Unsupervised Intrusion Detection based on FCM and Vote Mechanism

Longlong Li, Qin Chen, Shuiming Chi and Xiaohang Liu
College of Computer Science, Hangzhou Dianzi University, 310018, Hangzhou, China
Jinhua Biqi Network Technology Co., Ltd., 321017, Jinhua, China

Abstract: Two different Intrusion Detection Systems (IDS) are by far dominant in current commercial network attacks detection: misuse detection and anomaly detection. Both of them are motivated by the limitations of knowledge-based approaches which may cause the limitations of detection performance. In this paper, we present an Unsupervised Intrusion Detection System (UIDS) capable of detecting network attacks based on FCM-Vote algorithm which do not need any previous knowledge of either the characteristics of attack or training set. FCM-Vote based IDS uses a novel network malicious ows detection approach based on FCM clustering and clustering fusion. The new approach improve the robustness of UIDS which based on clustering and has high accuracy in detecting the network malicious ows. In order to validate these advantages of such an Unsupervised Intrusion Detection System, we evaluate its ability in a well-known existing dataset of traffic features. We particularly show the low rate of false alarm of FCM-Vote based IDS by comparing its performance against traditional anomaly detection which based on limited knowledge.

Keywords: Intrusion detection, clustering algorithm, clustering of fusion, subspace clustering

INTRODUCTION

Distributed Denial of Service attacks (DDoS), buffer overflow attacks, network scanning activities, and spreading worms or viruses are examples of the different threats that daily compromise the integrity and normal operation of the network. Intrusion detection is a technique to discover invasive by collecting and analyzing information on the protected systems. Its main function is to provide real-time monitoring, detection and recognition system intrusion or attempted intrusion alarm for network and computer systems (Hansman and Hunt, 2005). Two different Intrusion Detection Systems (IDS) are in current commercial network attacks detection: Misuse detection and anomaly detection. Both of them depend on the limitations of knowledge-based approaches. But it is not easy to maintain an accurate and up-to-date normal-operation profile, particularly in a dynamic and evolving context where new services and applications are constantly emerging (Casas et al., 2012). Unsupervised Anomaly Detection (UAD) has been studied in the last few years which uses Data Mining techniques to extract patterns and similar structures, without relying on signatures or baseline traffic profiles (Androulidakis et al., 2009). Some methods for unsupervised detection of network attacks have been proposed in the past (Portnoy et al., 2001; Eskin et al., 2002; Leung and Leckie, 2005; Lakhina et al., 2005) the majority of them are based on clustering techniques and outliers detection. Clustering addresses multiple-class problems without modifying the characteristics of the analyzed traffic. Unfortunately, even if hundreds of clustering algorithms exist (Jain, 2010), one of the major drawbacks in current cluster analysis techniques is the lack of robustness.

In this study, we present an approach of UAD based on FCM and Voting mechanism which places the emphasis on the robustness of clustering results and the capable of detecting malicious ows without any previous knowledge. The main idea is to combine the notions of Sub-Space Clustering (SSC) (Parsons et al., 2004) and the voting mechanism to produce these multiple independent partitions. We divide the process of intrusion detection into four phases: (1) Traffic detection, (2) Features clustering analysis and (3) Outliers analysis. The structure of UAD system is shown in Fig. 1.

The first step is just to detect the presence of an anomalous time slot where an attack or other kind of traffic anomaly might be hidden. We simply compute the difference between consecutive time slots of some very basic traffic metrics such as number of bytes, packets and IP ows. An abrupt change in any of metrics ags the presence of an anomalous behavior which triggers the second step of the analysis. Modeling anomalies as abrupt changes in network traffic is a standard approach (Thottan and Ji, 2003; Barford et al., 2002). In the second step, we extract the data flow characteristics from the previous step. Before we take the process of cluster
analysis, we should reduce the dimension of the traffic which may divide the high-dimensional data into several sub-spaces. Clustering results are usually divided into three categories: The normal ow, anomaly ows and outliers. Then we submit the data which have been identified as anomaly ows to control center. Finally, we analyze the outliers which generated by the second step. We take process of outliers analysis by voting mechanism to calculate outlier weight of each outlier ow. The outlier ow, whose weight exceeds the noise threshold value, will be defined as an anomaly ow and be submit to be control center. Experimental results show that the FCM-Vote based UAD system can detect malicious automatically and efficiently.

The remainder of this study is organized as follows: Section 2 will describe some applications on intrusion detection which based on clustering algorithm. Following that, We conduct a description of the current challenges and present our solution ideas based on the actual situations in section 3. Section 4 demonstrates the design of the core algorithm solutions in the following section and make a detailed description of the algorithm structure and function of the main module. Section 5 evaluates the performance in different aspects, by the comparing with other intrusion detection approaches to show the advantages in reducing the false positive rate in the authentication algorithm and increasing detection efficiency. Finally, we summarize this study and look into the future trends in section 6.

RELATED WORK AND CONTRIBUTION

Two different Intrusion Detection Systems (IDS) are by far dominant in current commercial network attacks detection: Misuse detection and anomaly detection. Anomaly intrusion detection system means the establishment of the normal pattern outline, if the difference value between the our profile and the normal value exceeds the specified threshold, the system will report an incident. Aim to build the target of detection by intrusion as the destination, anomaly intrusion detection has the advantage that does not rely on signatures. However, how to build targets used to detects abnormal or define the outline of the normal mode is difficult to resolve.

The research on IDS developed rapidly in the past few years. Some approaches of unsupervised detection have been proposed (Portnoy et al., 2001; KDD Cup, 1999; Eskin et al., 2002). The main idea of these methods is based on the clustering. Portnoy et al. (2001) uses a single-linkage hierarchical clustering method to cluster data from the KDD99 (KDD Cup, 1999) dataset, based on the standard Euclidean distance for inter-patterns similarity. Eskin et al. (2002) reports improved results in the same data-set, using three different clustering algorithm: Fixed-Width clustering, an optimized version of k-NN and one class SVM. Leung and Leckie (2005) presents a combined density-grid-based clustering algorithm to improve computational complexity, obtaining similar detection results. In recent years, studies and experiments show that Clustering Fusion can improve robustness, stability and other properties of intrusion detection. Topchy (Guha et al., 2000) found that Clustering Fusion algorithm has a better clustering effect than a single clustering algorithm. This study will introduce a clustering analysis based on Data Mining and the Clustering Fusion, we combine these technologies and implement the traffic clustering in our FCM-vote based IDS.
FRAMEWORK OF RESOLUTION

Traffic detection: We capture ows in consecutive time slots of fixed length $\Delta T$ and analyze the different time-series associated with each aggregation key at each new time slot. An abrupt change which may present an anomalous behavior will trigger the step of the analysis. We model the abrupt changes in network traffic by a standard approach.

Features clustering analysis

Sub-space clustering (SSC): Network traffic is often massive and high-dimensional, for example, the KDD99 data sets used in this study includes 41 features. This fact makes the final clustering results become very unstable because of the interference from irrelevant attributes. It is intolerable to the security workers who are focus on accuracy of the network intrusion detection. Furthermore, the detection of the high dimensional ows increases the time cost.

Therefore, in order to reduce the dimension of aggregated flows by Sub-Space Clustering (SSC), we need to divide subspaces for characteristics based on correlation properties of the network traffic before performing the process of clustering analysis. Without loss of generality, let $S = \{S_1, S_2, \ldots, S_k\}$ be the set of $n$ aggregated-flows. Each flow $S_i = \{A_{i1}, A_{i2}, \ldots, A_{ir}\}$ is described by a vector of $r$ features, e.g., src_bytes, dst_bytes, srv_count and so on. $A_i = \{A_{i1}, A_{i2}, \ldots, A_{ir}\}$ is the corresponding feature space, and $A_i = \{A_{i1}, A_{i2}, \ldots, A_{ir}\}$ will be referred to as the feature space. Through attribute sub-space partition, each subspace of the attributes set is independent which is perfectly adapted for parallel computing architectures. In this study, we show the dimensionality reduction of aggregated flows in KDD99 data set and separate the total 41 features of KDD99 data set into four categories: Features of TCP connection, features of TCP content, features of time-based network traffic and features of host-based network traffic, in which the features of time-based network traffic statistics contains “same host” features and “same service” features. Each category is constructed using only $r$ traffic features, this permits us to analyze the structure of high-dimension aggregated flows using a finer-grained resolution. We refer the categories constructed according to certain rules as the subspace of the total feature space. Instead of directly partitioning the complete feature space, clustering analysis will be carried out within each subspace. We separate the complete feature space into $k$ subspaces and let the set of subspaces $P = \{P_1, P_2, \ldots, P_k\}$ and $P_i = \{S_1, S_2, \ldots, S_n\}$ is one of the subspaces. Each $S_i$ in the subspace $P_i$ is constructed using only $r_{im}$ traffic features. We cluster the flows which are described by a set of features in different subspaces of smaller dimensions and obtain different partitions of the flows in $S$ and separate each subspace $P_i$ into several partitions by clustering analysis.

Clustering analysis: Clustering is one of the critical methods in Data Mining. Clustering algorithm is to separate the certain subspace $P_i$ into the $c$ partitions and $V^*_c = \{V^{(1)}, V^{(2)}, \ldots, V^{(c)}\}$ is the set of $c$ clustering centers in sub-space $P_i$. $V^{(i)}(1 \leq i \leq c)$ is one of the clustering centers. We assume that $S_i = \{A_{i1}, A_{i2}, \ldots, A_{ir}\}$ is one of the feature vectors in subspace $P_i$ and $A_{i1}, A_{i2}, \ldots, A_{ir}$ are $r$ traffic features in subspace $P_i$. In this study, we introduce the idea of Fuzzy C-Means (FCM) algorithm into our process of clustering and let the matrix $U_i$ be the membership matrix of subspace $P_i$ where $u_i$ is the membership threshold. We set the $c = 2$ and define the element $u_{ij}(a)$ of $U_i$ as the membership of the feature vector $S_i$ belonging to the clustering centers $V^{(i)}$ where:

$$u_{ij}(a) \in [0,1], \sum_{a=1}^{2} u_{ij}(a) = 1$$

As we start the process of clustering, we need to initialize the initial clustering centers $V^0_c = \{V^{(1)}, V^{(2)}\}$. The clustering algorithm also need to detect the outliers, we define the set $O_i$ to save the outliers in subspace $P_i$. FCM-based clustering analysis shown in Algorithm 1 presents a pseudo-code for Clustering Analysis. There are several input parameters in this algorithm: Subspace $P_i$, the number of clusters $c$, fuzzy factor $M$ and the iteration termination condition $\sigma$. The objective function of clustering algorithm can be expressed as:

$$J(U,V) = \sum_{i=1}^{c} \sum_{a=1}^{2} u_{ij}(a)^M d_{ii}$$

$$d_{ii} = ||S_i - V^{(i)}||$$

In order to make the objective function $J(U,V)$ of Eq. 1 to be the minimum value, center $V^{(i)}$ and membership matrix $U_i$ can be calculated by Eq. 3:

$$V_{ij} = \frac{\sum_{i=1}^{n} u_{ij}(a)^M S_i}{\sum_{i=1}^{n} u_{ij}(a)^M}$$

If $d_{ii} \neq 0$:

$$u_{ij}(a) = \frac{1}{\sum_{i=1}^{n} \left( \frac{||S_i - V^{(i)}||^2}{||S_i - V^{(i)}||^2 + \sigma} \right)^{1/(M-1)}}$$

135
If \( d_{a} = 0 \):

\[
 u_{i}[i][a] - 1, u_{g}[g][a] - 0, g \neq i
\]

(5)

Algorithm 1: FCM-based clustering analysis

Require:
- Subspace, \( P \)
- The number of clusters, \( c \)
- Fuzzy factor, \( M \)
- Membership threshold, \( T \)
- The iteration termination condition, \( \sigma \)

Ensure:
- The current subspace clustering center set, \( V_{i} \)
- The membership matrix, \( U_{i} \)
- The outlier set, \( O_{i} \)

1: Initialize \( V_{i} \)
2: check the value of \( \sigma \)
3: save the size of as \( n \)
4: for \( i \) in 1:n do
5: \[
 u_{i}[i][a] = \left( \frac{1}{\sum_{g \neq i}^{c} \left( \frac{d_{a}^{-M}}{d_{a}^{-M} - d_{g}^{-M}} \right)^{2}} \right)^{1/M}
\]
6: end for
7: for \( a \) in 1:2 do
8: \[
 v_{a} = \frac{\sum_{i}^{c} u_{i}[i][a]}{\sum_{i}^{c} u_{i}[i][0]^{M}}
\]
9: if \( ||V_{a} - V_{a-1}|| < \delta \) then
10: break
11: else
12: \( V_{a} - V_{a-1} \)
13: end if
14: end for
15: for \( i \) in 1:n do
16: save \( u_{i}[i][a] \) to set \( U_{i} \)
17: end for
18: end for
19: for \( i \) in 1:n do
20: if \( u_{i}[i][1] < \eta \) and \( u_{i}[i][2] < \eta \) then
21: \( O_{i} = O_{i} \cup S_{i} \)
22: end if
23: end for
24: return \( (V_{i}, U_{i}, O_{i}) \)

Clustering fusion based on vote mechanism: In order to obtain the results of the detection, we merge the clustering centers \( \{V_{i}, V_{j}, \ldots, V_{k}\} \) where \( V_{i} = \{V_{i}, V_{j}, \ldots, V_{k}\} \), \( 1 \leq i \leq k \). We design a consensus function based on vote mechanism which combines the clustering results through the co-associated matrix and generates the final results \( R = \{r_{i}, r_{j}\} \). We define \( c = 2 \) as the number of the clustering center and let \( r_{i}(1 \leq i \leq 2) \) be the set of the final results. Let \( U = \{U_{i}, U_{j}, U_{k}\} \) be the set membership matrix of \( k \) subspaces where \( U_{i} \) is the membership matrix of subspace \( P_{i} \). The Vote based Clustering Fusion algorithm selects \( V_{i} = \{V_{i,1}, V_{i,2}\} \) as a category identifier to identify the category of other classes by the largest number of votes. Clustering results are fixed by designing the consensus vote mechanism clustering function. The algorithm of clustering fusion based on vote mechanism can be described as algorithm 2.

Algorithm 2: Clustering fusion based vote mechanism

Require:
- The results of subspace clustering of feature vectors, \( V = \{V_{i}, V_{j}, \ldots, V_{k}\} \), \( V_{i} = \{V_{i,1}, V_{i,2}\} \), \( 1 \leq i \leq k \);
- Subspaces of feature space, \( P = P_{1}, P_{2}, \ldots, P_{P}(1 \leq i \leq k) \);
- Set of membership matrix, \( U = \{U_{i}, U_{j}, U_{k}\} \);
- Membership threshold, \( \eta \)

Ensure:
- The final results \( R = \{r_{i}, r_{j}\} \)

1: Initialize matrix \( B_{n}[2] \)
2: Select the first clustering result \( V_{i} \) as a reference to switch the category label
3: for \( i \) in 1:n do
4: for \( j \) in 1:k do
5: for \( a \) in 1:2 do
6: if \( B_{n}[i][a] \) then
7: \( B_{n}[i][a]++ \)
8: end if
9: end for
10: end for
11: if \( B_{n}[i][1] > B_{n}[i][2] \) then
12: \( r_{i} = r_{i}, S_{i} \)
13: else
14: \( r_{i} = r_{i}, S_{i} \)
15: end if
16: end for
17: return \( R = \{r_{i}, r_{j}\} \);
We initialize matrix B[n] (Casas et al., 2012) to save the votes which belong to a certain category in line 1. Line 2 shows the process of selecting the first clustering results as a reference to switch the category label. Then, we traverse subspaces and select a clustering result as a benchmark and determine which category belongs to by the largest number of votes. Finally, we'll return \( R = \{ r_1, r_l \} \) as the set of fusion results.

**Attack detection:** Finally, the results of clustering fusion will be two clustering centers, respectively implement normal traffic clustering and anomaly traffic clustering. The flows in anomaly traffic clustering can be considered as the malicious flows that may contain network attack and the our FCM-Vote based detection will filter them to ensure security of current network. Then we take the process of outlier analysis.

In the case of outlier analysis, we build a rareness or dissimilarity ranking of flows. Each value of this dissimilarity value corresponds to the cumulative separation of an outlier to the biggest cluster in each of the different subspaces. The biggest cluster in each subspace statistically represents normal operation traffic; hence, those flows that present a bigger dissimilarity value correspond to flows that are very different from the majority of the traffic through the different subspaces. The flows with biggest similarity value with the center of small-size clusters are flagged as anomalies. Since, each subspace in the clustering process may produce outlier, we also need to merge the results of outlier analysis by vote mechanism. We identify the outlier flows by calculating the maximum number of outlier votes. We also need to set the outlier threshold \( \lambda \). If the number of outlier votes is greater than \( \lambda \), the current feature vector will be regarded as anomalies. The process of vote mechanism will refer to algorithm 2.

**EVALUATION**

**Simulation settings:** To validate the proposed FCM-Vote based Intrusion Detection System, we will use KDDCUP99 intrusion detection data sets as experimental subjects. Therefore, in order to avoid dependence on measurement selection and eliminate the impact to the clustering results caused by the differences among the attribute measures, we need to preprocess attribute values. The pretreatment can be divided into two steps: Standardization and normalization. The process of normalization will normalize the standardization value to the interval \([0,1]\).

**Setting of M:** Before carrying out the specific intrusion detection test, we need to test the weight \( M \) which is used to determine the degree of clustering fuzzy. \( M \) is usually taken \([1.25, 2.5]\) in practical applications and (O’Callaghan et al., 2002) proposed the M-optimal selection method based on fuzzy decision. We test the value of preferably M in the intrusion detection system 2000 times and calculate the variance of clustering results in Fig. 2.

Figure 2 shows the variance of the value of \( M \) in the selected range. When \( M \) take the value of 2, the clustering results can be the most ideal, so FCM algorithm will take 2 as the preferred value of \( M \) in our experiment.

**Clustering results:** To show the effect of clustering, we take 10 percent of KDD99 as the subjects and cluster in one of the subspaces, the experimental results are shown in Fig. 3.

Figure 3 presents the detection and automatic characterization of the anomaly traffic which may contain network attacks such as DDoS attack, buffer over flow attacks, network scanning activities and so on in one of the subspaces separated by SSC. We define a curtain

![Fig. 2: Variance of clustering results by different value of M](image_url1)

![Fig. 3: Clustering results of FCM algorithm c](image_url2)
threshold in the outlier filter and anomaly filter and compare the Euclidean distance between feature vector and the clustering center with the threshold to determine the category of the current feature vectors. We demonstrate the outstanding performance of FCM clustering in Fig. 3. Ordinate represents the Euclidean distance between feature vector and the small-size clustering center. On the other hand, Abscissa represents the Euclidean distance between feature vector and the big-size clustering center. We filter the anomaly flows by anomaly filter which identified the small-size clustering results and filter the outlier flows by comparing the Euclidean distance between flow and the big-size clustering center with a certain threshold.

**Intrusion detection results:** To show the performance of our FCM-Vote based IDS, we test 20,000 data and gradually increase the number of data to 100,000, Table 1 shows the results of the experiment.

Our FCM-Vote based IDS improve the performance of the existing anomaly detection by using the Data Mining techniques in traffic clustering that does not rely on previous knowledge. Table 1 shows that our system maintaining a high accuracy.

**Robustness of clustering fusion:** We compared the variance of clustering results after running 20,000 randomly selected set of data through FCM-Vote algorithms and single FCM algorithm 100 times in Fig. 4. Figure 4 shows that the results of comparing the results obtained by different clustering strategies. The fact is that clustering fusion algorithm based on vote mechanism has higher robustness than the single clustering algorithm.

**Rate of false alarm:** We take 20000 samples from Table 1 as the training set of supervised anomaly detection. Then we compared the rate of false alarm between FCM-Vote based detection and traditional supervised anomaly detection on the random data set sample from KDD99. The experimental results are demonstrated in Fig. 5.

![Graph showing variance comparison](image1)

![Graph showing rate of false alarm](image2)

**Table 1: Test results of intrusion detection**

<table>
<thead>
<tr>
<th>Data</th>
<th>Missing rate (%)</th>
<th>False rate (%)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20000</td>
<td>0.75</td>
<td>1.885</td>
<td>97.385</td>
</tr>
<tr>
<td>40000</td>
<td>0.83</td>
<td>1.786</td>
<td>97.404</td>
</tr>
<tr>
<td>80000</td>
<td>0.93</td>
<td>1.362</td>
<td>97.708</td>
</tr>
<tr>
<td>100000</td>
<td>1.23</td>
<td>2.428</td>
<td>96.342</td>
</tr>
</tbody>
</table>

The results of Fig. 5 reveal the fact that the proposed IDS maintains a lower false alarm rate in intrusion detection than the traditional supervised anomaly detection dose. As the traditional anomaly detection relies on the training set, the imperfection of the training set which selected by developer may cause the high rate of false alarm in the case of the limited training sets.

**CONCLUSION**

We proposed a novel intrusion detection approach based on Sub-Space Clustering (SSC) and clustering fusion in this study and implement the new approach in our FCM-Vote based Unsupervised Intrusion System (UIDS) to implement the detection of network attack hidden in the daily traffic and verify the advantages in accuracy, efficiency and robustness of new approach by comparing with the traditional methods of intrusion detection in our experiment. In a diametrically opposite perspective, we place the emphasis on the development of unsupervised knowledge-independent detection algorithms. There are many valuable issues to be studied, such as collaborative intrusion detection system, attack intention recognition algorithms and real-time response of intrusion would be direction of development in the future.

**ACKNOWLEDGMENTS**

This study is supported in part by the Zhejiang Provincial Technical Plan Project (No. 2012C11026-4).
REFERENCES


