB certified Assessor.