International Journal of Innovative Computing, Information and Control
Volume **14**, Number **1**, February **2018**

ICIC International ©2018 ISSN 1349-4198
pp. **379–386**

ANALYST INTUITION INSPIRED NEURAL NETWORK BASED CYBER SECURITY ANOMALY DETECTION

Teik-Toe Teoh, Yok-Yen Nguwi, Yuval Elovici, Wai-Loong Ng and Soon-Yao Thiang

ST Electronics-SUTD Cyber Security Laboratory
Singapore University of Technology and Design
8 Somapah Road, Singapore 487372, Singapore
{ teiktoe_teoh; yuval_elovici }@sutd.edu.sg

Received March 2017; revised July 2017

Abstract. *Internet revolution has brought advancement to the world's economy, business, technology and communication. It also brings forth the risk of cyber-attack penetration. It is a challenge to detect cyber-attack accurately and timely. In this work, we adopted a large network dataset containing malware attack data and trained it to recognize a cyber security attack to establish an expert system. The characteristics of attacker's IP addresses can be extracted from our integrated dataset for statistical data extraction. The cyber security expert annotates the weight of each attribute and constructs a scoring system through log history annotation. We adopted a special semi supervise method to classify cyber security log into attack, unsure and no attack by first splitting the data into 3 clusters using fuzzy K-means (FKM), then manually label a small data (analyst intuition) and finally train the neural network classifier multi-layer perceptron (MLP) based on the manually labelled data. By doing so, our results were more encouraging as compared to finding anomalies within cyber security logs without analyst intuition's labelling. The latter generally creates a large amount of false detections.*

Keywords: Cyber security, Neural network, Big data, High velocity, Multi-layer perceptron (MLP)

1. **Introduction.** Internet revolution which started since mid-1990s, has transformed the world's communication infrastructure. It has positively impacted the world's economy, business, technology, and human communication to bring people closer. Unfortunately, it also presents opportunity and loop holes for unintended cyber-attacks at individual, organizational, and political level. Two recent attacks at large scales have impacted the society negatively amounting to huge monetary losses. "Wannacry" (or WannaCrypt) [1] is a cyber-attack happened in May 2017 targeting Microsoft Windows system that has affected more than 230,000 computers in 48 hours to over 150 countries. The ransomware demanded a payment to unlock the infected system. WannaCry caused ambulances to be diverted, shutting down non-emergency services, nabbed machines at Telefonica in Spain, and affected airlines and ministry [2]. The other cyber-attack "Petya" Crypto Ransomware first made its debut in March 2016. It crippled many organizations across Ukraine, US, Russia and other countries. "Petya" is a self-replicating worm that can build a list of target machines and spread by itself [3].

The above cases are just a tip of an iceberg. The breadth and depth of cyber-attacks today are too numerous to name and measure. It is a concerning problem that requires in-time and ahead of time prevention and mitigation strategy. This explains the low success rate of in-time prevention for cyber-attacks. In this work, we attempt to develop an expert system that provides quick response to detect anomaly amongst network log files. The system tags on the expertise of cyber security experts and allows them to input suitable weights for different attributes. The cyber security experts also contribute to the scoring system based on the keywords in log file. We then adopt fuzzy K-means (FKM) algorithm to create clusters of attackers and non-attackers to segregate the attack-related traffic from the network dataset.

In this study, we collected a total of 36 Gigabytes of data. The total amount of Malware instances is about 60 cases. We apply data mining techniques to studying the statistical data obtained from the integrated dataset. These analytics help to identify attack related traffic from normal traffic as well as extracting attack patterns. The fuzzy K-means (FKM) clustering algorithm was performed to create attacker and non-attacker clusters on the time-related and connection-related data obtained from the integrated dataset. As there are 36 million log files and 4 are with attack information, we have an unbalanced dataset. Hence, we picked a computer which had 4 attacks and 1047 non-attack instances. With inputs of analyst intuition, we create labels for another 47 attack logs, which is then used to train the system.

The FKM algorithm created three clusters in total: (i) cluster-1 consists of no attackers, (ii) cluster-2 consists of uncertain number of attackers, and (iii) cluster-3 consists of 364 non-attackers. One of the issues in cyber security is that different network security systems and tools generate log files in different format that renders complexity in consolidation. This research demonstrates the integration and analysis of dataset for identifying attack-related traffic that can potentially lead to easier threat detection in cases where attacks occur on multiple platforms.

2. **Related Works.**

Some recent works on anomaly detection were published by Bereziński et al. [4], Kosek et al. [5], and Amer et al. [6,7]. Bereziński et al. [4] adopted an entropy-based approach to detect modern botnet-like malware based on anomalous patterns in network traffic. Kosek et al. [5] proposed an ensemble model anomaly detection method with non-linear regression models and anomaly scores based on correlation analysis. Partially labelled data was used to train the model. Amer et al. [6,7] adopted nearest neighbor, clustering, and support vector machine to detect anomaly.

The conventional methods of providing security measure against cyber-attacks employ tools such as firewalls, authentication tools, and VPNs. However, these mechanisms always have vulnerabilities that are caused by design or implementation flaws [8]. Hence, monitoring systems have been developed but still require human intervention. The research in [8] developed a Minnesota Intrusion Detection System (MINDS), which is a suite of different data mining-based techniques to detect different types of attacks. The MINDS system contains an anomaly detection approach, which is effective in detecting anomalies in network traffic and preventing DoS (denial-of-service) attacks. In this approach, a model is built with normal data and the deviations in the given data are detected using the normal model. The anomaly detection algorithms have advantages over other techniques as they can detect the threats or attacks when deviated from normal usage even if there are no signatures or labelled data. Also, unlike other detection schemes such as misuse of detection scheme, MINDS system does not require any explicitly labelling of training data set. The MINDS system uses an LOF (local outlier factor) algorithm that detects outliers in data by comparing the densities of various regions in the network data.

In 2013, a system called “Beehive” [9] was proposed which works on the problems involved in automatically mining and extracting knowledge from dirty log data received from various security systems involved in an enterprise. This system was evaluated on the data collected over a period of two weeks at EMC and results were compared with Security Operations Center reports, antivirus software alerts and feedback from enterprise security specialists. It was found that Beehive detected malware infections and policy violations that were undetected by these aforementioned security tools. The algorithm was designed based on K-means clustering algorithm, but does not need to specify the number of clusters. Initially, it selects a random vector as the first cluster, identifies the furthest vector away from the initial hub, and reassigns all the vectors to these clusters with the closest hub. Apart from detecting malicious activity on the network, many studies were conducted on detecting exploited systems using honeypots on enterprise networks [10]. Zhang et al. [11] extended the work of [12] by providing machine-learning features that automatically detect VPN account compromises in a university network.

The work in [13] proposes an attack model for detecting APTs that are more sophisticated than worms, Trojan horses and other malwares. This model is flexible enough to work with large dataset and can accommodate any context processing algorithm that is used for threat detection. The attack model uses the concept of attack trees and attack pyramids to develop models of APT threats, using a large-scale distributed computing framework to establish event correlations as well as time correlations. An attack pyramid is nothing but a model of an APT and the detection framework is based on this model. All the areas where the attack evolves such as user plane, network and application plane are represented as the lateral planes in the pyramid. The goal of the attack occupies the top of the pyramid. These planes change based on the environments where the events are recorded. To reach the goal, the attacker explores the vulnerabilities and navigates from one plane to another plane, which makes the attack look like a tree spanned across multiple planes.

The work in [14] demonstrated good use of K-means for identifying DDOS attack with the right threshold value. However, the usage of association algorithms in this context may lead to identifying too many rules and are not always guaranteed to be relevant [15]. Hence, this research uses logistic regression classification algorithm along with fuzzy K-means algorithm which is good for fraud detection and classification of attack traffic from the network traffic. Also, the results obtained from both algorithms are matched against each other and inferences are drawn to ensure the accuracy of the results.

3. Methodology.

We adopted a semi supervised approach of classifying cyber security log into three classes: namely attack, unsure and no attack. The model is depicted in Figure 1. Network data are extracted from McAfee and CheckPoint software. The data is then channelled to ArcSight, a tool for Enterprise Security Management (ESM) and Security Information and Event Management (SIEM). ArcSight collects security log data. The data is then viewed from Graylog software, which is an open source software for log file viewing. Log files are then converted to comma separated format (.csv) for subsequent processing. In this study, we collected 3 days of data that chalks up to 36 million of log files amounting to 36 Gigabytes of data. We extracted 864 data from the 3.6 million log for training and testing. The total amount of Malware instances is about 60 cases. Network traffic data comes in at very high speed resulting in more than 1000 log files being generated every second, and we use batch processing instead of real-time processing. We extract 864 log files, out of which 500 of them do not have attack information.

We examine the statistical data obtained from the integrated dataset which consists of attacks like Malware, Trojan, Passing off, Soft1026, and Virus. The expert system allows cyber security expert to enter their inputs to form the scores. These analytics help identify the attack related traffic from normal traffic as well as extract attack patterns. The fuzzy K-means (FKM) clustering algorithm was performed to create attacker and non-attacker clusters on the time-related and connection-related data obtained from the integrated dataset. The clustering algorithm forms 3 clusters: Strong, Average, and Mild. The

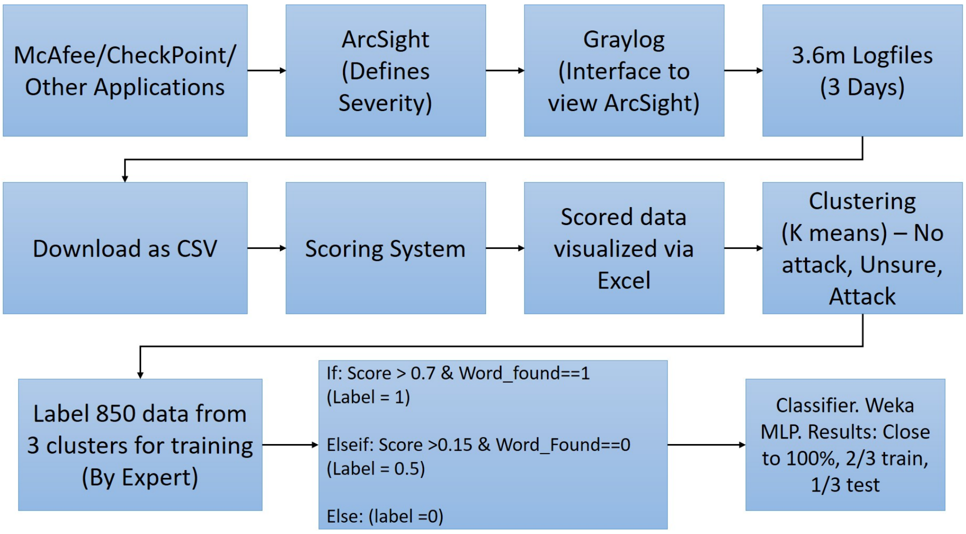
McAfee/CheckPoint/ Other Applications
ArcSight (Defines Severity)
Graylog (Interface to view ArcSight)
3.6m Logfiles (3 Days)
Download as CSV
Scoring System
Scored data visualized via Excel
Clustering (K means) – No attack, Unsure, Attack
Label 850 data from 3 clusters for training (By Expert)
If: Score > 0.7 & Word_found==1 (Label = 1)
Elseif: Score >0.15 & Word_Found==0 (Label = 0.5)
Else: (label =0)
Classifier. Weka MLP. Results: Close to 100%, 2/3 train, 1/3 test

FIGURE 1. Proposed model that detects anomaly from big data

Algorithm 1: Scoring System with Expert Labeling or Analyst Intuition

```
Given Log String = [ ];
Repeat the following until Log String is empty:
  1. Look for words that contain certain keywords like Malware, word found = 1.
  2. For all log, assign a weight to the attribute and a score to the attribute, example
     severity weight is 0.8 and very high is 0.9, then the score will be 0.72
  3. Visualized the data using excel (see Figure 3)
  4. Then we cluster the data into 3 clusters using FKM.
  5. Label the output as attack, unsure, no attack. Labeling rules:
        if score[i-1] > 0.7 and word_found[i-1] == 1:
            label[i-1] = 1
        elif score[i-1] > 0.15 and word_found[i-1] == 0:
            label[i-1] = 0.5
        else:
            label[i-1] = 0
  6. With that, we train our neural network MLP model and test the model against
     the test data with our trained classified.
```

distance measure used is K-means. Prior to clustering, fuzzification was performed. We extract a sample of 864 datum for processing. The system first looks for keywords among data like worm, malware and marks the feature as 1 when keywords are encountered. Expert weightage is then given and forms the scoring. Algorithm 1 outlines the process of expert labelling.

The scoring system involves cyber security expert who manually annotates the log files. This method is following the analyst intuition (AI2) [16]. The original data is not labelled. From the log files, words are given a certain weight and score is created from there. We then use excel to visualize the data and manually label the data into 3 classes: attack, unsure and no attack. From there, we train our model and use the model for classification. The classification system takes in expert view to provide weightages according to the impacts of different attacks as shown in Table 1. Soft1026 is the specific virus that is potentially infectious and hence given a higher weightage.

TABLE 1. Weights for different attacks

Types of Attacks	Weights
Soft1026	0.8
Trojan	0.7
Malware	0.5
Worm	0.4
Virus	0.3
Forced_Off	0.5
Failed_Login	0.4
Severity	0.5
Very_High	0.9
High	0.7
Medium	0.5

Next, we then cluster the data into clusters using fuzzy K-means (FKM). The fuzzy K-means (FKM) clustering algorithm was performed to create attacker and non-attacker clusters on the time-related and connection-related data obtained from the integrated dataset. The data is split into 3 clusters based on K-means algorithm. The 3 clusters are no attack, unsure and attack. We then train the data using multi-layer perception neural network using 2/3 of the data. The remaining 1/3 of the data is used for test.

4. **Experimental Results.** The dataset used in this research was provided by Singapore Technology Engineering. The dataset is for research purposes. The various categories in the dataset include virus, worms, Trojan, etc.

Figure 2 denotes the histogram of data distribution. Analyst intuition method produced 2 key attributes, score and WordFound. The scoring labelled by expert is in the range

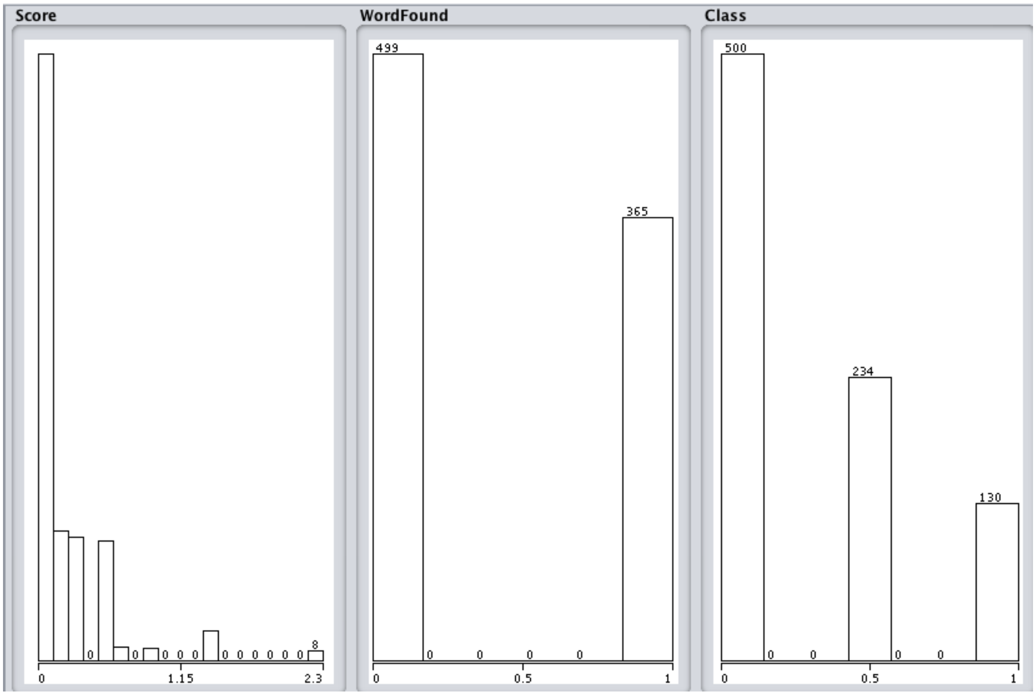

FIGURE 2. Data distribution. Score is labelled by expert, WordFound denotes the presence of keywords, and the last is the number of instances for each class.

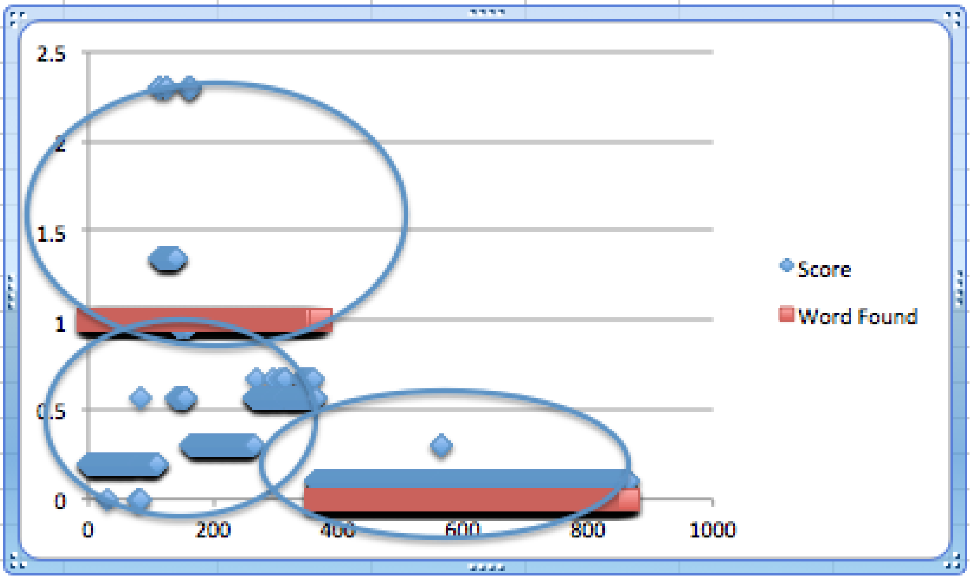

Figure 3. Visualization of data using the score and WordFound attributes and the clusters formed

Table 2. Fuzzy K-means centroid experimental results with full data

Attribute	**Full Data (864)**	**Cluster 0 (365)**	**Cluster 1 (499)**
Score	0.2624	0.4834	0.1008
Type	0.4225	1	0

Table 3. Fuzzy K-means centroid experimental results with training data

Attribute	**Total Data (570)**	**Cluster 0 (226)**	**Cluster 1 (344)**
Score	0.2463	0.4681	0.1006
Type	0.3965	1	0

of 0 to 2.3. WordFound is a discrete data which denotes a 1 when there is a presence of suspected keywords. Three types of clusters could be formed, 0 for clusters without attack, 0.5 for unsure attacks, and 1 for attacks. Figure 3 presents the clusters formed according to the attributes of score and WordFound.

Table 2 illustrates the experimental results of K-means clustering using full data. We have a total 864 data, among which 365 belongs to cluster 0 and 499 belongs to cluster 1. Table 3 presents the experimental results of K-means clustering using training data. The full data is split into 66% of train data versus 34% of test data. These percentages translate to 570 instances of training data as shown in Table 3. The full data centroids results display a significant distance between cluster 0 and cluster 1. The centroid distance in train data is in line with full data. K-means results recommend 2 clusters to be formed. These results help to shape our classification design. We add in one more cluster and make the total classes to be 3 instead of 2 and find the results to be positive.

The sample data is split into 66% as training set, and the remainder as test set. We train the classifier using multi-layer perceptron neural networks. The total computation time is 0.13 seconds. The results are illustrated in Table 3 showing the results of test instances.

Tables 4 and 5 denote the experimental result using simple linear regression and multi-layer perceptron classification. Correlation coefficient denotes the correlation between

Table 4. Simple linear regression experimental results

Correlation coefficient	**0.8953**
Mean absolute error	0.0059
Root mean squared error	0.0439
Relative absolute error	13.8111%
Root relative squared error	45.0866%
Total Number of instances	390

Table 5. Multi-layer perceptron experimental results

Correlation coefficient	**0.9018**
Mean absolute error	0.1776
Root mean squared error	0.2123
Relative absolute error	52.6133%
Root relative squared error	55.3529%
Total Number of instances	294

predicted and actual target values. The range of correlation coefficient is between 0 and 1. The higher the value is, the closer it is to the target result. Simple linear regression yielded accuracy of 89.53% while the result of multi-layer perceptron is slightly higher at 90.18%. Mean absolute error measures the absolute difference between actual and predicted value. The mean absolute error of simple linear regression is very low at 0.59% while multi-layer perceptron is at 17.76%. These suggest the results are reasonable with relatively high accuracy. The biggest challenge in anomaly detection has been the need to rely on human analysis and sorting of information.

5. Conclusion.

This research has explored Malware attack dataset and developed a unique fuzzy K-means (FKM) clustering algorithm. FKM algorithm successfully created clusters of attacks, unsure and non-attacks. We presented a model that can detect anomaly. The process involves reduced human analysis time and higher accuracy. The highest accuracy is achieved through the use of multi-layer perceptron classifier with 90.18% of accuracy which is higher than the current state-of-the-art of 85% [17] we can find to benchmark. The long term goal would be to automate the integration process of dataset to send the statistical data to machine learning and data mining algorithms which would make a complete end-to-end process of identifying attack-related traffic from the network dataset.

Acknowledgment. This work was supported by both ST Electronics and the National Research Foundation (NRF), Prime Minister's Office, Singapore under Corporate Laboratory@University Scheme (Programme Title: STEE Infosec – SUTD Corporate Laboratory).

REFERENCES

[1] J. M. Ehrenfeld, WannaCry, cybersecurity and health information technology: A time to act, *Journal of Medical Systems*, vol.41, no.7, p.104, 2017.

[2] M. Rawson, Cyber-crime: WannaCry should make people treat cyber-crime seriously, *The Economist*, London, 2017.

[3] Symantec Security Response, *Petya Ransomware Outbreak: Here's What You Need to Know*, https://www.symantec.com/connect/blogs/petya-ransomware-outbreak-here-s-what-you-need-know, 2017.

[4] P. Bereziński, B. Jasiul and M. Szpyrka, An entropy-based network anomaly detection method, *Entropy*, vol.17, pp.2367-2408, 2015.

[5] A. M. Kosek and O. Gehrke, Ensemble regression model-based anomaly detection for cyber-physical intrusion detection in smart grids, *Proc. of IEEE Electrical Power and Energy Conference*, 2016.

[6] M. Amer, M. Goldstein and S. Abdennadher, Enhancing one-class support vector machines for unsupervised anomaly detection, *Proc. of the ACM SIGKDD Workshop on Outlier Detection and Description*, Chicago, IL, pp.8-15, 2013.

[7] M. Amer and M. Goldstein, Nearest-neighbor and clustering based anomaly detection algorithms for rapidMiner, *RapidMiner Community Meeting and Conference*, 2012.

[8] V. Chandola et al., *Data Warehousing and Data Mining Techniques for Computer Security*, Springer, 2006.

[9] T.-F. Yen et al., Beehive: Large-scale log analysis for detecting suspicious activity in enterprise networks, *Proc. of the 29th Annual Computer Security Applications Conference*, 2013.

[10] J. Levine et al., The use of honeynets to detect exploited systems across large enterprise networks, *IEEE Systems, Man and Cybernetics Society Information Assurance Workshop*, 2003.

[11] J. Zhang et al., Safeguarding academic accounts and resources with the university credential abuse auditing system, *The 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2012.

[12] M. J. Chapple, N. Chawla and A. Striegel, Authentication anomaly detection: A case study on a virtual private network, *Proc. of the 3rd Annual ACM Workshop on Mining Network Data*, 2007.

[13] P. Giura and W. Wang, Using large scale distributed computing to unveil advanced persistent threats, *Science*, vol.1, no.3, 2013.

[14] R. Zhong and G. Yue, DDoS detection system based on data mining, *Proc. of the 2nd International Symposium on Networking and Network Security*, Jinggangshan, China, 2010.

[15] E. García et al., Drawbacks and solutions of applying association rule mining in learning management systems, *Proc. of the International Workshop on Applying Data Mining in e-Learning (ADML)*, Crete, Greece, 2007.

[16] K. Veeramachaneni and I. Arnaldo, *AI2: Training a Big Data Machine to Defend*, https://people.csail.mit.edu/kalyan/AI2_Paper.pdf.

[17] R. M. Pierson, *AI Detects Cyber-Attacks with Accuracy after Attending MIT*, https://readwrite.com/2016/04/19/mit-ai-detect-cyber-attacks-accuracy-pl4/.