Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 4175 – 4180

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

Network Covert Channel Detection with Cluster based on Hierarchy and Density

Qian Yuwen[a]*, Song Huaju[b], Song Chao[a], Wang Xi[a], Leng Linjie[a]

[a]*School of electronic and optical engineering, Nanjing University of Science and Technologe, Nanjing, 210094, china*
[b] *School of Biochemical and Environmental, Nanjing Xiao Zhuang University, Nanjing, 21008, china*

Abstract

In order to solve the problem one detection algorithm can only detect one kind of network covert channel, The detection approach hierarchical and density based cluster was purposed. Because the coding scheme of the covert channel would cause many similar data occurred repeatedly, the detection algorithm cluster based on density can be used to detect several kinds of the covert channels. Moreover, the detection approach cluster based on hierarchy and density is able to tackle of detection a noisy channel. Several detection tests were conducted, the detection results show that the algorithm can work well to distinguish the covert channel from normal network traffic even the noise level was about 20%.

Keywords: network; network security; steganography; covert channel; cluster

1. Introduction

The development of computer networks and intrusion detection systems are forcing hackers to seek more subtle way in stealing information. Network covert channel is a good way for these hackers. A network covert channel is a communication channel that allows two cooperating processes to transfer information on the network in a manner that violates the system's security policy [1]. Network covert channel works not only as a tool for hacking, but also an important approach to transmit secret information such as private keys. Accordingly, detection of the covert channel is a hot research field.

* Corresponding author. Tel.: +086-025-84314567-411;
E-mail address: admon1999@163.com

1877-7058 © 2011 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.proeng.2012.01.639

The initial covert channels were developed in the single systems, in which processes working with classified information level via the use of shared resources. As the increase of the network, more and more network covert channels have been used by attackers to communicate with compromised hosts [2]. Therefore, our focus here is on network covert channels to discourage the use of these channels.

Initially, the detection of the network covert channel uses the signature based approach. People build and update a signatures database. When a signature is found in the traffic monitored, the sensor will alarm. By this method, many network covert channels could be distinguished from the normal traffic. For example, Schear et al. found covert channels in HTTP responses by verifying response header fields against the corresponding object metadata [3]. However, this approach can not deal with those network covert channels which were not founded before. In order to solve the problem, a new detection method based on the abnormal behaviours of the network protocol was purposed. Through monitoring the operators of certain protocols, people can acquaint with the normal operators of some protocols. When there are some abnormal operators, it indicates that some covert communications may be existed. But when being familiar with this detection method of the covert channel, hackers may design the covert channel with operators similar with the normal operators, which may cheat the detectors.

According to these problems, a new detection idea was purposed that was if there are covert channels in network traffic, some regular characters of the traffic would appear due to the modulation of the covert channel, which causes the covert traffic to be different from the normal traffic. Through monitoring these characters, covert channel would be recognized. In general, the widely used approach to find these features of the traffic is the statistical method. Sohn et al. developed a support vector machine based approach to detect covert channels embedded in ICMP echo packets and the sequence number field in TCP header [4]. Borders et al. developed a tool for detecting covert channels over outbound http tunnel based on a similar method [5].

In this exploration, in order to detect several kinds of covert channel with noise, we present our detection algorithm by cluster based on density and hierarchy. Section 2 introduces the detection method. Section 3 validates the effectiveness of our detection method by experiments with other detection algorithms. Finally, section 4, concludes the paper and discusses for our future work.

2. Methods of Detecting Cover channels

In this paper, our focus is on creating a mechanism that can detect covert channels in the network traffic. It has been suggested in the research of Cabuk that detecting a covert timing channel by disclosure the similarity among the inter packet delays. However, the same idea can be used in detecting a covert storage channel. The designer may code the normal value of the field in network packet to covey covert information, which would cause some same values of the one field in the network packet occurs repeatedly. As a result, we can monitor the similarity of the value of some fields in the network packet to decide if there are covert channels exist.

2.1. Density and Hierarchical based Clustering

Density-based clustering algorithms characterize the data distribution by the density of each data object. Clustering is the process of identifying dense areas in the object space. Conventional density-based approaches classify a data object as one of the cores of a cluster if has more than n neighbours within neighbourhood [6]. Clusters are formed by connecting neighbouring core objects and those non-core objects either serve as the boundaries of clusters or become outliers. Since the noises of the data set are typically randomly distributed, the density within a cluster should be significantly higher than that of the noises. Therefore, density-based approaches have the advantage of extracting clusters from a highly noisy

environment. In the following, we list the definitions of terminologies regarding to density-based clustering convenience of presentation.

Definition 1: (ε -neighbourhood of a point) The ε -neighbourhood of a point p, denoted by $N_\varepsilon(p)$, is defined by:

$$N_\varepsilon(p) = \{q \in D \mid dist(p,q) \le \varepsilon\} \tag{1}$$

Definition 2: (core point condition) A point p is directly density-reachable from a point q if $p \in N_\varepsilon(q)$ and $N_\varepsilon(q) \ge MinPts$. MinPts is a minimum number of points in a ε -neighbourhood of the point.

Definition 3: (density-reachable) A point p is density reachable from a point q. If there is a chain of points $p_1, p_2, \ldots, p_n$, $p_1=q$, $p_2=p$ such that is p_{i+1} directly density-reachable from.

However, the performance of density-based clustering is quite sensitive to the parameters of object density, namely, for a complex data set, the appropriate parameters are hard to specify. A density-based clustering algorithm tends to result in a large number of trivial clusters due to the noise.

For solving the problem of trivial clusters, in this work we present an algorithm based on hybrid strategy between the hierarchy and density based approaches. Considering of some cluster points formed by the noise with a low density, we use a hierarchical clustering algorithm generates a hierarchy of nested clusters according their density. Now, we list the definitions of hierarchical clustering based on density.

Definition 4: The density of the cluster C is defined as:

$$V(C) = \frac{|C|}{B(C)} \tag{2}$$

|C| is the number of the points in the cluster C , the density V(C) is the value of |C| divided by B(C), which would be computed by how many unit ball the cluster C covering.

Definition 5: The distance of the cluster C_i, C_j:

$$d = \frac{d_0(C_i, C_j)}{\mathrm{cov}(V(C_i), V(C_j))} \tag{3}$$

The distance of the clusters use Euclid distance, and $d_0 = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \ldots + |x_{im} - x_{jm}|^2}$. x_{ij} is the value of the projection of the core object of the ith cluster on the jth dimension.

Definition 6: difference agrees of the cluster C_i and C_j:

$$CR\left(V(C_i),\ V\ (C_j)\right) = \left|\ V(C_i),\ V\ (C_j)\right| \tag{4}$$

It would be used as a criterion to measure the difference of the distribution of two clusters.

By measuring the density for each data object, density and hierarchical based cluster captures the natural distribution of the data. Intuitively, a group of covert traffic will form a dense area, and these samples with the highest density within the group become the medoid of the cluster. Therefore, where a cluster is formed by noise objects, it has low density. So we can merger some trivial clusters according their densities.

2.2. Detection Algorithm

Detection algorithm of Covert channel is composed of two subroutines, they are clustering algorithm and detecting algorithm. When detecting, the detection algorithm will call the clustering component. Before introducing the detection algorithm, we should investigate the data obtained from the network sensor, which mode can be described as:

$$S = T \times X + W \tag{5}$$

$\boldsymbol{X}$ is the input process of the network sensor, which can capture packets from the network. $\boldsymbol{T}$ is the modulation scheme of the covert channel. $\boldsymbol{W}(\boldsymbol{W}\in R^{m})$ is the noise injected during the transmitting process, $\boldsymbol{S}$ $(S_1...S_n)$ $(\boldsymbol{S}\in R^{m})$ is the observing vector and m is the dimension of feature set, n is the number of the samples. Then, the similarly of samples S can be measured by the Euclid distance:

$$Dist(S_i,S_j)=\sqrt{\sum_{k=1}^{m}(S_{ik}-S_{jk})^2},1\le i,j\le n \tag{6}$$

The clustering algorithm works according the data in the set $\boldsymbol{S}$, and then gets the number of the clusters and the core objects of each cluster. Let $\boldsymbol{R}$ to be the set of core objects, which initial stat of is null. If R_j ($j\in Z^+,R_j\in R$) is a medoid of a sample S_i($i\in Z^+,S_i\in S$), it could be written as $F[S_i]=R_j$. Let k_j to be the number of the samples belongs to the cluster with core object R_j, which initial value would be 0. The clustering algorithm is described as:

Stage one: finding core objects

Step1 select one sample S_i randomly as the core from the data source $\boldsymbol{S}$, if there was a core object R_j in the $\boldsymbol{R}$, satisfied $S_i\in N(R_j)$, then go to *step*2. Or, S_i would be taken as the new core object, $\boldsymbol{R}=\boldsymbol{R}+\{S_i\}$, $R_{jk}=S_{ik}$, k=1…N, goto *step*3;

Step2 let $F[S_i]=R_j$, count the sample of the core objects $k_j=k_j+1$;

Step3 if all the samples of the data source are correspond to a core object, go to *step*4, or go to *step*1;

Step4 Modify the core object set $R_{jk}=\sum_{A}S_{ik}/k_j,k=1...K,A=\{S_i\mid F[S_i]=R_j\}$. Updating scanning times, $c=c+1$. the initial value of c is 1; If the number of the scanning is little than c, go to *step*1, start to scan the data set again.

Step5 When $k_i\ge p$, R_j is a available core object of one cluster, or it is invaluable. Then, all the samples in the clusters responses to an invaluable core object will be allocated to an available core object, namely:

$$F[S_i]=R_j\Rightarrow Si\in N_e(R_j)\Rightarrow dist(S_i,R_j)\le\varepsilon \tag{7}$$

Stage two: Merging clusters

Step6 According to the character of the data to compute the density V_i of cluster R_i and the density V_i of cluster R_i.

Step7 computer the difference of these clusters cov($V(R_j)$,$V(R_i)$), and select the minimum of the them, descript as C_{min}. if $C_{min}<\sigma$, then go to step 8, else go to step 6.

Step8 Computer distance between of C_i and C_j, merging C_i, C_j into a new cluster C_k(k=1,2,3…)

Step9 until all the data was processed, or go to step6.

In the algorithm: ε is the distance of the cluster, p is the least of the number of the points in the neighbour, c is the scanning times, σ is the density of a cluster.

The clustering method could be robust to noise, outliers, and the parameters. It is well recognized that the covert traffic data are usually noisy and the rules behind the data are unknown. The algorithm of cluster is a sub component of the detection algorithm. The recognizing algorithm is described as:

Step1 normalizing, which could be written as:

$$S(i,j)=\frac{S^*(i,j)-S_{\min}(j)}{S_{\max}(j)-S_{\min}(j)} \tag{8}$$

$S^*(i,j)$ is the observing value of the jth feature of the ith sample, and $S(i, j)$ is value after normalizing. $S_{max}(j)$, $S_{min}(j)$ is the maximum value and the minimum value of the jth feature.

Step2 let $\boldsymbol{A}$ to be the unit matrix with the columns m, projecting $S(i,j)$ to the d-dimension(the initial value of d is 1):

$$z_i=\sum_{j=1}^{m}a_j\times S(i,j),\ i=1,2,...,n \tag{9}$$

z_i is the value after projecting $S(i,j)$ to the ith d-dimension.

Step3 call Algorithm 1 to clustering on every dimension, namely, (k_1,…,$k_{m/d}$) = Clustering(Z_1,…,$Z_{m/d}$).

Step4 if k_i=0 ($1 \le i \le m$), there are no any clusters in these features. It may no covert channel in the traffic. Feature S_i would be deleted from feature set. if k_i>1 some covert channel may exist, exit. if k_i=1, d=d+1, go to step2.

Step5 When i= m or P is empty, exit.

If there are several clusters in the lower dimensions, it can be concluded that there may be covert channels existed. When there is only one cluster in the lower dimensions, and there are clusters in the higher dimension, the conclusion is also drawn that the covert channel may be appeared, or there is no any covert channel.

3. Results and Analysis

Our experiments are set up to investigate how our detection tests perform against five typical covert channels: (1) ICMP Shell (2) Covert length channel [7] (3) Covert TCP [8] (4) IPCTC [6] and (5) multiplex covert channel which was mixed by four covert channels before. We collected network packets from these network channels by three steps: First, different computers send covert information with different kinds of covert channels. Second, one sender communicate with one receiver with these five kinds of covert channels. The sender can switch the kind of the covert channel. Before switching, the sender must inform the receiver, so that the receiver can synchronize with it. Third, packets from data set NZIX-II were injected into the covert traffic as the noise [9]. All packets obtained by these three steps served together as the data set.

For detecting the covert channel accurately, the parameters of the detection algorithm should be tuned. At first, it is the detection window, which means how many packets to be deal with in one times of detection process. As suggestion by literature [6], we run the detection one time for 2000 packets. Secondly, we tune the MinPts and Eps, and we found that when MinPts was set to be 300 and the Eps was set to be 1.2×10^{-3}, the detection algorithm works very well.

Because the accuracy of the detection test was also much influenced by the noise injected, we now consider the detection test under the scenarios of noise. We define the noise level is the ratio of the number of normal packets in the traffic and total packets. Then, different levels of the noise were injected into the covert traffic. The perfomance of the detection test is shown as figure 2. For the purpose of examining efficiency the detection test, we select 500, 2000, 5000, 8000, 11000, 16000 as the window sizes. The efficiency of the detector is shown as figure 3.

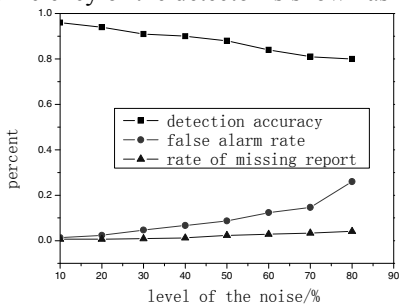

Fig 2 curves of detection performance with different level of noise

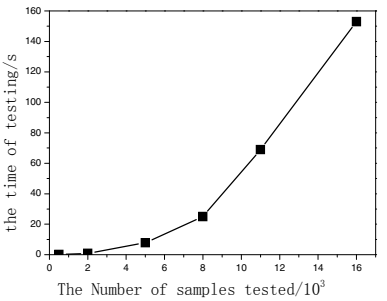

Fig 3 the speed of the detection of the covert channel

In Figure 2, we can see that our detection algorithm works effectively on the covert channels in the case of noise being relatively small. As injecting more noise, the positive ratio varies slightly. But when the level of noise is up to 60%, it drops dramatically, which is because that the samples can not longer be classified into one cluster.

We can know from figure 3, when the size of the window is 2000, the time of testing these records is less than 1 second. Even if the window size is 5000, the time required for testing all the records is only about 10 seconds. But when the detection window is increasing, the detection time increase fast. It is

because that the time of testing the high dimensional data increases in a nonlinear mode as the recorder increases. Therefore, it is favourable for the efficiency and the real-time of the detector when choosing a smaller window size.

Our last experiment is to compare three detection algorithms of the covert channel, which are the approach based on entropy [10], ε-Similarity approach [6] and cluster based on density and hierarchy. The packets produced by five channels, IPCTC, ACK command, Covert TCP, ICMP Shell, multiplex covert channel, are used as the abnormal traffic. These five channels worked with noise level of 20%.

These three detection algorithm is of a higher detection rate toward the simple covert channel, the recognition rates were both more than 90%. But the entropy based method and the ε-Similarity based method are almost out of work to detect a complex covert channel. The density and hierarchy based clustering method is able to woks well on such type of covert channel, which detection rate is 94.11% and false alarm rate is 9%. Hence, the methods based on density clustering not only can accurately detect the traditional simple covert channel, but also can distinguish the complexity of covert channels from normal traffic.

4. conclusions and future work

We introduced an approach based on density and hierarchy to detect covert channels, which make use the character of the covert channel that values of some fields would be similar, but these values would be random in legal traffics. We designed the detection algorithm based on the cluster. According to the algorithm, the detection tool was explored and detection experiments were conducted. Our experimental results show that the approach is capable of detecting different kind of covert channels.

This work was an initial exploration of the detection of the complicated covert channel. Future work will include the investigation of other kinds of covert channel, and add them into the set of detection object to expand of the training set and test set. And we plan to investigate to other detection methods to couture with the covert channel.

Acknowledgement

This work is supported by the National Natural Science Fundamental of China (No.60974129, 70931002).

References

[1] Lampson B W. A note on the confinement problem. Communications of the ACM, 1973,16(10):613-115.

[2] S. Zander, G. Armitage, P. Branch. A Survey of Covert Channels and Countermeasures in Computer Network Protocols. IEEE Communications Surveys and Tutorials. 2007, 9(3): 44-57.

[3] N. Schear et al. Glavlit: Preventing Exfiltration at Wire Speed, Proc. 5th Wksp of Hot Topics in Networks, Nov. 2006.

[4] Sohn T, Moon J, Lee S, Lee D H. Covert channel detection in the ICMP payload using support vector machine. ISCIS, 2003, 828-835.

[5] K. Borders and A. Prakash, Web Tap: Detecting Covert Web Traffic, ACM on CCS. 2004, pp. 110–120.

[6] S. Cabuk, C. E. Brodley, C. Shields, IP covert timing channels: Design and detection. ACM Conference on Computer and Communications Security, Washington, USA: ACM, 2004: 178-187.

[7] Girling C G. Covert Channels in LAN's. IEEE Trans. Software Engineering. 1987, SE-13(2): 292-96.

[8] C. Rowland. Covert channels in the TCP/IP protocol suite. First Monday: Peer-reviewed Journal on the Internet, 2(5),1997.

[9] WAND Research group. NZIX-II trace archive, data available http://pma.nlanr.net/traces/long/nzix2.html.

[10] Gianvecchio S, Wang H. Detecting covert timing channels: An entropy based approach. ACM on Computer and Communications Security. Alexandria, USA: ACM , 2007: 307-316.