ISSN 2278-3091

Volume 9 No.2, March -April 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse103922020.pdf

https://doi.org/10.30534/ijatcse/2020/103922020

Information Technology Security Infrastructure Malware Detector System

[1]Jaevier A. Villanueva, [2]Luisito L. Lacatan, [3]Albert A. Vinluan

[1] AMA University, Philippines, jaevier.villanueva@gmail.com

[2] AMA University, Philippines, lllacatan@amaes.edu.ph

[3] AMA University, Philippines, albert.vinluan@gmail.com

ABSTRACT

Technology had been automated since the advent of the digital era. It paved the way to ease technical outages through manual resolution. While the globe is overwhelmingly dominated by technology, its growth remains dynamic. This study aims at improving the efficiency of existing IT security infrastructure processes and creating a malware detection system. The researchers gathered the data to IT staff and IT experts through face-to-face interviews, observations and questionnaires. Questions are about malware incident detection, data protection and reliability of the proposed program. The IT department wants a more effective program to manage these dilemmas, based on the study conducted. To counter this, the researcher built a system to upgrade their regular malware detector processing. The system would also be important not only for the client but also for other IT firms. Malware detector is handled properly and in a timely manner which reduces customer business costs. This study would also benefit not only the companies but also the clients.

Key words: Infrastructure, malware, questionnaire, security

1. INTRODUCTION

In earlier years an intersection reason of circumstances has forced IT technologies to end up engineered in more affiliation methods so that they can be more adaptable and tailored to the business centers of a couple of different company affiliations.

Division of IT(Information Technology) are the most dynamic aspects of associations in which Information Innovation Infrastructure Library (IIIL),a part of these division is seen as the most commonly used IT system. This norm is required to help the relationship units in the implementation of quality-based systems, with the concluding goal to improve the method for IT organizations. On the same plane, ITIL itself comprises five areas, specifically: organization method, organization diagram, organization operation, organization move and constant organization change. Every of these sections includes a few devices. The layout of each of ITIL techniques goes hand in hand with description, intent of interest, Critical Factors and Key Indicators. One of the fundamental ITIL procedures is Problem Management process.

[1] Malicious software is known as malware. Computer malware is a program that, when executed, reproduces and infects a computer, poses a threat to the integrity of the system [1],[2].

We depend on anti-scanners for a long time to detect malicious and defend our computer against malicious attacks. Throughout the years a few anti-malware scanners have been produced with varying performance levels. Currently solitary scanner was sufficient to distinguish a large portion of the malicious file. Be that as it may, as time passes by, the malware designer has built up their skills and malware database has grown and reproduce to such a degree, that no solitary malicious detector can defend us from them all. Analysts have discovered that the merging of the intensity of numerous of anti-malicious drastically improves recognition precision and productivity contrasted with any single anti malicious for instance[3],[4].Therefore, independent networked anti - malicious and resources such as [5],[6], etc. was established to resolve this task.

Despite the fact that the numerous anti malicious administrations and assets available, a large portion of them are utilized for illuminate or as a subsequent purposes source. Not any of them give a detailed choice on anyhow a stated example is malicious or not. In place of, they go about as a total of information and describe the individual's outcomes that every anti malicious returned. The usefulness of building the solution will not stop with the individual stop-users. As a result, emphasized by[7],[8],[9]. The effect of the insufficiency of agreement, continuity and validity among many present solution reached after consideration on anti-malicious is rangy, Influencing not just scholastic scientists and merchants of anti-malicious, yet additionally IT experts who supervise security tasks in big business systems and government organizations answerable foe shielding basic framework from digital assaults. In this manner, be on control and decisive datasets are the basic needs to make a framework to detect and the advancement that continuous assault reaction framework against current system-based assaults.

A malicious identification framework will require huge data about the dataset for preparing and testing purposes to show up at right choices.

In this article, we come up with a (1) model to distinguish output of malicious file based on the dataset available. (2) To manipulate malicious system to acquire highest veracity that let us find out the best attributes that enables the model to predict veracity measure. (3) To acquaint method which help to lessen construction that meet time constraint by recognizing a close to ideal subset.

2.LITERATURE REVIEW

We have briefly identified and addressed past and current work and system which commercially available involving malware detector in this chapter. It includes work on the model and design of detector, collaborative malware identification, and profit-oriented anti-malicious with various combined look through mechanism.

[10] In experiments, a single anti-malware application was shown to be inadequate to identify all current malware on a computer. While their findings have shown, in a limited way, that unite multiple anti-malicious system execute preferable review and false negative levels.[11] Based on the evidence shows that the recognition capacity of AVs are substantially progressed, and their results likewise shows that none of them has accomplished a steady identification score.[12] Make known a new paradigm for user of malware detection focused on delivering the anti-malicious system as a cloud-based application. Their model utilized different, mixed equal identification mechanism; a strategy alluded to as protection.[13] Introduced a similar cloud-based filtering and nearby examining investigation and proposed an exchange off to utilize the best pieces of both.

[1],[14],[15],[16] A disruptive process which grouping of malware. A deprecated run of architecture has certainly damaged the platform. Utilizing the allotted unsafe program that are performed on the contaminated machine and spreading activity are regularly requested. Malware occur in not at all like gather and list.

[17],[18] Clustering is a ubiquitous method for the analysis of data, implemented in virtually every discipline. The aim of the clustering is to find patterns in the data that you are working on which can be used to group data items.

[17],[19] [1] Random Forest (RF) is impressive compared to other known AI calculations. This involves almost no data preparation and simulation but typically results in an incorrect outcome— random selection of decision trees, making predictions accurate.

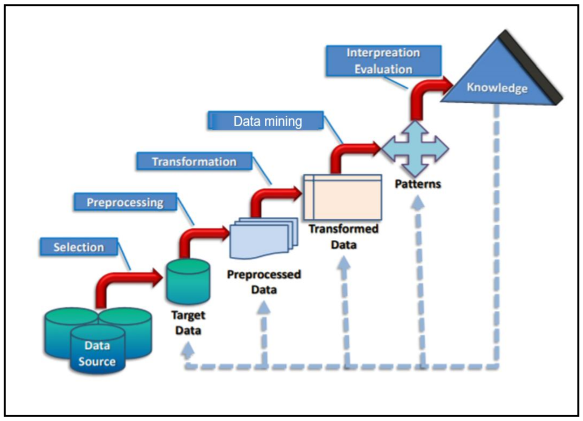


Figure 1: Knowledge Discovery Database (KDD) Process [20]

The process illustrates in Fig. 1 The term Knowledge Discovery Database alludes to the expansive procedure of discovering Knowledge in data and stress the "significant level" utilization of specific data mining techniques. The process are typically: Selection, pre-processing, transformation, data mining, and evaluation [1], [20],[21].

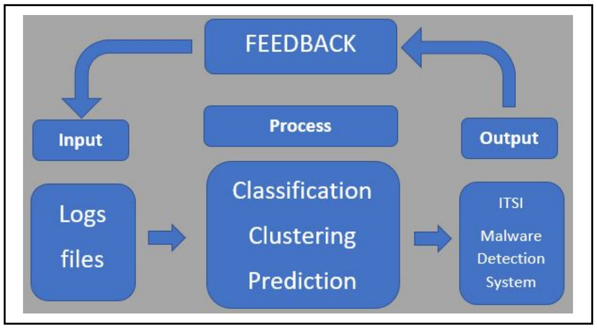


Figure 2: Conceptual framework

Fig 2 provides a graphical view on the conceptual structure of the proposed system. This process will manage files issues within the organization infrastructure. The system will classify depending on its priority if it is high or low.

2. METHODOLOGY

The Related research work was conducted to explain the nature of the phenomena under review and to establish the necessary parameters and relationship structures for the study. To fill in as the premise of the hypothetical part of this paper, these related writing and studies were gathered and arranged.

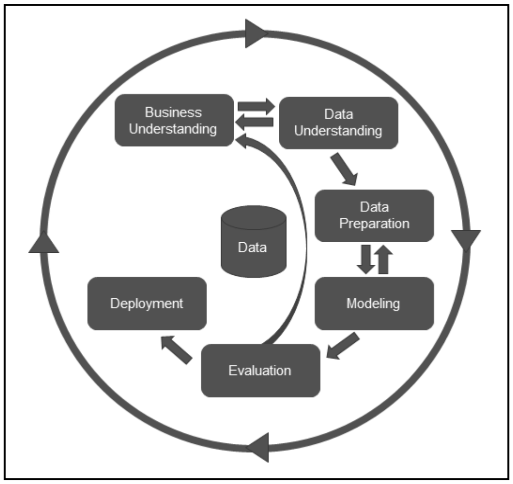


Figure 3: CRIPS-DM Process [21]

3.1 Methodology

[21][22] Fig. 3 pronounce as an instrument to perceive better and quicker outcomes from data mining. It is an all-around demonstration organized thought in arranging a data mining venture. The CRISP-DM consists of six (6) phases:

3.1.1 Business Understanding

Data classification has two face. To begin with, the preparation sets of information were controlled by breaking down a lot of preparing dataset instances until a data model was fabricated which portrays a predetermined set of class marks. Besides, the procedure models utilizing Random forest decision Algorithm, and K-means were applied to the test data to decide the order, figure and bunch pace of the model.

3.1.2 Data Understanding

Considering the pertaining relevant behavior for the identification process is an important issue. The selection of incorporated features and patterns of behavior is based on two parameters: the patterns of behavior need to be relevant for classifying, forecasting and clustering in detecting malicious based on the generated model and that the system can gather information about patterns must be as high as possible.

3.1.3 Data Preparation

The unprocessed data extracted from the database were cleanse by extracting only the important attributes using SQL scripts to avoid invalid, unique and mislaid values. The extracted data were transformed in an MS Excel file (CSV) and then set aside to the way that can be accept by data mining software. These files occur together without conflict with the WEKA, Orange and R Programming tools in structuring the model.

3.1.4 Modeling

In this phase, different data mining classification, forecast and cluster techniques were tested in order to infer the prediction of malware. The datasets were occurred together into data mining tools software which implements different classification and auto regression algorithms. This study compared the accuracy of several classification, time series and cluster algorithms utilizing the point by point classification table outcomes. The outcome for every data mining model will be assessed so as to decide the better data model that will be incorporated in the prototype of IT Security Infrastructure Malware Detector System.

3.1.5 Evaluation

The model assessment is a necessary piece of the model advancement process. It helps in finding the best model that speaks to the information and how well the picked model will function later. The most widely recognized approach to assess a specific model is to confirm their exhibitions on the test datasets. Assessment of the model can be recognized by experimentally acquiring the number of right forecasts to the absolute number of expectations. Examination methods of the determined model will outline its exactness and it is an iterative procedure in which all contending models are assessed dependent on precision. In the event that the exactness of the model is excessively low, the model is considered to underfit, when the precision is excessively high, the model is overfit.

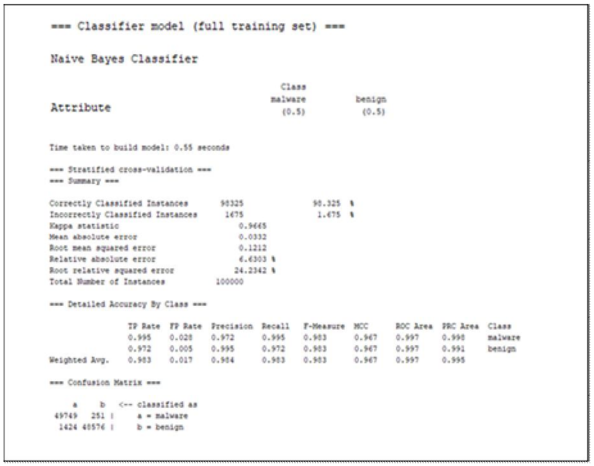
3.1.6 Deployment

The knowledge produced must be useful to the user. To effectively use the created models, the users carry out deployment steps like generating reports or executing iterative data mining process.

3.2 Algorithm performance

Three data mining methods were used in this study namely: WEKA, Orange and R Programming.

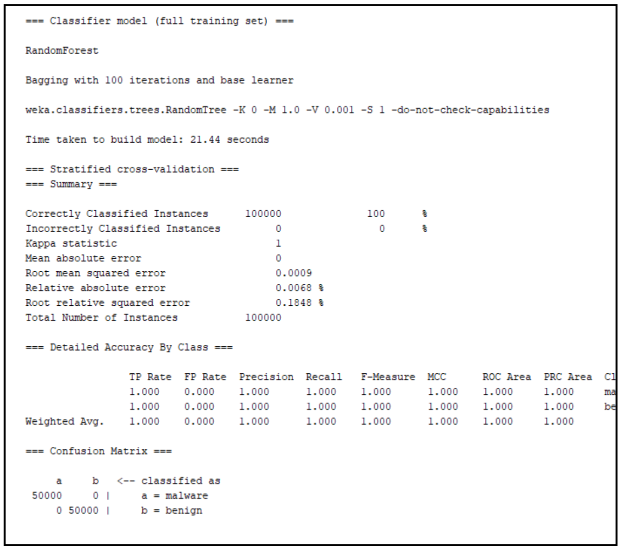
3.2.1 WEKA Data mining



```
=== Classifier model (full training set) ===

Naive Bayes Classifier

                                    Class
Attribute                         malware          benign
                                    (0.5)           (0.5)


Time taken to build model: 0.55 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       98325               98.325  %
Incorrectly Classified Instances      1675                1.675  %
Kappa statistic                          0.9665
Mean absolute error                      0.0332
Root mean squared error                  0.1212
Relative absolute error                  6.6303 %
Root relative squared error             24.2342 %
Total Number of Instances           100000

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.995    0.028    0.972      0.995    0.983      0.967    0.997     0.998     malware
                 0.972    0.005    0.995      0.972    0.983      0.967    0.997     0.991     benign
Weighted Avg.    0.983    0.017    0.984      0.983    0.983      0.967    0.997     0.995

=== Confusion Matrix ===

     a     b   <-- classified as
 49749   251 |     a = malware
  1424 48576 |     b = benign
```

Figure 4: Naïve Bayes Algorithm

Fig. 4 shows that 98,325 instances were uploaded and tested using Naïve Bayes Algorithm correctly classified 98.325 % with 1.675 % incorrectly classified instances.

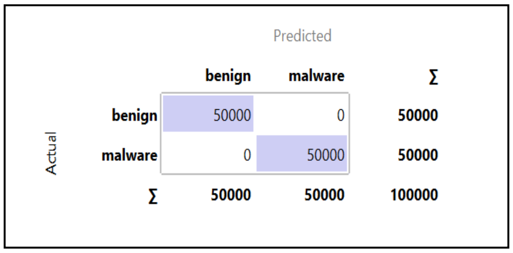


```
=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner

weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Time taken to build model: 21.44 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      100000              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0.0009
Relative absolute error                  0.0068 %
Root relative squared error              0.1848 %
Total Number of Instances           100000

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Cl
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     ma
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     be
Weighted Avg.    1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

     a     b   <-- classified as
 50000     0 |     a = malware
     0 50000 |     b = benign
```

Figure 5: Random Forest Algorithm

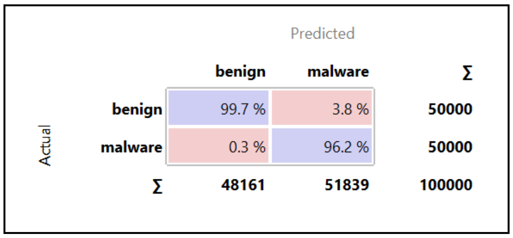
Fig. 5 shows that 100,000 instances were uploaded and tested using Random Forest Algorithm correctly classfied 100% with 0.0% incorrectly classified intances.

3.2.2 Orange Data mining



		Predicted		
		benign	malware	∑
Actual	benign	50000	0	50000
	malware	0	50000	50000
	∑	50000	50000	100000

Figure 6: RF Predicted

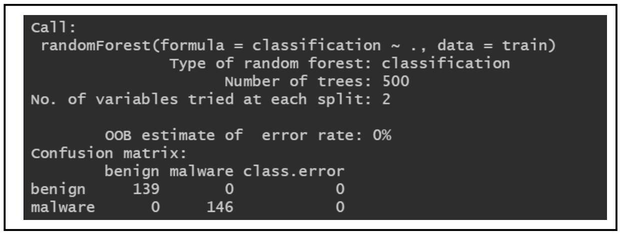


		Predicted		
		benign	malware	Σ
Actual	benign	99.7 %	3.8 %	50000
	malware	0.3 %	96.2 %	50000
	Σ	48161	51839	100000

Figure 7: NB Predicted

Fig. 6 and 7 revealed that (RF) Random Forests foresee 100 percent (%) accuracy contrasted with (NB) Naïve Bayes Algorithm which misclassified 0.30 percent instances. This convey that the Orange Data Mining software suggests Random Forests Decision Algorithm as it is further accurate apart from Naïve Bayes Algorithm.

3.2.3 R- Programming



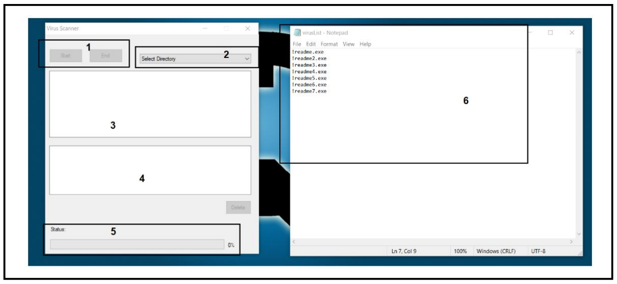
```
Call:
 randomForest(formula = classification ~ ., data = train)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 0%
Confusion matrix:
        benign malware class.error
benign     139       0           0
malware      0     146           0
```

Figure 8: Error Rate using Training Data

Fig. 8 . illustrates that based on the OOB or out of bag estimate, the error rate of Random Forests Decision Algorithm shows that there is no error or misclassified instances. This clearly shows that the algorithm used is 100% accurate.

3. RESULT AND DISCUSSION

The output of this study is an Information Technology Security Infrastructure Malware Detector System that deals to detect malicious file. As shown in fig. 9. VB.net was utilized to develop the system. ITSI Malware Detector System uses a directory known as malware directory which has lots of cipher from different present malware. At the point when the framework filters the file, it will take a suggestion of that cipher and variance it and the suggestion in their malware word reference, in the matter that the records systematize, thereupon the malware hit on a unit is pronounce. In short, the matter and recognition process changes on the archive of know malware. A hearty malware finder is increasingly equipped for finding malicious cipher time and again cascade by virtual lawbreakers for their benefits. The moment when malign is recognized to exist inframework word reference, the malware identifier makes sure about the PC by hindering all exercises of the malware.



```
!readme.exe
!readme2.exe
!readme3.exe
!readme4.exe
!readme5.exe
!readme6.exe
!readme7.exe
```

Figure 9: Developed ITSI Malware Detector System

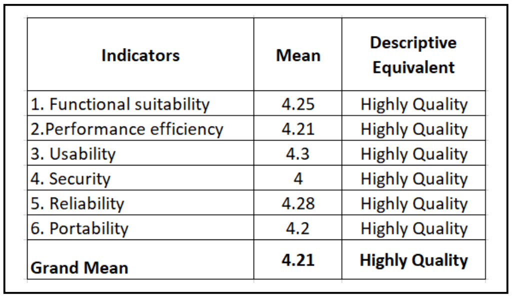
4.1 Malware Code Reference

Regularly, these malicious file appear to originate from genuine sources and their malignant goals remain covered up. The malware code of reference assumes the key job in distinguishing the noxious file. It could be anything, for example, the worm, or malwaree and this could be one of the crucial motivations to why that a large number of the present anti - malicious depends on single malware strain.

Rather, the mark offers an unbelievable chance to cross-check against a wide scope of effectively-recognized documents that contain code into which the noxious highlights are inserted. This method is known as heuristics.

In the wake of distinguishing the malicious file on the PC, the entire rundown of affectedrecords can be short-recorded and for the most part the quantity of infected documents is massive.

Table 1: Summary of Result



Indicators	Mean	Descriptive Equivalent
1. Functional suitability	4.25	Highly Quality
2.Performance efficiency	4.21	Highly Quality
3. Usability	4.3	Highly Quality
4. Security	4	Highly Quality
5. Reliability	4.28	Highly Quality
6. Portability	4.2	Highly Quality
Grand Mean	**4.21**	**Highly Quality**

The following Table 1 sums up the content of the built application's software. The respondents strongly agreed that the application was of high quality, as shown by the mean 4.24 average. Overall, this implies that respondents generally felt that the system built was practical, effective, accessible, safe, reliable and portable.

4. CONCLUSION

The developed system may be used as an effective software for Inforamtion Technology Security Infrastructure Platform Malware Detector. The final testing, the respondents are already positive and satisfied with the reliability, funcionality of the system. The researcher found out that this system met the intention of the investigation. The procedures and approach on the proposed system were evaluated "high Quality". The development of the Inforamtion Technology Security Infrastructure Malware Detector system was succesful.

ACKNOWLEDGEMENT

We should thank the unknown observers and the individuals from the School of the Graduate Studies at AMA University, Philippines, for their input on this work.

REFERENCES

[1] J. A. Villanueva, R. Juanatas, and L. L. Lacatan, **"Malware predictor using machine learning techniques,"** *Test Eng. Manag.*, vol. 82, no. March, pp. 5665–5674, 2020.

[2] M. Divya, M. Monika, and N. Kanimozhi, **"Detecting Malicious Facebook Application using Digital India Scheme,"** *Int. J. Trend Sci. Res. Dev.*, vol. Volume-2, no. Issue-3, pp. 555–558, 2018. https://doi.org/10.31142/ijtsrd10964

[3] B. Maschinen, A. Investition, G. Beschaffungen, B. Ersatzbeschaffungen, and S. Mittelherkunft, **"Cyber Security Challenges,"** vol. 4, no. 06 1, pp. 10–15.

[4] R. Luh, **"Advanced Threat Intelligence: Interpretation of Anomalous Behavior in Ubiquitous Kernel Processes,"** 2019. https://doi.org/10.1016/j.cose.2019.03.015

[5] **"Free Online Virus Scan | Trend Micro."** [Online]. Available: https://www.trendmicro.com/en_ph/forHome/products/housecall.html. [Accessed: 17-Apr-2020].

[6] **"Kaggle: Your Home for Data Science."** [Online]. Available: https://www.kaggle.com/. [Accessed: 17-Apr-2020].

[7] A. A. Selcuk, F. Orhan, and B. Batur, **"Undecidable problems in malware analysis,"** *2017 12th Int. Conf. Internet Technol. Secur. Trans. ICITST 2017*, pp. 494–497, 2018. https://doi.org/10.23919/ICITST.2017.8356458

[8] A. Olawale Surajudeen, **"Malware Detection, Supportive Software Agents and Its Classification Schemes,"** *Int. J. Netw. Secur. Its Appl.*, vol. 4, no. 6, pp. 33–49, 2012. https://doi.org/10.5121/ijnsa.2012.4603

[9] T. Atkinson, **"Hunting ELFs: An investigation into Android malware detection,"** no. February, pp. 1–3, 2017.

[10] A. Moser, C. Kruegel, and E. Kirda, **"Limits of static analysis for malware detection,"** *Proc. - Annu. Comput. Secur. Appl. Conf. ACSAC*, pp. 421–430, 2007. https://doi.org/10.1109/ACSAC.2007.21

[11] H. Deylami and R. C. Muniyandi, **"Taxonomy of Malware Detection Techniques :,"** 2016.

[12] R. Sagar, R. Jhaveri, and C. Borrego, **"Applications in security and evasions in machine learning: A survey,"** *Electron.*, vol. 9, no. 1, pp. 1–42, 2020. https://doi.org/10.3390/electronics9010097

[13] **"Advanced threat predictions for 2020 | Securelist."** [Online]. Available: https://securelist.com/advanced-threat-predictions-for-2020/95055/. [Accessed: 17-Apr-2020].

[14] O. Aslan and R. Samet, **"A Comprehensive Review on Malware Detection Approaches,"** *IEEE Access*, vol. 8, pp. 6249–6271, 2020.

[15] Y. Park, **"Efficient Validation of Control Flow Integrity for Enhancing Computer System Security,"** ***Control***, 2010.

[16] S. K. Debray, K. P. Coogan, and G. M. Townsend, **"On the Semantics of Self-Unpacking Malware Code,"** p. 13, 2008.

[17] L. C. Hileman, E. M. Kramer, and D. A. Baum, **"Differential regulation of symmetry genes and the evolution of floral morphologies,"** *Proc. Natl. Acad. Sci.*, vol. 100, no. 22, pp. 12814–12819, 2003. https://doi.org/10.1073/pnas.1835725100

[18] N. M. Alotaibi, M. Abdullah, and H. Mosli, **"Agent-based big data mining,"** *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 1, pp. 245–252, 2019. https://doi.org/10.30534/ijatcse/2019/4481.12019

[19] O. Khattab, "International Journal of Advanced Trends in Computer Science and Engineering Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse20852019.pdf **A Comprehensive Survey on Vertical Handover Security Attacks during Execution Phase,"** vol. 8, no. 5, pp. 1965–1968, 2020. https://doi.org/10.30534/ijatcse/2019/20852019

[20] G. M. Pangilinan, M. A. F. Quioc, and L. L. Lacatan, **"Integrating artificial neural network and smartbot on the development of an E-learning platform,"** *Test Eng. Manag.*, vol. 82, no. February, pp. 5570–5575, 2020.

[21] A. Elacio, **"Digital Transformation in Managing Employee Retention using Agile and C4 . 5 Algorithm,"** no. February, 2019.

[22] H. Nagashima and Y. Kato, **"APREP-DM: A Framework for Automating the Pre-Processing of a Sensor Data Analysis based on CRISP-DM,"** *2019 IEEE Int. Conf. Pervasive Comput. Commun. Work. PerCom Work. 2019*, pp. 555–560, 2019. https://doi.org/10.1109/PERCOMW.2019.8730785