RTP timestamp steganography detection method

Wanxia Yang, Shanyu Tang, GuanPing Wang

Abstract

A histogram cosine similarity matching method for real-time transport protocol (RTP) timestamp difference vectors and a clustering method of the area between the best-fit curves of 2 RTP timestamp difference sequences is presented. These 2 methods realize timestamp-based least significant bit (LSB) steganography detection respectively. A clustering analysis of the area between the 5th-degree polynomial best-fit curves with message windows w of 20, 50, 100, and 200 was conducted. The results indicated that when the message window w was 100, the result was the best when the characteristic extraction time was shortest, and the initial clustering accuracy was 84.5%. Through further analysis, the clustering accuracy was increased to 100% in the 2nd round of clustering based on whether the mean distance from a data point in an initial cluster to each cluster center was changed.

Keywords: RTP, timestamp, model-fitting, area difference, cluster, covert channel

1. INTRODUCTION

Covert channels comprise some of the main content of research in the field of information hiding. With the rapid development of network technology and the widespread popularization of the internet, the formation of network covert channels is evitable[1,2]. There is a great deal of streaming media data in the network that needs real-time transmission; the real-time transfer protocol/RTP control protocol (RTP/RTCP) provides important services for this purpose and has become one of the main objects and carriers used in network covert channels. Because each RTP data packet consists of 2 parts, the protocol head and the effective data (payload), the redundant segment or the valid data of in a network protocol can be used as a load to construct a network covert channel.

Because a network covert channel embeds hidden information in the redundant segment of the network protocol, it is strongly concealed and difficult for security equipment and detection equipment

to identify[3]. Even when the covert channel is uncovered, the hidden information is not recovered in transmission due to the special mechanism used in the construction. Second, studies have shown that even if a data packet carries 1 bit of data, the network covert channel can illegally transmit 26 GB of information from a large website within a year; its practical value is extremely high. Evidently, the RTP/RTCP is widely used in network steganography constructs as one of the main streaming media transmission protocols. The means to use the redundancy of the RTP/RTCP for information hiding and detection has become a trend of development and a focus of research.

The structural organization of the thesis is as follows. The next part of the study analyzes the status of related research in constructing storage network covert channels and the detection method. Part 3 presents a detection method based on similarity matching of the histogram of the difference vector between RTP timestamps. Part 4 presents a clustering analysis-based method of detecting the area between best-fit curves for RTP timestamp difference sequences. Part 5 is an analysis of the experiments and results. Part 6 is the summary.

2. RELATED RESEARCH

Network covert channels are usually classified as storage covert channels and time convert channels. The network jitter and transmission delay mainly have greater impacts on time convert channels due to the instability of internet transmission. In this study, only the construction of a storage covert channel and the corresponding detection method is discussed.

At present, storage covert channels are mainly constructed based on the storage segment of transmission control protocol/internet protocol (TCP/IP) packets. The method for detecting concealment primarily focuses on the use of the reserved segment of protocol header, the fill segment, and the covert channel constructed from the unused segment. These segments are often fixed at certain values; only a change in the protocol segment's value needs to be examined to determine whether a covert channel is present; the detection is quite simple. In addition, using IP identification (IPID), changes in information such as the IP lifetime and the TCP initial sequence number have followable patterns; this type of covert channel can be detected through statistical analysis, which performs mode matching with normal data packets [4,5]. Reference [6] used cluster analysis to detect a storage covert channel in a network using the identifier segment and the header checksum segment in IP headers as carriers. Liu et al. [7] extracted a feature of a recursive graph for network storage data sequencing, which effectively completed the detection of a network storage covert channel. While the detection accuracy is high, the detection process is more cumbersome. Reference [8,9] simulated the method of detecting a

network storage covert channel with optimization based on the improved correction entropy but found that this resulted in an increased computational intensity. References [10] and [11] used Markov models to perform detection by analyzing the transition between TCP states for TCP-based covert channels.

Although the current commonly used steganography method, which is based on the insensitivity of TCP-based timestamps to the least significant bit (LSB), can increase the timestamp, a slight time delay in TCP data packet transmission can make the timestamp consistent with the actual time. Second, because many factors can cause the timestamp to change, the network monitoring and detection technique cannot determine all the reasons for a change in the TCP timestamp; therefore, using timestamps can make it easier to hide information [12]. There are very few reported methods for effectively detecting covert channels constructed using this type of method. Additionally, the RTP/RTCP is based on the user datagram protocol (UDP) or the TCP [13,14] to ensure real-time data transmission and protocol control, as shown in (Fig. 1). There are even fewer results for hiding and detecting information in timestamps in the RTP/RTCP. Therefore, this paper presents a detection method based on difference histogram similarity matching using RTP timestamps and LSB steganography and a method of detecting based on clustering using the area difference between 2 best-fit curves.

3. DIFFERENCE HISTOGRAM SIMILARITY MATCHING

3.1 Difference Histogram of the Timestamps in an RTP Packet Flow

The timestamp field of an RTP packet for the streaming media transmission protocol occupies 32 bits, which is the first sampled byte of RTP packet data. It is mainly used to reconstruct the sequence of received data packets on the receiving end. The initial timestamp is random, and the subsequent timestamps increase progressively. For instance, with voice packets, the timestamp increase is equal to the packet interval times the sampling rate; if the packet interval is 30 ms and the sampling rate is 8000 Hz, the timestamp of each successive packet increases by 240. Evidently, the RTP packet timestamp increases monotonically. Fig. 2 shows the original timestamps of 10001 RTP packets extracted in this study; the results show linear growth. However, the information hidden in the timestamps is bound to change their original distribution; this paper uses difference histograms to express the change in an RTP packet's timestamp before and after steganography. Fig. 3 (a) and (b) are the difference histograms of an RTP packet's timestamp for the least significant bit before and after steganography. It is evident that the difference is large, which means one can use the histogram similarity matching method to detect the RTP packet's timestamp in LSB steganography.
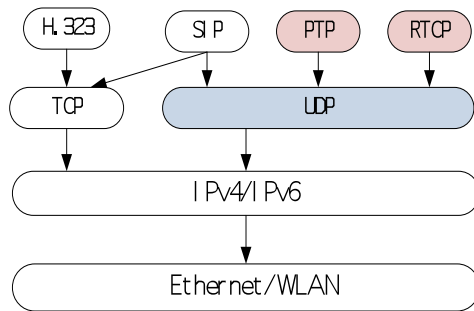
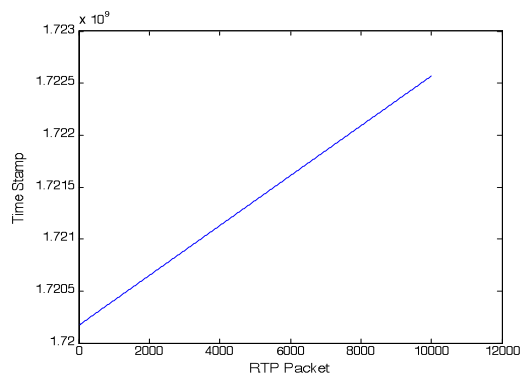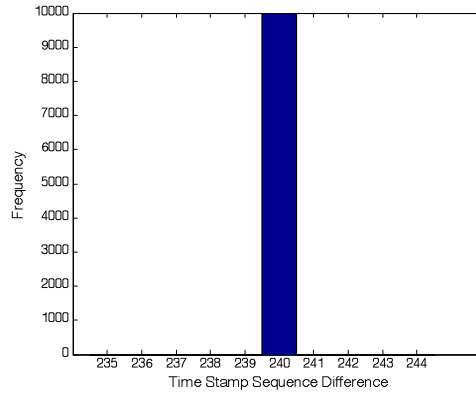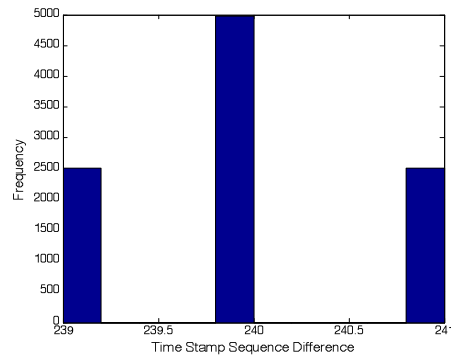**Figure 1.** Structure of the RTP/RTCP

Figure 2. Original RTP packet flow timestamps

(a)

(b)

Figure 3. Difference histograms for RTP packet flow
timestamps before and after steganography

3.2 Difference Histogram Similarity

In the analysis of an RTP packet data stream, the timestamp of each RTP packet in data stream is called τ_i ($1 \leq i \leq N$), and the adjacent timestamp is differentiated to obtain the difference sequence D = d1, d2, …, dN-1, where

$$d_i = \tau_{i+1} - \tau_i. \qquad (1)$$

Then, D is segmented based on a certain size, and the data in each segment D_i are detected. The frequency of the timestamp difference is used as the vertical coordinate of histogram, and the difference sequence serves as the horizontal coordinate to calculate the histogram, which forms a vector u∈R^{10} of detected objects from the histogram. Cosine similarity is used to calculate the cosine of 2 vectors in measuring the difference between the 2 sub-sequences. The closer the cosine is to 1, the closer to 0 degrees the angle is, which indicates that the 2 vectors are similar. Compared to distance measurement, cosine similarity focuses more on the difference of 2 vectors in terms of their direction, but not their distance or length. The directionality in the difference timestamp sequence changes over time.

Therefore, this study uses cosine similarity to measure the similarity of 2 timestamp sequences to determine whether steganography is present in the timestamps. The formula is as follows:

$$S = \cos(\theta) = \frac{\sum_{i=j=1}^{n}(d_i \times d_j)}{\sqrt{\sum_{i=1}^{n}(d_i)^2} \times \sqrt{\sum_{j=1}^{n}(d_j)^2}} \qquad (2)$$

where d_i and d_j are the vectors of the 2 sequences and n is the length of a sequence. The specific calculation steps are as follows:

(1) Calculate the timestamp difference sequence for 2 adjacent RTP packets, $D = d_1, d_2, \ldots, d_{N-1}$.

(2) Divide D into 250 sequences of length 40 and detect the data in each sequence D_i. The frequency of the timestamp difference is taken as the vertical coordinate of the histogram, and the difference sequence is the horizontal coordinate in calculating the histogram of its distribution, which forms a vector $u \in R^{10}$ of the detected object from the histogram.

(3) Calculate the cosine similarity S of the 2 vectors. When the similarity is $S \geq 0.95$, the 2 vectors are of the same type, and vice versa.

4. MODEL FITTING AND CLUSTERING ALGORITHM IMPLEMENTATION

4.1 Modeling with polynomial fitting

The RTP timestamp increases monotonically by a fixed increment; therefore, its difference sequence is constant. Additionally, the steganography changes the original distribution of the timestamps, which means that the difference sequence is no longer constant. This difference can be clearly expressed by means of model fitting, as shown in Fig. 4. The x-axis is the number of a channel message in the sequence, and the y-axis is the difference in its timestamp.

Assuming the timestamp difference sequence of a w+1 message window in a certain communication process is (d1, d2, ..., dw) (w>=1), a set of points to be fitted, P={i, d$_i$) | i=1, 2, ..., w; w≥1}can be obtained; P is the set of time differences in the sequence for the message's serial number and transmission. A polynomial model of the inter-channel RTP timestamp difference can be obtained by polynomial fitting. The polynomial fitting model (shown in Fig. 4) illustrates the effect and variation of the timestamp due to the steganography nicely. Additionally, the results serve as the clustering subject, which determines whether the steganography is present by means of a clustering algorithm.
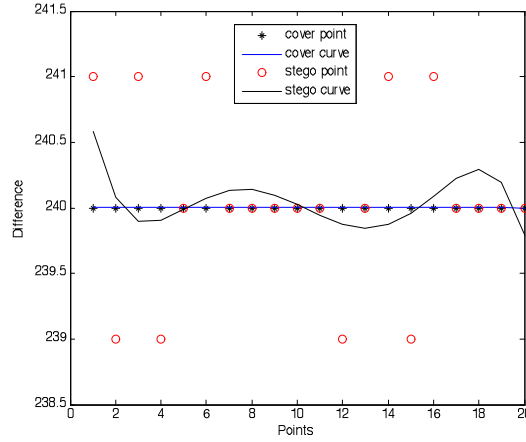
Figure 4 . The best-fit curves of the RTP timestamp difference
sequences for normal and steganographic channels

Polynomial fitting optimization

Optimizing the curve fitting process, that is, finding an optimal function that matches the set of data, is critical. One of the most commonly used approaches is the least squares method[15]. The measured timestamp difference sequence data are labeled $\{d_k\}(k=1,2,3,\ldots,w)$, where w is the of data points in the window. The fitting function is represented by a polynomial $\hat{d}_k$; then,

$$\hat{d}_k = a_0 + a_1 d + a_2 d^2 + \ldots + a_k d^k = \sum_{j=0}^{k} a_j d^j \qquad (3)$$

where $j = 0,1,3,\ldots,k$ and $\hat{d}_k$ is the estimated value of d_k. In this case, the square of the distance between the observation point and the estimated point is $(\hat{d}_k - d_k)^2$. The core idea of the least squares method is to minimize the weighted sum of the squares of the residual error (or deviation) E_k between each point of model and the actual observed values. That is, the value of $E_k = \sum_{k=1}^{w} (\hat{d}_k - d_k)^2 = \sum_{k=1}^{w} (\sum_{j=0}^{k} a_j d^j - d_k)^2$ is minimized to obtain the parameter's value [15]. The partial derivative of E_k with respect to a_j is taken, as shown in equation (4), to obtain a linear equation of degree k+1.

$$\frac{\partial E}{\partial a_j} = 2\sum_{k=1}^{w} d^j (\sum_{j=0}^{k} a_j d^j - d_k) = 0 \quad j = 0,1,2,\cdots,k \qquad (4)$$

The k+1 undetermined coefficients a_j (j=0, 1, 2, …, k) can be obtained by solving the set of equations.

Selection of the polynomial's order

Selecting the order is of the utmost importance in polynomial fitting. Usually, the higher the order is, the more accurate the fit is and the smaller the residual error is. However, as the order increases, over-fitting may occur, and the algorithm's complexity increases, which may even lead to a reduction in the number of clusters due to high dimensionality. Therefore, a higher order may not lead to a better result. Only the appropriate order of the polynomial can accurately reflect the variation tendency of the timestamps and correctly determine the presence of hidden information through clustering in real time. As shown in Fig. 5 (a) and (b), the accuracy of the 3rd-order polynomial fitting is too low, and the accuracy is higher at the 7th order than at the 5th, but this takes too long. Fitting to a 5th-order polynomial not only clearly reflects the difference in the timestamps before and after steganography but also takes the least time. After the time and accuracy of best-fit polynomials of different orders were compared experimentally, the 5th order was selected for this experiment.
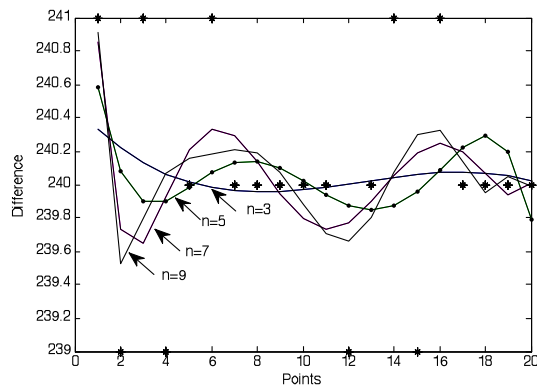
(a)
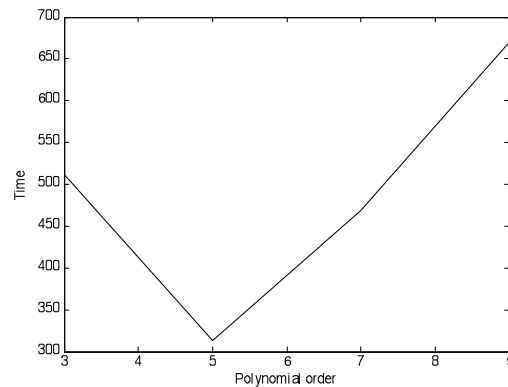
(b)

Figure 5.   Best-fit curves and fitting times for different orders

4.2 Selection and Extraction of Cluster Characteristics

Polynomial coefficients of the best-fit curve

The result of polynomial fitting depends on the coefficients. Table 1 provides the coefficients that were constructed using 2 sets of channels: a normal one, c(x), and a steganographic one, h(x). It is evident that the coefficients of the best-fit polynomial for the normal channel are different from those for the steganographic channel. However, the coefficients of the normal channel and the highest-order coefficient of the two channels are small and 5 orders of magnitude smaller than the coefficients of the lowest order. The value of a polynomial depends mainly on the highest term, and the weight of each coefficient is different; therefore, direct use of polynomial modelling for clustering is not appropriate.

Calculation of the area between the best-fit curves

It was shown above that polynomial fitting cannot be directly used for clustering. However, by analyzing and comparing the best-fit curves (Fig. 4) of the normal and steganographic channels, it was found that the area enclosed by c(x), h(x), x=1 and x=w had a larger difference; this difference can be used to distinguish between normal and steganographic channels. To better reflect this difference, the following formula is used to calculate the area under the absolute value of the difference between two curves; this integral is used as the clustering subject.

$$S_d = \int_1^w |c(x) - h(x)| dx. \qquad\qquad (5)$$

Table 1: c(x) and h(x) fitting coefficients

Order number model	5	4	3	2	1	0
c(x)	-4.37e-18	2.166e-16	-4.06e-15	3.62e-14	-1.50e-13	240
h(x)	-4.41e-05	0.002	-0.04	0.37	-1.36	241

4.3 Clustering algorithm implementation

The core of clustering is to select a method for measuring the similarity or distance and to construct an objective function[16,17,18]. K-means clustering is a clustering algorithm that uses the square of the error, a criterion function, and the most commonly used Euclidean distance to measure similarity. This

algorithm is simple, effective, and flexible enough for processing large data sets. Therefore, this study used the results of k-means clustering as basic information to calculate the average distance to the center; then, a determination was made based on the change in the average, and a more accurate clustering result was obtained. It was found from experiments that the average distance from a

data point to the center remained constant after clustering in a normal channel but varied in a steganographic channel. As shown in Fig. 6, a covert channel can be accurately identified accordingly. The specific steps are as follows:

(1) First, calculate the area between the best-fit curves $\{S_d\}$ for the timestamp difference sequences between normal channels and between normal and steganography channels for window length w.

(2) Select initial values from the clustering subject $\{S_d\}$ multiple times, and search for the k most appropriate centers for each initial value $\{C_1, C_2, \ldots, C_k\}$.

(3) Calculate the residual distance between each data point and each center $R(i, k)$ using the following equation; cluster data points that are closest to the center point into the group represented by that center:

$$R(i,k) = \sqrt{(S_{d_i} - C_1)^2 + (S_{di} - C_2)^2 + \cdots + (S_{di} - C_k)^2} \qquad (6)$$

(4) Use Equation (7) to calculate the center of each group,

$$\mu_k = \frac{1}{N_k} \sum_{S_{di} \in c_k} S_{di} \qquad (7)$$

where N_k represents the number of data points in group C_k and S_{di} represents all data points in C_k.

(5) Repeat steps (3) and (4) until the error squared and the criterion function start to converge; that is, the value of the cluster's center no longer changes. By so doing, the cluster centers μ_k of each group of data and the distance from each point to the center of each cluster R_k can be obtained.

(6) Calculate the average distance R_k between each data point and each cluster center μ_k using Equation (8),

$$M_i = \frac{1}{N_k} \sum R_k \qquad (8)$$

where i=1, 2, …, n and N_k is the number of centers in group μ_k.

(7) Compare the value of M for each data point to the value of M for a normal data point; if they are the same, then the channel is normal; otherwise, the channel is covert.
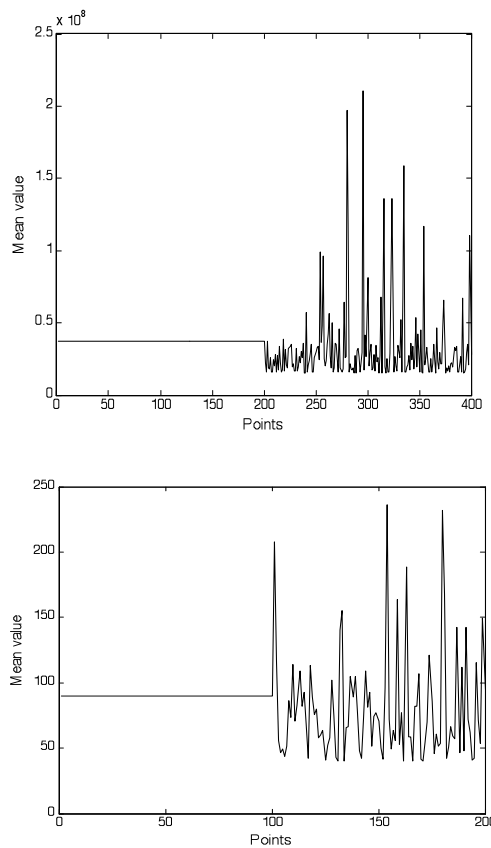

Figure 6. Variation of the average distance between each point and each cluster center when the window w is 50 and 100

5. EXPERIMENTAL RESULTS AND ANALYSIS

5.1 Experimental process

The data collection platform used in this experiment includes a switch used in a local area network (LAN), a normal user, and a pair of covert communicators. All the local network traffic goes through the switch; by configuring a switch mirror port, all data packets entering and leaving the mirror port are copied and sent to a monitoring port. The experimental data are obtained from the monitoring port using the WinPcap data collection structure in a Windows environment. First, the network card is put into promiscuous mode to receive all the data packets on the network. The main task is to filter, statistically analyze, and cache the RTP packets in the netgroup packet filter (NPF) through the definition of rue base. The specific steps are as follows:

(1) Normal data: 10001 normal channel timestamp segments of RTP data packets are collected using the above described experiment platform.

(2) Hidden data: by random bit replacement of the LSB of the RTP timestamp header field on both sides of the hidden communication using the collection platform, 10001 hidden data packets are obtained.

(3) The difference histogram of the RTP timestamps of normal and steganography data packets are calculated using the method and steps in section 3.2; from this, a vector for the monitoring subject, $u \in R10$, is generated. Then, the cosine of the 2 vectors is calculated, and cosine similarity is applied. If the similarity to a normal data packet is less than 0.95, then the packet is steganographic. The experimental results are shown in Fig. 7.
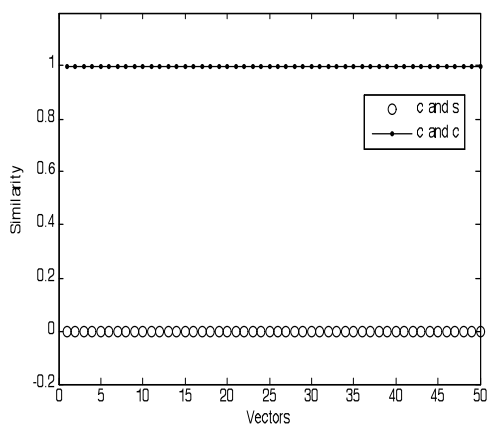

Figure 7. Cosine similarity

(4) The results are fitted to a 5th-order polynomial, and the RTP timestamp sequence is optimized using the method in section 4.1; the area between the 2 best-fit curves is selected as the clustering subject.

(5) Use the steps in section 4.3 to perform clustering. In the experiment, the value of the curve-fitted message window w has a significant influence on the modeling and clustering results; this study includes experiments for w values of 20, 50, 100, 150 and 200. The focus of the comparison is on the time required to extract the characteristics of each cluster, the rate of change of the average distance between a data point and each cluster center,

the accuracy of the initial and final clustering results, and the rate of false positives in the initial clustering results for different values of w. The results are shown in Table 2. In addition, the curves of initial and final clustering results are performed to compare the detection results of in Table 2. As shown in Fig. 8. And the training time 13.2s of the algorithm in this paper is compared with the training time 14.7s of the algorithm in [6]. The detection algorithm in this paper has the advantage of in training time.

5.2 Analysis of results

It can be seen from Fig. 7 that the cosine similarity of the RTP timestamp difference histogram vector is 1, whereas the cosine similarity between the RTP timestamp difference histogram vector for the normal and steganography channels is 0; therefore, the proposed method is excellent at detecting steganography channel timestamps.

Table 2: Characteristic extraction time and clustering results

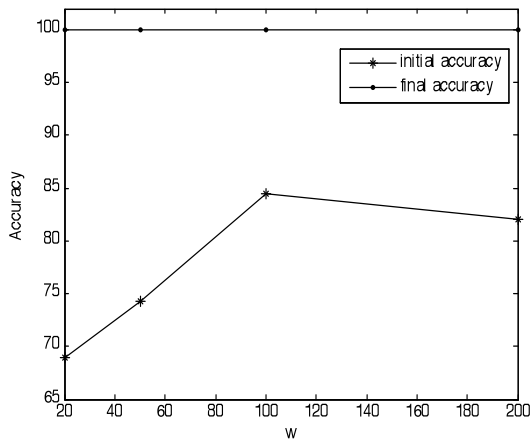| w parameter	20	50	100	200
time T/s	878.5	313.4	153.8	62.3
Mean change %	91.6	87.5	65.6	90
initial accuracy %	68.9	74.3	84.5	82
final accuracy %	100	100	100	100
initial false positives %	50.7	34	23.7	26.5

Figure 8. comparison curve of initial and final clustering results

It can be seen from Table 2 that the proposed method for clustering RTP timestamp difference sequences based on the area between 2 best-fit curves can used for detecting steganography channels using timestamps. When the message window w is 100, the characteristic extraction time is quite short, the probability of a change in the average distance between each data point and each cluster center is the lowest, the accuracy of the initial clustering is the highest, the false positive rate

is the lowest, and the clustering result is the best. With different values of w, the results of the 2nd round of clustering in this study are all 100% accurate. Moreover, the experimental verification showed that the window w cannot be too small; using a value of w that is too small can lead to an increase in the number of clustering subjects, which reduces the clustering accuracy. At the same time, w cannot be too large; using a value of w that is too large increases the difficulty of modeling. Therefore, the suitable range for w is 50-200.

6. CONCLUSIONS

This paper presented a histogram-based cosine similarity matching method for RTP timestamp difference vectors and a clustering method based on the area between the best-fit curves for 2 RTP timestamp difference sequences;

timestamp-based LSB steganography detection was realized. The first method is simple, reliable, and very accurate, whereas the 2nd method produces very good initial clustering results when the window w is 100; it reached an accuracy of 84.5%. Based on the initial clustering results, a 2nd round of clustering was performed based on whether the average distance between each pair of clusters had changed. The results of the 2nd round of clustering were excellent for the different windows, and 100% accuracy was reached.

Innovation1: Proposed a method for applying clustering to RTP timestamp difference sequences based on the area between 2 best-fit curves; simply and effectively realized RTP timestamp-based LSB steganography detection.

Innovation 2: Based on the average distance between the center of the initial cluster and those of each of the other clusters, a 2nd round of clustering was performed; this increased the accuracy to 100%.

AUTHOR'S CONTRIBUTIONS

WY and ST designed the algorithm and conceived the experiments. WY performed the experiments. WY analyzed the data and wrote this paper. All authors read and approved the final manuscript.

COMPETING INTERESTS

The authors declare that they have no competing interests.

REFERECES

1.X Duan, H Song, C Qin and M.K. Khan. "Coverless Steganography for Digital Images Based on a Generative Model." Computers, Materials & Continua, vol. 55, no. 3, pp. 483-493,2018.

2. Ke Zhou, Wenjie Liu, Kun Tang, Ping Huang, Xubin He,"Alleviating Memory Refresh Overhead via Data Compression for High performance and Energy Efficiency,"*IEEE Transactions on Parallel and Distributed Systems*, Print ISSN: 1045-9219. DOI: 10.1109/TPDS.2017.2763141.

3.Y Cao, Z Zhou, X Sun and C Gao. "Coverless Information Hiding Based on the Molecular Structure Images of Material," Computers, Materials & Continua, vol. 54, no. 2, pp. 197-207, 2018.

4. WANG Peng, LAN Shao hua, ZHANG Jing, LIU Guangjie,"Covert timing channel method based on TCP timestamp option,"*Journal of PLA University of Science and Technology (Vatural Science Edition)*,Vol.16,No.2,pp.18-23,2015.

5. LIU Guang-jie, DAI Yue-wei, YE Deng-pan, GU Xia, WANG Zhi-quan, "Steganlaysis Against an IPIDs-based Protocol Steganography," *Computer Engineering*, Vol.33,No.24, pp.217-219,2007.

6. YUAN Jian, WANG Tao, "Detection Algorithm Based Network Storage Covert Channel Clustering Analysis," *Computer Engineering*, Vol.41,No.9,pp.168-173,2015.

7. LIU Biao, LAN Shao-hua, etc, "Approach Based on Recurrence Plot to Detect Covert Timing Channels," *Computer Science*, Vol.42, No.2, pp.114-117,2015.

8. LIANG Jun,"The Improvement and Optimization of the Network Storage Covert Channel Test Simulation," *Computer simulation*, Vol.34,No.2, pp.406-409,2017.

9. Jiangtao Z., Guan   L.,Yuewei D, "A Covert Channel Detection Algorithm Based on TCP Markov Model," Multimedia Information Networking and Security (MINES), 2010 International Conference on IEEE, nan jing china , 2010,pp893-897.

10. Fahimeh Rezaei, Michael Hempel, and Hamid Sharif, "Towards a Reliable Detection of Covert Timing Channels over Real-Time Network Traffic," *IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING*, Vol.14, No.3, pp.249-264,2017.

11. Jiangtao Zhai, Guangjie Liu, Yuewei Dai," Detection of FTP Covert channel based on Markov Model," *International Journal of Advancements in Computing Technology*, Vol.4,No.17,pp.410-417,2012.

12. Wand Changda，Huang Lei，and Liu Zhifeng," Information Hiding Algorithm of IP Covert Timing Channels and Its Performance Analysis,"*Journal of Computer Research and Development*, Vol.53,No.5，pp.988-999,2016.

13. Huang Yongfeng,Li Sunbin, "Network Convert communication and its detection technology," First edn. Tsinghua university press, Beijing, 2016.

14. Wang Yongji, Wu Jingzheng, Zeng Haotao et al, "Covert channel research," *Journal of Software*, Vol.21,No,9, pp.2262-2288,2010.

15. Ning-Bo Wang, Li-Xiang He, Wei-Xin Ren,Tian-Li Huang, "Extraction of influence line through a fitting method from bridge dynamic response induced by a passing vehicle," *Engineering Structures*,pp.648-664,2017.10.1016/j.engstruct.2017.06.067

16. Yuanyuan Ma, Xiangyang Luo, Xiaolong Li, Zhenkun Bao, and Yi Zhang,"Selection of Rich Model Steganalysis Features Based on Decision Rough Set α-Positive Region Reduction," IEEE Transactions on Circuits and Systems for Video Technology, 2018, published online, DOI 10.1109/TCSVT.2018.2799243..

17. AM Abdalla, SM Abodahab, A Khan, "Rotational effect on thermoelastic Stoneley, Love and Rayleigh waves in fibre-reinforced anisotropic general viscoelastic media of higher order,"*Computers, Materials & Continua*,Vol.53,No.1, pp.52-72,2017.

18. Jinwei Wang, Ting Li, Yun-Qing Shi, Shiguo Lian, Jingyu Ye,"Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics," *Multimedia Tools and Applications*,Vol.76,No.22,pp.23721-23737, 2017.

Wanxia Yang (First author) is a PhD research student majoring in Data Communications Security, supervised by Prof. Shanyu Tang who is Chair Professor of Information Security in the School of Computing and Engineering at the University of West London, St Mary's Road, Ealing, London W5 5RF, UK (E-mail: Shanyu.Tang@uwl.ac.uk; Tel: +44 (0)20 8231 2948; Fax: +44 (0)20 8231 2402).