Analysis of IDS alerts by generalising features and discovering emerging patterns

Mahdi Maleki* and Seyed Mansour Shahidi

Faculty of Computer Engineering,
Ayatollah Boroujerdi University,
Boroujerd, Iran
Email: Mahdi.maleki@abru.ac.ir
Email: M.shahidi@abru.ac.ir
*Corresponding author

Abstract: One of the significant problems in using intrusion detection systems is the high volume of low-level alerts. In this paper, an appropriate analysis of cyber alerts has been used to reduce low-level alerts utilising a range of available features of attacks. It has also benefited from the discovery of emerging patterns to improve situational awareness in cyber-attacks. Moving to different levels of generalisation and extraction of rules; based on attribute-oriented induction and emerging patterns is a remarkable achievement of this. To evaluate the proposed method, a new CICIDS2017 database is used to eliminate the defects of the previous datasets. The results show a decrease in alerts at the rate of 99% at the lowest generalisation level and an average of 25% at other generalisation levels. In addition to normal traffic, 14 different types of attacks have been identified. The DoS Hulk attack has the highest frequency with 8.16%, and the heartbleed attack having the lowest frequency with 0.0004% frequency. On average, 18 overlap (TO-EP) pattern, 63 relatively subsumption-overlap patterns (SO-EP) and 92 similar (SIM-EP) patterns have been extracted at four generalisation levels.

Keywords: intrusion detection system; feature generalisation; multi-dimensional data mining; online analytical processing; OLAP; multistage attacks.

Reference to this paper should be made as follows: Maleki, M. and Shahidi, S.M. (2022) 'Analysis of IDS alerts by generalising features and discovering emerging patterns', *Int. J. Reasoning-based Intelligent Systems*, Vol. 14, No. 1, pp.56–65.

Biographical notes: Mahdi Maleki received her Bachelor's degree in Computer Engineering from Isfahan University of Technology in 1999, and then in 2006 he obtained a Master's degree in Computer Architecture from Amirkabir University of Tehran. In 2016, he received his Doctorate in Computer Science in the field of Artificial Intelligence. His interests are in the areas of artificial intelligence, intrusion detection systems and embedded systems.

Seyed Mansour Shahidi is pursuing his PhD in University of Isfahan. He completed his graduation in Hardware Engineering in 2005 from University of Isfahan, and postgraduation in Computer Architecture in 2008 from Amirkabir University of Technology. Her research interest includes information technology, digital processing and quantum computing.

1 Introduction

Today, due to the strong dependence on communication and computer systems, the effects of cyber-attacks are enormous. Therefore, the use of defence systems such as anti-malware, firewalls, and intrusion detection systems is inevitable. Analysing the output of defence systems before, during, and after an attack significantly contributes to cybersecurity. By analysing network traffic and security sensor outputs before the attack, it is possible to identify existing defects in the network and running applications. It makes the system more resistant to potential future attacks removing the holes. Analysing the online alerts during the raid discovered the attack and its type and the primary intention of the attackers. After the attack, by examining the event log files, network traffic, and issued alerts, the damaged points, the severity of the attack, and the attack source can be identified, which is very important in the discussion of cyber forensics. The growth of computer networks and the increase in the number of defence components in organisations have led to many security alerts. According to Vaarandi (2009) and Viinikka et al. (2009), each sensor generates an average of thousands of alerts per day, about 90% of which are false alerts (Long et al., 2006). Advanced and coordinated attacks such as botnets (Haas et al., 2016) can also significantly increase alert rates by executing multiple attacks in a short time. Growing intrusion detection systems lead to new issues, the

most important of which are: first, managing the volume of alerts generated and reducing them so that they do not obscure the attacks, and second, discovering the alert correlation to detect attack scenarios, especially in the advanced type of multi-stage attacks and covert attacks. Proper decision-making requires the analysis and processing information received to extract valuable and practical information from overhead information. Many efforts have been made for the above purposes. New data mining methods have created clear prospects for cyber-attack analysis. This paper uses attribute-oriented induction and emerging patterns techniques to analyse attacks.

2 Related works

With the expansion of network dimensions, collaborative intrusion detection systems (CIDS) were introduced (Carlos et al., 2016; Emmanouil et al., 2015; Min et al., 2005). In these systems, the defence components in different executive areas, with the participation of each other, produce the necessary alerts. After processing and discovering the relationship between the alerts, coordinated cyber-attacks are identified. In the detection of dependencies, these methods are divided into one-dimensional (Estan et al., 2003; Onwubiko, 2012) and multi-dimensional (Sheikhi et al., 2020; Locasto et al., 2005) categories. In the one-dimensional approach, the detection of dependencies is done by focusing on one feature (like discovering all the alerts with the same source address). These methods are simple but do not provide the ability to diagnose correctly.

In contrast, multi-dimensional methods with a focus on several features have an excellent ability to detect. Instead, the volume of calculations is very high, especially in high network traffic. Various methods try to turn raw alerts into a template that maintains the overall state of the attacks and reduces false alarms. In one approach, graphs centred on network motif have been used to minimise alerts (Steffen and Mathias, 2018). In another method, among all the features, several features (includes source address, source port, destination port and protocol) are selected based on their analysis and role in identifying different attacks. The graph (lattice graph) is formed based on the root of the source address, and the number of different patterns in the graph nodes is calculated. Then, according to the threshold values, some of the alerts are removed (Zhou et al., 2009). In another approach, the frequent itemset mining in features has been used to find alerts with a certain number of repetitions.

The algorithms in this area are a subset of the apriori (Agrawal and Srikant, 1994), FP-growth (Han et al., 2000), and Eclat (Zaki, 2000) methods. The main problem with the above methods in reducing alerts is the lack of attention to logical and conceptual communication between the features, which will cause the lack of necessary transparency in the output of these systems and low ability to make decisions in detecting attacks. One way to data generalisation in data mining is to convert low-level data into higher-level concepts and eliminate unrelated features. This work, also called concept description, provides characterisation and discrimination between different data classes. This critical method in data generalisation is the attribute-oriented induction method (Beneditto, 2004; Han et al., 1992, 2011; Han and Fu, 1995). The basis of this method is the feature generalisation using the hierarchy between them. Relationship databases provide the data needed for data mining in the attribute-oriented induction method by queries (languages such as SQL) (Meo et al., 1998; Muyeba and Marnadapali, 2005). Queries with SQL structure were created with the aim of data mining query language (DMQL) (Elfeky et al., 2000). Attribute-oriented induction has been used in extracting rules (Cheung et al., 2000), classification (Cai et al., 1990), clustering (Wu and Xie, 2003), and in discovering repetitive patterns (Warnars, 2016). Another interesting technique used in this article is emerging patterns. The basis of this method is the discovery of knowledge through the analysis of emerging trends from a dataset to another dataset (Dong and Li, 1999). This method can also be used to measure large changes in the itemsets in different classes and use it to differentiate classes (Dong et al., 1999). In Zhang et al. (2000), a method based on information-based classifier has used the aggregation of emerging patterns for classification, which has been more accurate than the previous methods and has reduced the training time in the classification and the method. In Li et al. (2004) is an instance-based classifier that makes decisions based on emerging patterns. The advantage of this method over the previous ones is reducing the number of samples and dimensions of samples (number of features) and improving speed and accuracy.

The purpose of this paper is to properly analyse cyber alerts by reducing low-level alerts using the hierarchy of attack features. The discovery of emerging patterns in attack and normal network classes has also been used to improve situation awareness in organisational cyberspace. Moving to different levels of generalisation in cyberspace attacks and extracting rules based on emerging patterns is a notable achievement of this article in acquiring knowledge in this field.

The organisation of the article is as follows: in Section 2, the attribute-oriented induction algorithm and emerging patterns are described. In Section 3, the proposed method based on the above algorithms in the field of cyber-attacks has been developed and implemented following the network used in the dataset. In Section 4, the output of the method on the presented dataset is displayed. The results and rules produced are also evaluated. Section 5 concludes the study.

3 Research method

Conceptual relationships between a range of existing features make it possible to describe them better, and as a result, to understand the alerts better and to identify existing threats. Therefore, algorithms have been used in data mining to generalise alerts by considering relationships between features and reducing the volume of alerts while maintaining the relationships between them. In this method, first, the data related to the specific domain is collected by the dataset query, then the incomplete records are corrected in the pre-processing stage. The generalisation step involves attribute removal or attribute generalisation. Then, the aggregation operation is performed on the duplicate records, and the output of the concepts is generated. The database is then divided into two different attack and normal classes, and according to the method provided in Warnars (2016), frequent patterns and similar patterns are found between the two classes. Then, the output of this method is given to the unit of discovery of emerging patterns, and according to a specific threshold value, emerging patterns are discovered. The steps of the proposed method are as shown in Figure 1. The concepts produced can be represented by various representations and mappings such as diagrams, rules, etc. In the next section, the main algorithm of the proposed method is described. Also, the proposed method and implementation of its essential components are stated.

3.1 Attribute-oriented induction

This method, which is one of the data mining methods, is based on generalising the data considering the hierarchy between the features. Data generalisation consists of two main parts: attribute removal and attribute generalisation.

- Attribute removal: Columns in which the number of different states of attributes is greater than the threshold value are deleted if one of the following conditions occurs:
  1. There is no conceptual hierarchy for that column.
  2. There is a conceptual hierarchy for this column in other columns.
- Attribute generalisation: If the number of different states of attributes of a column exceeds a particular threshold value, if there is a series of hierarchical concepts for that column, the column's attributes will be generalised.

Determining threshold values depends on the attributes and scope of the application. These values are chosen by the expert or user, depending on which attributes need more generalisation and which attributes need less generalisation. This option is called attribute generalisation control. This value is determined so that the properties are not generalised too much (overgeneralisation) and the features are not at low-levels (under generalisation). In either case, the rules will not produce the necessary and valuable information. There are several ways to determine threshold values, two of which are:

1. attribute generalisation threshold control
2. generalised relation threshold control.

In the first, threshold values are defined for all attributes or individual attributes. In data mining, these values are usually between 2 and 8, and the expert or user selects the specific value. In the second method, the threshold value is selected for the number of rules (generalised output), and usually, in data mining systems, the number is between 10 and 30. For example, if the number of output relations is greater than the threshold value after the generalisation, generalisation is performed on the attributes again. In the proposed algorithm, the first method is used. According to the number of hierarchical levels in the attributes, this value is changed for the address attributes (IP source, IP Dest.) in the range of 8 to 3 to obtain different generalisation outputs. For other attributes, the number 2 has been selected due to the lower levels.

Figure 1 Steps of the proposed method

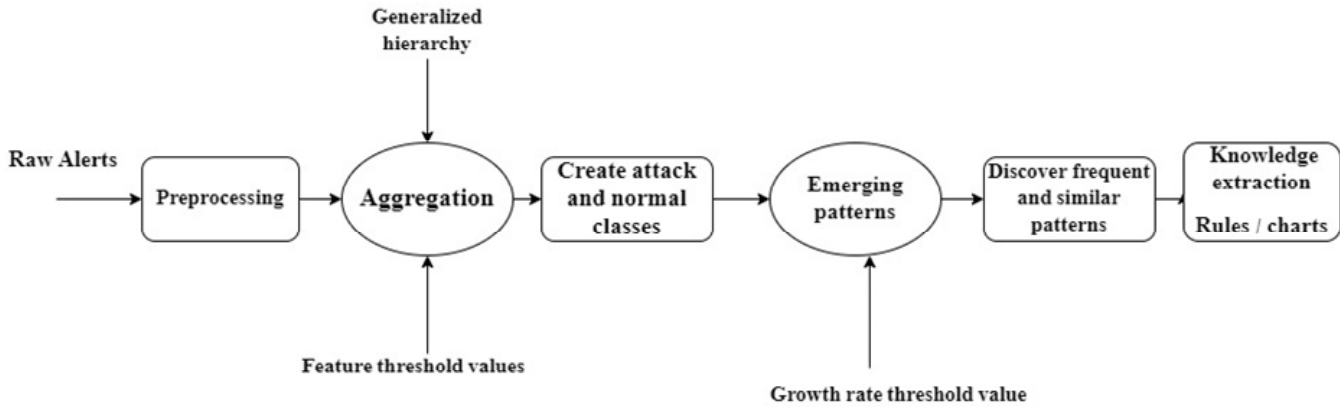


Figure 2 illustrates the hierarchy of the concept for the position attribute. As shown in Figure 2, in generalising the position attribute, the direction of movement is as follows:

Street > city > province > country > all

Figure 2 Conceptual hierarchy for location feature

The attribute-oriented induction algorithm consists of the following steps shown in Table 1.

Table 1 AOI steps

Step	*Process detail*	*Process name*
1	Attributes related to the work area are extracted from the database.	Data capturing
2	The defective data values are corrected.	Pre-processing
3	The initial generalisation step is performed: a The number of different modes of each attribute in the columns is calculated. b The attribute removal or attribute generalisation is performed, and if generalised, the value of each attribute is replaced by the generalised value.	Generalisation
4	The aggregation operation is performed throughout the database table. This is done by merging duplicate records and calculating the number of duplicate records.	Aggregation
5	If the number of records in the table does not reach the desired level (the information is not generalised to the desired level), Steps 2 and above are repeated.	Termination condition test

3.2 Emerging patterns

The use of emerging patterns is one of the new ways to gain knowledge on a database. These patterns are called a set of items whose support's value changes significantly from one set of data to another set of data. Discovering these patterns is very useful in finding emerging trends in time series datasets and distinguishing different data classes. Important relationships of emerging patterns are as follows:

Suppose $D1$ and $D2$ are two datasets, and X is an itemset from this dataset. The amount of support X in the dataset D is defined as follows:

$$\sup_D(X) = \frac{count_D(X)}{|D|} \tag{1}$$

In this equation, $count_D(X)$ is the number of records in the dataset that includes the set of items X and $|D|$ is the total number of database records. The growth rate function is obtained from the following equation:

$$GrowthRate(X) = \begin{cases} 0, & \text{if } \sup_{D1}(X) = 0 \text{ and } \sup_{D2}(X) = 0 \\ \infty, & \text{if } \sup_{D1}(X) = 0 \text{ and } \sup_{D2}(X) \neq 0 \\ \dfrac{\sup_{D2}(X)}{\sup_{D1}(X)}, & \text{otherwise} \end{cases} \tag{2}$$

Given the threshold value of the growth rate ρ, the itemset X from database $D1$ to database $D2$ is called the emerging pattern of degree ρ (ρ-EP) if:

$$GrowthRate(X) \geq \rho \tag{3}$$

In the next section, the main algorithm will be developed according to the proposed network for detecting cyber-attacks, as well as a series of conceptual hierarchies for the desired features. An intuitive way to select features to generalise has also been developed. Finally, the discovery of emerging patterns is designed according to the problem.

3.3 The proposed method

In this section, the attribute-oriented induction and emerging patterns algorithms have been modified according to the necessity of detecting cyber-attacks. A good feature of the proposed method is that discovering emerging patterns is presented on high-level output data, not on low-level data, which will speed up the discovery of emerging patterns. According to the articles and reports posted on the vulnerability alert sites, the essential features for detecting attacks are source IP address, destination IP address, source port, destination port and protocol (Zhou et al., 2010; Estan et al., 2003; ICS-CERT Advisories, 2019; Internet Storm Center, 2019; Haas et al., 2019; Hu et al., 2006; NIST, 2019). In addition to these features, counting and labelling features have also been added to the data. Counting property, which is created by an initial value of 1 for each alert, is increased in the generalisation process, and can be used as an important measurement of alert for the decision-maker. In addition, using the label field, the output of generalised alerts can be categorised according to normal traffic and attack traffic (all types of attacks). After the initial extraction of the data according to the stated features, pre-processing of the data is performed. For example, incomplete records are deleted, or some attributes (count attribute) are assigned. One of the key components of the proposed algorithm is to generate the necessary structures for the concept hierarchy for features that have such relationships. The design of such structures created by experts is very much related to the target network. These structures guide the algorithm in the generalisation process as a pre-knowledge. Figures 3–5 show the tree structures for

the concepts hierarchy for IP addresses, port numbers and protocols. According to the target network shown in Figure 6, these structures are designed according to the network of the CICIDS2017 dataset (Sharafaldin et al., 2018).

Figure 3 Conceptual hierarchy for IP addresses

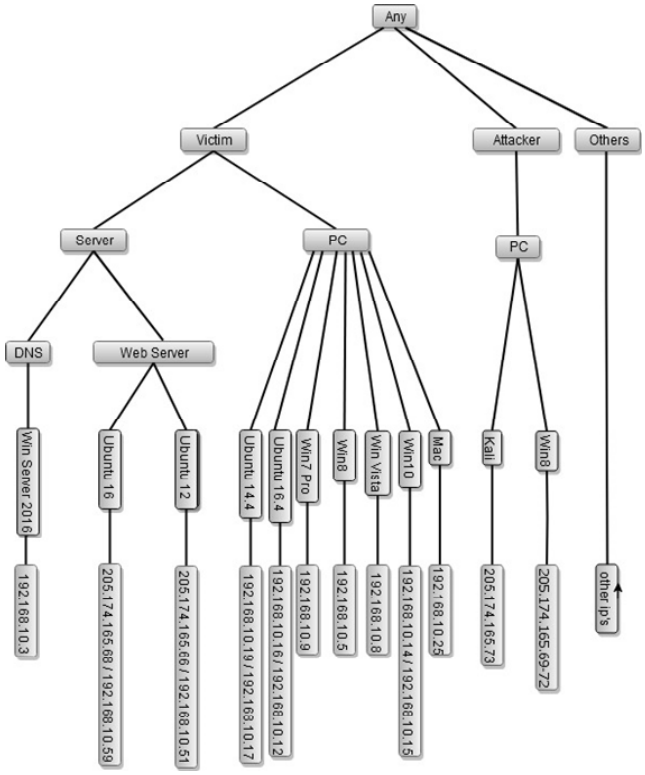


Figure 4 Conceptual hierarchy for ports

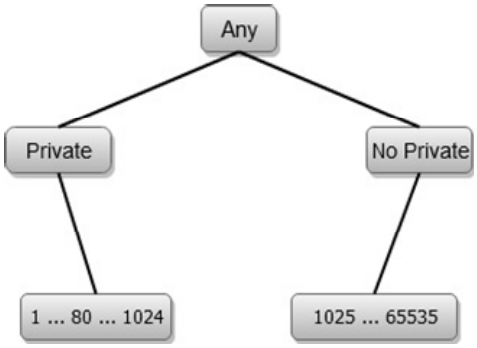
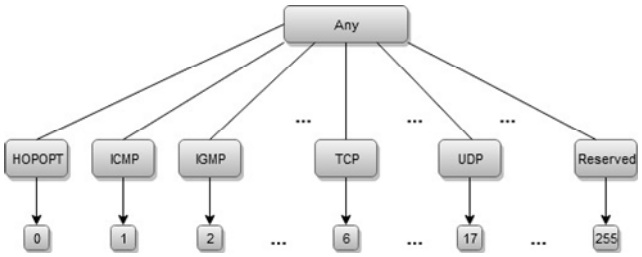
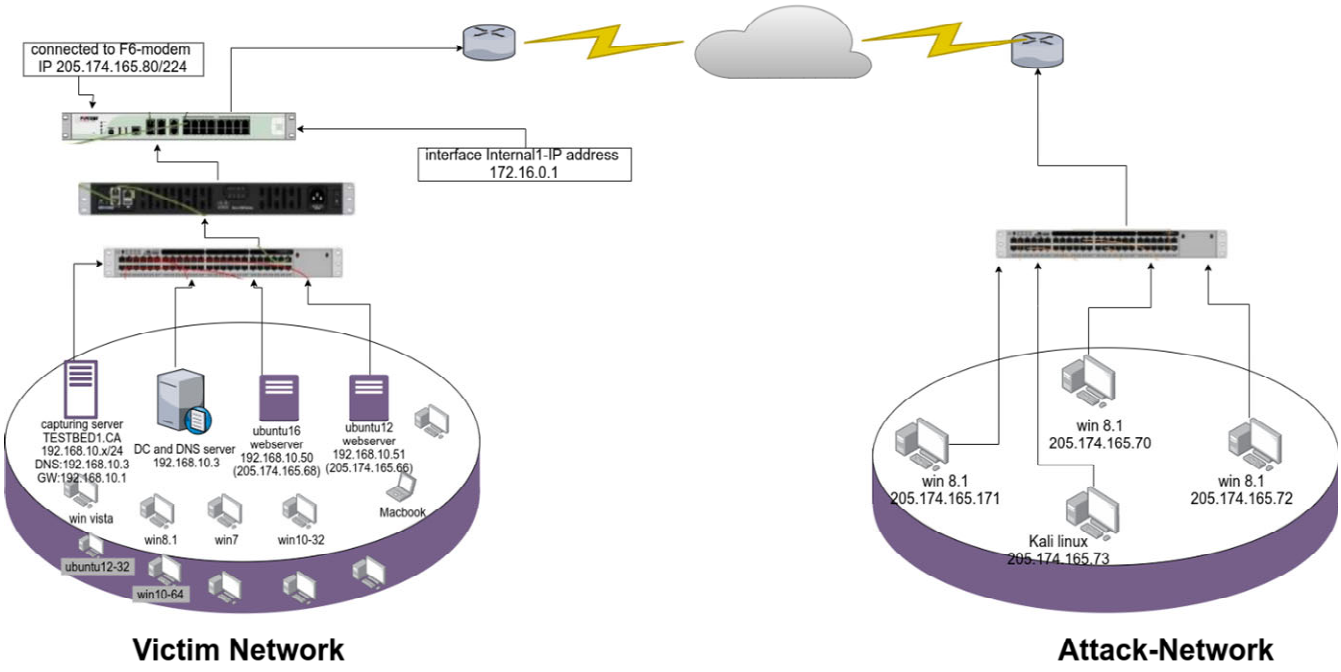


Figure 5 Conceptual hierarchy for protocol



In the following, the attribute-oriented induction algorithm is applied to the dataset. After generalising the relationships, the database is divided into two parts, the attack database and the normal database. According to Warnars (2016), the generalised relationships between the two databases are compared and categorised into different classes. Then, by implementing the emerging patterns algorithm on these classes, various trends have been extracted, from which knowledge can be obtained in the form of different rules.

Figure 6 Test network structure (see online version for colours)



Source: Edited from Sharafaldin et al. (2018)

3.3.1 Categorise generalised relationships in emerging patterns

The i^{th} relation from the D_x database is shown with r_i^x and is defined as follows:

$$r_i^x = \left\{ A_1^x, A_2^x, A_3^x, \cdots, A_m^x, \left| r_i^x \right| \right\} \tag{4}$$

where A_1^x is the value of the first attribute, and m is the total number of attributes in each relation and $r_i^x \mid$ is the total number of relations i in the database x. The attribute A_1^x is part of A_2^x and is represented by the symbol $A_1^x \subset A_2^x$. If in the generalisation hierarchy, the attribute A_2^x is part of the ancestor of nodes A_1^x. If r_i^1 and r_j^2 are two relationships from database $D1$ and $D2$, the following categories are defined in the emerging patterns according to the relationships between the features:

- Totally overlap-emerging patterns (TO_EP): If all the features are equal in the two relationships.
- Totally subsume-emerging patterns (TS_EP): If for all the features in two relations one is a subset of the other or vice versa: $A_i^1 \subset A_j^2$ or $A_j^2 \subset A_i^1$.
- Subsume overlap-emerging patterns (SO_EP): If between two relations, some features are equal, and some features are a subset of another feature.
- Similar emerging patterns (Sim-EP): If at least 90% of the features (m – 1 features) are equal between the two relations.
- Frequent emerging patterns (Freq-EP): If at least 90% of the features of a relationship in $D1$ database are a subset of the features in $D2$ database.

3.3.2 Heuristic function H(c)

By analysing the cyber alerts and extracting the relationships between the main features of the alerts, such as what appeared in the trees of the conceptual hierarchy, we find that the levels of the hierarchy are not set equally in all the features. For example, the hierarchy for IP address columns has the highest (level 6), while port and protocol columns have the lowest value (level 3). Therefore, to have the required transparency for the generalised values, we must start the generalisation from the IP columns. Otherwise, columns with lower levels will quickly generalise to the value of *any*, while columns with higher levels will be placed in the middle values, and this is solved in the heuristic function:

$$H(c) = \mathrm{MAX}\left(Depth\left(C.H.T.(c_i) \right) \right) \tag{5}$$

The function *Depth* gives the depth of the conceptual hierarchy tree $C.H.T(c_i)$, and the variable c_i refers to the column i or the attribute i.

4 Results and discussion

The new CICIDS2017 dataset was used to evaluate the proposed method. The advantages of this free dataset over conventional datasets (KDD'99, DEFCON, CAIDA and CDX) are solving problems such: lack of proper balance between the number of attacks and the number of normal traffic, existence of packets with intentionally hidden and unspecified features that make it challenging to analyse attacks, restrictions on a variety of attacks, especially new attacks and the lack of a complete set of features and metadata (CAIDA, 2019; CDX 2009 Dataset, 2019; Cowan, 2003; K.D.D. Cup, 1999; Thomas et al., 2008). The network used in this experiment is shown in Figure 6, which consists of two main parts: the victim network and the attack network. Python programming language and Pandas library have been used to implement the method (McKinney, 2011). The Anytree Library has also been used to implement hierarchy trees (Anytree 2.7.3 Documentation, 2019). Table 2 shows the first four alerts of program output at both level 1 (GL1) and level 3 (GL3) for attack and normal classes. Various generalisation levels have been performed from GL0 (without generalisation) to GL5 (maximum generalisation). More than this level of generalisation will not generate new information. The GL0 output has very detailed information. The high volume of alerts (about 28,300,000) and many numerical variables make it difficult for even experts to make quick and appropriate decisions. However, with the generalisation at higher levels, the volume of alerts has been reduced. A more relevant picture of the attacks has been obtained by generalising the numerical values to the corresponding letter values. Table 3 shows the rate of decrease in alerts at different levels. As can be seen, the GL1 had the largest decrease (99%) because the alerts were not generalised, and the low-level alerts, which are high in number (which mostly have normal traffic), were generalised. However, at other levels of generalisation, the reduction rate is lower (25%) because operations are performed on less generalised alerts. Table 4 shows the attacks that were discovered and their traffic ratios. The calculated values are derived from the following equations:

$$f_i = \frac{f}{n} \times 100 \tag{6}$$

In the above equation, f_i is the percentage of alerts of type i attacks, and f is the number of alerts of type i attacks, and n is the total number of alerts.

$$R_i = \frac{A_{i-1} - A_i}{A_{i-1}} \times 100 \tag{7}$$

In the above equation, A_{i-1} is the number of alerts at the generalisation level i – 1, and A_i is the number of alerts at level i, and R_i is the percentage reduction of the alerts at the generalisation from level GL_{i-1} to level GL_i.

Table 2 First four alerts of GL1 and GL3 for attack and normal classes

Normal							
Row ID	*Source IP*	*Source port*	*Destination IP*	*Destination port*	*Protocol*	*Label*	*Count*
GL1: generalisation level #1 (IP + PORT + PROTOCOL)							
0	Kali	No private	Win 8_V	No private	TCP	Benign	7
1	MAC	No private	Kali	Private	TCP	Benign	31
2	MAC	No private	Others	No private	TCP	Benign	105
3	MAC	No private	Others	No private	UDP	Benign	107
GL3: generalisation level #3 (IP)							
0	Any	No private	Server	No private	TCP	Benign	816
1	Any	No private	Server	Private	TCP	Benign	289
2	Any	No private	Victim	No private	TCP	Benign	283
3	Any	No private	Victim	No private	UDP	Benign	2
Attack							
Row ID	*Source IP*	*Source Port*	*Destination IP*	*Destination port*	*Protocol*	*Label*	*Count*
GL1: generalisation level #1 (IP + PORT + PROTOCOL)							
0	Kali	No private	Win 10	No private	TCP	Bot	348
1	Kali	No private	Win 7pro	No private	TCP	Bot	146
2	Kali	No private	Win 8_V	No private	TCP	Bot	108
3	Kali	No private	Win Vista	No private	TCP	Bot	103
GL3: generalisation level #3 (IP)							
0	Any	No private	Server	No private	TCP	PortScan	133,891
1	Any	No private	Server	Private	TCP	DDoS	128,024
2	Any	No private	Server	Private	TCP	DoS GoldenEye	10,293
3	Any	No private	Server	Private	TCP	DoS Hulk	231,073

Table 3 The rate of reduction of alerts at different levels of generalisation

Generalisation level	*Number of alerts*	*Reduction rate of alerts*
GL0 (IP + PORT + PROTOCOL)	2,830,365	0.00%
GL1 (IP)	294	99.99%
GL2 (IP)	91	69.05%
GL3 (IP)	69	24.18%
GL4 (IP)	44	36.23%
GL5 (IP)	23	47.73%

Figure 7 Frequency of emerging patterns at different levels of generalisation (see online version for colours)

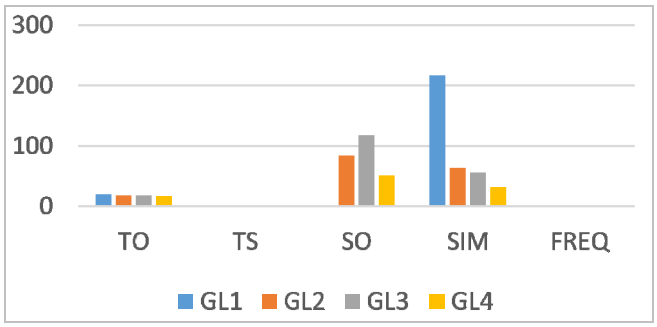


Table 4 Ratio of number of alerts in normal traffic and attacks

Attacks	*Number of alerts*	*Percentage*
BENIGN	2,273,097	80.30%
DoS Hulk	231,073	8.16%
PortScan	158,930	5.61%
DDoS	128,027	4.52%
DoS GoldenEye	10,293	0.36%
FTP-Patator	7,938	0.28%
SSH-Patator	5,897	0.21%
DoS slowloris	5,796	0.20%
DoS Slowhttptest	5,499	0.19%
Bot	1,966	0.07%
Web attack – Brute Force	1,507	0.05%
Web attack – XSS	652	0.02%
Infiltration	36	0.00%
Web attack – Sql injection	21	0.00%
Heartbleed	11	0.00%
Total	2,830,743	

Table 5 A few examples of emerging patterns from the generalised level 1

EP	*i*	*j*	*Over.*	*Subs.*	*Grow rate*	*Count 1*	*Count 2*	*|n1|*	*|n2|*
TO-GL1	0	2	5	0	62.84	7	108	2,271,407	557,640
SO-GL2	0	14	4	1	10.15	283	705	2,271,407	557,640
SIM-GL1	0	1	4	0	84.96	7	146	2,271,407	557,640

Table 6 Number of different emerging patterns discovered on GL1 to GL4

EP type	*GL1*	*GL2*	*GL3*	*GL4*	*Average*
TO	20	18	18	17	18
TS	0	0	0	0	0
SO	0	84	118	51	63
SIM	217	64	56	32	92
FREQ	0	0	0	0	0

Table 5 shows three examples of emerging patterns extracted. The EP column describes the type of emerging pattern and the generalisation level used to extract this pattern. Columns *i* and *j*, respectively, refer to the corresponding relation row in the normal and attacks data table, according to Table 2. The over and subs-column indicates the number of similar and sub-section attributes found in the emerging pattern, respectively. The growth rate column is calculated based on equation (2). The next two columns calculate the number of similar relationships, and the last two columns indicate the total number of relationships in normal and attacks data. Table 6 shows the number of emerging patterns by type of pattern and level of generalisation used. As shown in Table 6 and Figure 7, similar patterns (sim) with frequency 217 have the highest frequency, and TS and FREQ patterns with frequency 0 have the lowest frequency. By analysing the database, it is clear that the reason for the zero frequency of the above patterns is the symmetry of the hierarchy of generalisation of port and protocol features and the identical generalisation of these features.

Here are some conclusions from the program's outputs:

1 In addition to the benign alerts, there are 14 different types of attacks, which according to Table 4 of DoS Hulk attacks with a frequency of 8.16% have the highest frequency and heartbleed attacks with a frequency of 0.0004% have the lowest frequency among the attacks.

2 From the GL0 outputs, it is clear that a bot attack was carried out from address 192.168.10.15 with port number 54012 to address 205.174.165.73 with port number 8080 with protocol number 6 and count number of 1. As can be seen, low-level data with low repetition does not allow for the right decision on the severity of the attacks.

3 Going to higher generalisation levels (GL2–GL5) will give you a higher view of the alerts and discover the trend between the alerts. As an example, in the last generalisation phase, three types of DoS: Hulk, port scan, and DDos attacks were more common. By moving backward in generalisation, more detailed information can be obtained. For example, DDos attacks have attacked web server the most.

4 According to row 1 of Table 5 and Table 2, precisely the same alerts have been found with kali origin address and Win-8 destination and TCP protocol with a growth rate of 63 times. High growth rates indicate a high probability of attacks with these characteristics.

5 According to row 3 of Table 5 and Table 2, similar alerts with kali origin and TCP protocol with a growth rate of 85 have appeared. By looking at this pattern, the analyst will conclude that most likely, the source of the attacks is the kali system, and the protocol used is TCP.

6 According to Figure 7, the highest frequency of the discovered patterns is related to SIM, and then SO patterns and TS and FREQ patterns have not been found.

Roll-up and drill-down in the alerts provided by the proposed method allows for online analytical processing (OLAP) operations and multi-dimensional data mining.

5 Conclusions

In this paper, a combination of two effective methods of data generalisation from data mining field called attribute-oriented induction and emerging patterns has been used, and it has been developed and implemented in order to detect cyber-attacks in the target network by the CICIDS2017 dataset. For this purpose: first, the levels of hierarchy are designed for important features in identifying attacks. An intuitive method for feature selection in the generalisation process is then presented. Also, by the emerging patterns method, the growth rate of high-level alerts generated in both normal and attack classes was evaluated, and on average, 18 TO_EP, 63 SO_EP, and 92 Sim-EP were extracted at four generalisation levels. By these patterns and extracting rules from them, situational awareness in the field of cyber-attack detection will increase. The results indicate the appropriate generalisation (99% at generalisation level 1, and at least 25% at other generalisation levels) of alerts generated in intrusion detection systems in order to detect and accurately differentiate attacks. With normal traffic, 14 different types of attacks were identified, of which DoS Hulk attack with a frequency of 8.16% had the highest and heartbleed attack with a frequency of 0.0004% had the lowest.

Also, by various generalisation outputs, it is possible to move forward and backward at different levels of alerts necessary for proper analysis in multi-dimensional data mining in the field of cyber-attack detection. Compared to the standard methods in this field, the proposed method has brought more general situational awareness and knowledge acquisition by integrating generalised hierarchies and emerging patterns. It also works better in terms of the number of generated output rules and runtime. In future research, by implementing criteria, the quality of alerts provided to the user at different levels can be calculated, and false and real alerts can be calculated with probabilistic values. This requires a closer look at other effective features of cyber-attacks in network packets, also implementing other data warehousing tools, such as cutting, rotating, and projection, other than moving forward and backward, will be future actions.

References

Agrawal, R. and Srikant, R. (1994) 'Fast algorithms for mining association rules', in *Proc. 1994 Int. Conf. Very Large Data Bases (VLDB'94)*, Santiago, Chile, pp.487–499.

Anytree 2.7.3 Documentation (2019) [online] https://anytree.readthedocs.io/en/latest/intro.html.

Beneditto, M. (2004) 'Using concept hierarchies in knowledge discovery', *Lecture Notes in Computer Science*.

Cai, Y., Cercone, N. and Han, J. (1990) 'An attribute-oriented approach for learning classification rules from relational databases', in *Proceedings. Sixth International Conference on Data Engineering*, IEEE, pp.281–288.

Carlos, G., Sascha, H., Max, M. and Mathias, F. (2016) 'Analyzing flow-based anomaly intrusion detection using replicator neural networks', in *Annual Conference on Privacy, Security and Trust (PST)*.

CDX 2009 Dataset (2019) [online] https://www.usma.edu/centers-and-research/cyber-research-center/data-sets (accessed January 2019).

Center for Applied Internet Data Analysis (CAIDA) (2019) [online] https://www.caida.org (accessed January 2019).

Cheung, D.W., Hwang, H. and Fu, A.W. (2000) 'Efficient rule-based attribute-oriented induction for data mining', *Journal of Intelligent Information Systems*, Vol. 15, No. 2, pp.175–200.

Cowan, C. (2003) 'Defcon capture the flag: defending vulnerable code from intense attack', *Proceedings DARPA Information Survivability Conference and Exposition*, IEEE, Vol. 1.

Dong, G. and Li, J. (1999) 'Efficient mining of emerging patterns: discovering trends and differences', in *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.43–52.

Dong, G., Zhang, X., Wong, L. and Li, J. (1999) 'CAEP: classification by aggregating emerging patterns', in *Proceedings of the 2nd international Conference on Discovery Science. Lecture Notes in Computer Science*, Vol. 1721, pp.30–42.

Elfeky, M.G., Saad, A. and Fouad, S.A. (2000) 'ODMQL: object data mining query language', in *Proceedings of the International Symposium on Objects and Databases*, pp.128–140.

Emmanouil, V., Shankar, K., Max, M. and Mathias, F. (2015) 'Taxonomy and survey of collaborative intrusion detection', *ACM Computing Surveys (CSUR)*, Vol. 47, No. 4, pp.1–33.

Estan, C., Savage, S. and Varghese, G. (2003) 'Automatically inferring patterns of resource consumption in network traffic', in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*. pp.137–148.

Haas, S., Florian, W. and Mathias, F. (2019) *Efficient Attack Correlation and Identification of Attack Scenarios Based on Network-Motifs*, arXiv preprint arXiv: 1905.06685.

Haas, S., Karuppayah, S., Manickam, S., Mühlhäuser, M. and Fischer, M. (2016) 'On the resilience of P2P-based Botnet graphs', in *IEEE Conference on Communications and Network Security (CNS)*.

Han, J. and Fu, Y. (1995) *Exploration of the Power of Attribute-Oriented Induction in Data Mining*, PDF, Simon Fraser University.

Han, J., Cai, Y. and Cercone, N. (1992) 'Knowledge discovery in databases: an attribute-oriented approach', in *Proceedings of the 18th International Conference on Very Large Data Bases*, pp.547–559.

Han, J., Micheline, K. and Jian, P. (2011) *Data Mining Concepts and Techniques*, 3th ed., Elsevier.

Han, J., Pei, J. and Yin, Y. (2000) 'Mining frequent patterns without candidate generation', in *Proc. 2000 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'00)*, Dallas, TX, pp.1–12.

Hu, Y., Chiu, D. and Lui, J. (2006) 'Adaptive flow aggregation – a new solution for robust flow monitoring under security attacks', in *Proceedings of the 10th IEEE/IFIP Network Operations and Management Symposium (NOMS)*, pp.424–435.

ICS-CERT Advisories (2019) *Information About Current Security Issues, Vulnerabilities, and Exploits* [online] https://www.us-cert.gov/ics/advisories (accessed January 2019).

Internet Storm Center (2019) DShield.org [online] http://www.dshield.orgi (accessed January 2019).

K.D.D. Cup (1999) *Intrusion Detection Data Set*, The UCI KDD Archive Information and Computer Science, University of California [online] http://kdd.ics.uci.Edu/databases/kddcup99 (accessed March 2000).

Li, J., Dong, G., Ramamohanarao, K. and Wong, L. (2004) 'DeEPs: a new instance-based discovery and classification system', *Machine Learning*, Vol. 54, No. 2, pp.99–124.

Locasto, M., Parekh, J., Keromytis, A. and Stolfo, S. (2005) 'Towards collaborative security and P2P intrusion detection', in *Proceedings of the IEEE Workshop on Information Assurance and Security*.

Long, J., Schwartz, D. and Stoecklin, S. (2006) 'Distinguishing false from true alerts in snort by data mining patterns of alerts', in *Proc. of SPIE Defense and Security Symposium*, pp.62410B-1–62410B-10.

McKinney, W. (2011) 'Pandas: a foundational python library for data analysis and statistics', *Python for High Performance and Scientific Computing*, No. 14.9.

Meo, R., Psaila, G. and Ceri, S. (1998) 'An extension to SQL for mining association rules', in *Proceedings of Data Mining and Knowledge Discovery*, pp.2195-2224.

Min, C., Kai, H., Yu-Kwong, K., Shanshan, S. and Yu, C. (2005) 'Collaborative internet worm containment', *IEEE Security & Privacy*, Vol. 3, No. 3, pp.25–33.

Muyeba, M. and Marnadapali, R. (2005) 'A framework for post-rule mining of distributed rules bases', in *Proceeding of Intelligent Systems and Control*.

Onwubiko, C. (2012) *Situational Awareness in Computer Network Defense: Principles, Methods and Applications*, IGI Global, London.

Sharafaldin, I., Lashkari, A.H. and Ghorbani, A. (2018) 'Toward generating a new intrusion detection dataset and intrusion traffic characterization', *4th International Conference on Information Systems Security and Privacy (ICISSP)*, Portugal.

Sheikhi, S., Kheirabadi, M.T. and Bazzazi, A. (2020) 'An effective model for sms spam detection using content-based features and averaged neural network', *International Journal of Engineering (IJE), IJE Transactions B: Applications*, Vol. 33, No. 2, pp.221–228.

Steffen, H. and Mathias, F. (2018) 'GAC: graph-based alert correlation for the detection of distributed multi-step attacks', in *ACM/SIGAPP Symposium on Applied Computing (SAC)*.

The National Institute of Standards and Technology (NIST) (2019) *National Vulnerability Database* [online] https://nvd.nist.gov/vuln (accessed January 2019).

Thomas, C., Vishwas, S. and Balakrishnan, N. (2008) 'Usefulness of DARPA dataset for intrusion detection system evaluation', *Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security*, International Society for Optics and Photonics, Vol. 6973.

Vaarandi, R. (2009) 'Real-time classification of IDS alerts with data mining techniques', *Proceedings of the 2009 IEEE MILCOM*.

Viinikka, J., Debar, H., Mé, L., Lehikoinen, A. and Tarvainen, M. (2009) 'Processing intrusion detection alert aggregates with time series modeling', *Information Fusion Journal*, Vol. 10, No. 4, pp.312–324.

Warnars, H. (2016) 'Using attribute oriented induction high level emerging pattern (AOI-HEP) to mine frequent patterns', *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 6, No. 6, pp.3037–3046.

Wu, X. and Xie, L. (2003) *Attribute-oriented Induction and Conceptual Clustering*, pp.92–99, Computer Engineering, Beijing.

Zaki, M.J. (2000) 'Scalable algorithms for association mining', *IEEE Trans. Knowledge and Data Engineering*, Vol. 12, No. 3, pp.372–390.

Zhang, X., Dong, G. and Ramamohanarao, K. (2000) 'Information-based classification by aggregating emerging patterns', in *Proceedings of the 2nd International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'00)*, pp.175–188.

Zhou, C.V., Leckie, C. and Karunasekera, S. (2009) 'Decentralized multi-dimensional alert correlation for collaborative intrusion detection', *Journal of Network and Computer Applications*, Vol. 32, No. 5, pp.1106–1123.

Zhou, C.V., Leckie, C. and Karunasekera, S. (2010) 'A survey of coordinated attacks and collaborative intrusion detection', *Computers & Security*, Vol. 29, No. 1, pp.124–140.