Volume : 4 | Issue : 4 | April 2015 • ISSN No 2277 - 8179

Video Content Leakage Detection by Matching Traffic Pattern

Engineering

KEYWORDS : Streaming content, leakage detection, traffic pattern, degree of similarity.

Blessey.P.M PG Scholar, Anna University, S.A. Engineering College, Department of Computer Science and Engineering

Divya.P PG Scholar, Anna University, S.A. Engineering College, Department of Computer Science and Engineering

ABSTRACT

In recent years, streaming of multimedia services and applications has been rapidly increasing. Avoidance of the undesirable content-leakage by providing the trusted video delivery has been a critical mission. Conventional systems have concentrated on the leakage issue during the privacy preservation of the user by examining techniques based on the streamed traffic observation throughout the network. These conventional systems retain a high detection, which considerably degrades the varying length of the video due to the network traffic variation caused by the packet loss or network delay. To overcome such an issue, we propose a scheme called novel content-leakage detection, which provides high robustness against the varying video length. Comparing videos of varying length helps to determine the similarities and the differences between the compared videos. Hence the varying video length detection using the novel content-leakage detection scheme has improved the detection performance. Therefore a well-organized and effective proposed scheme estimates the packet loss, delay variation and varying length of the video with the help of test bed experiment.

INTRODUCTION

The rapidly developing technique that provide both wired and wireless high-speed networks is known as the broadband technology that has been progressing and popularized real-time video streaming applications and services over the Internet. Applications such as YouTube and Microsoft network provides service with various content ranging from entertainment data such as video clips, music clips, Tv clips, to education, and daily news feeds using technologies like streaming transmission application that has been supplied to the worldwide users. Moreover, the communications in intra-company networks or using virtual private networks (VPNs) is a streaming technique that has been increasingly used for the advancement in business. The critical challenge is that the video bit stream protection from the unauthorized users. A popular technique called the Digital Rights Management (DRM) is used to prevent the bit stream from illegitimate users and to prevent the author's copyrights. DRM techniques often make use of digital watermark or cryptographic techniques. Moreover, these approaches don't contain any important consequence as a effect of content re-distribution, decrypted or restored by the malicious node with the authorization of the legal users. In addition to this, re-distribution of the content is no longer complicated with the use of peer to peer streaming software. Therefore, traffic streaming would lead to leakage in the P2P networks.

On the other hand, leakage of the streaming content to the external network can be avoided by using the packet filtering technique in the firewall-equipped nodes. During this method, the packet header contains the information such as source and destination addresses (IP address), Port number of the outgoing traffic and the protocol type, of streamed packets that has been monitored. If the packets which have been inspected don't validate the filtering policy, then the packets get blocked and dropped. Hence, the packet filtering approach alone cannot able to avoid the content streaming leakage entirely since the header information from the unauthorized users is not mentioned in advance and can be spoofed easily.

LEAKAGE DETECTION

In this section, we discuss about the various situations that cause these video leakages and represent the existing traffic pattern based leakage detection technologies.

A. Scenario of Content leakage

Peer to Peer streaming software has been demanding software owing to the attractive streaming delivery of movies. Types of information is distributed throughout the network is improved by these kind of technologies. The situation and the reasons for a typical content-leakage have been described by the following steps as depicted in Fig. 1. In a secure network, a normal user receives a content of streaming data from the server. Hence, by using P2P streaming software, the normal yet malicious users redistribute the streaming content to a non-regular user external to its network. A watermarking and Digital Rights Management (DRM) technique barely detects or blocks the content leakage.

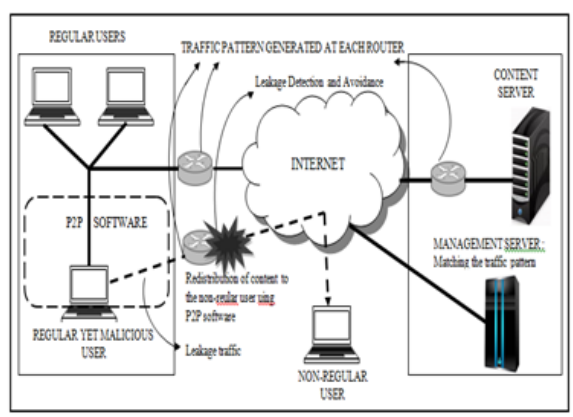

Figure 1: Leakage detection system

B. Leakage detection techniques

During the video streaming process, the varying sum of traffic emerges as a distinct waveform specific to the content. As a result of supervision, the retrieved information from distinct nodes in the network helps to detect the content-leakage. The network topology of the proposed leakage detection technique is explained in diagrammatic representation for easier understanding. The traffic pattern generation engine and the traffic pattern matching engine are the two main components of the network topology. The traffic pattern generation engine is installed at each and every router within the network where each router in the network topology monitors its traffic size and then traffic pattern is generated. The traffic pattern matching engine is employed in the management server, such that this engine employs a matching process to figure out the similarities between the traffic patterns and detects the content leakage depending on certain specific conditions. Therefore, the result is reported to the target edge router to block the leaked traffic.

C. Pattern generation algorithm

The performance of the traffic pattern generation process in a conventional system is discussed in detail. Traffic pattern generation process mainly depends on a either time slot-based algorithm or the packet size-based algorithm. Time slot-based algorithm is an easy solution to generate traffic patterns. Time slot-based algorithm is a simple and easy solution for the traffic pattern generation by adding the total number of traffic arrival for a particular time period. During this, packet delay may take place and there is a possibility that packets are stored in the subsequent slot, xi+1 rather than the primary slot xi. Thus, the accuracy of matching the traffic pattern is affected as a result of packet distortion due to delay and jitter. Furthermore, packet loss affects time slot-based algorithm. Packet size-based algorithm assigns slot for summing the amount of traffic arrived by monitoring the packet size. This algorithm mainly depends on the order of the packet arrival and its packet size. Therefore, it provides robustness to packet delay and jitter, but not to packet loss. The generated traffic pattern is expressed as an N-dimension vector as follows,

$$X_N = \left(x_1 x_2 \ldots\ldots x_N\right)^T$$

Where xi represents the volume of the ith chunk and N indicates the total number of chunks. Fig. 2. Explains both the time slot-based generation process where the time-slot, t is set to 0.1 milliseconds and packet size-based generation process slots are assigned by summing the total of arrival traffic by examining a packet size less than 200 bytes.

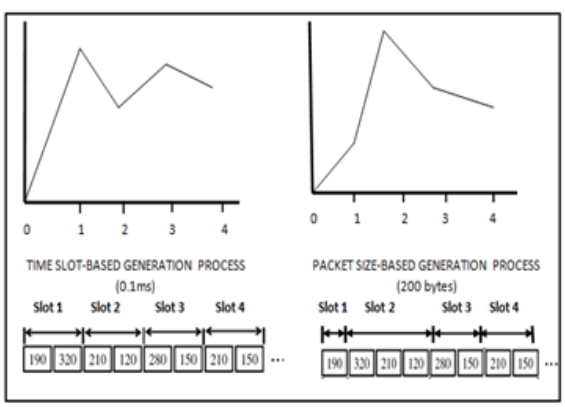

Figure 2: Generated Traffic Pattern Using Time-slot and Packet-slot based generation process

D. Pattern matching algorithm

During the pattern matching process, the degree of similarity defines the similarity range among the patterns. The original traffic pattern in the server-side is expressed as

XS = (x1; x2; :::; xS)t

The traffic pattern in the user-side is expressed as

YU = (y1; y2; :::; yU)t

Where S and U represents the number of slots and S>U server-side monitoring length is greater than the user-side. Comparison methodology is categorized as three stages.

At first, the size of the window is set as U that cuts off partial pattern, XU, from the traffic pattern of the server-side XS.

Compute the resemblances between the user-side traffic pattern YU and the partial pattern XU.

Finally, the window has been moved from left to right by one slot

Repeat all the above procedure windows get to the rightmost part of the server-side pattern Hence, the similarity between the patterns acquired is (S U + 1) values. Moreover, the maximum value is retrieved and the degree of similarity of the compared videos is represented. The other pattern matching algorithm is the cross-correlation matching algorithm which computes the coefficient that provides the degree of similarity among different traffic patterns and the dynamic programming matching algorithm which uses the distance between the different patterns to find out the similarity measure.

Time slot-based Traitor Tracing (T-TRAT), Packet size-based Traitor Tracing (P-TRAT) and Dynamic Programming based Traitor Tracing (DP-TRAT), are the conventional systems that have been described in Table I. T-TRAT uses the time slot-based pattern generation algorithm due to packet delay and jitter, which deteriorates the user-side traffic pattern. P-TRAT and DP-TRAT makes use of traffic pattern generation methods depending on packet size rather than time-slot. Thus, P-TRAT and DP-TRAT provides robustness against packet delay and jitter. The pattern recognition method used in the cross-correlation coefficient is influenced by packet loss. Dynamic programming matching algorithm not only provides robustness against the packet loss, but also the packet delay and jitter.

In this, we mainly focus on the illegal re-distribution of the data in a continuous flow, such as video or audio through the internet by an authenticated user to the external network. The existing approaches, observe information at each node in the streaming path. This information helps to produce a traffic pattern that looks like a unique waveform per content just like the fingerprint. The generated traffic pattern doesn't require the information in the packet header but the user privacy is retained by evaluating and comparing the traffic patterns that helps in leakage detection. However, the performance of the leakage detection is significantly degraded due to the presence of videos of varying length. Therefore an innovative leakage detection method is developed which is much more robust against the varying video length. Comparing the videos of varying length, the relationship between those different streamed videos is determined which indeed finds out the decision threshold enabling accurate leakage detection in video streaming with different lengths.

ENHANCED DETECTION TECHNIQUE

Compared to the previous detection techniques, the dynamic Programming based Traitor Tracing (DP-TRAT) is the best solution that provides high robustness to certain drawbacks such as jitter, packet loss and the packet delay. However, the accuracy of the DP-TRAT is affected due to the presence of videos of varying length. The varying length videos on the network may cause certain issues such as utilizing certain content. Here, we propose a new threshold determination method depending on the approximation of exponential. The proposed detection technique known as the exponential approximation –based threshold determination and leakage detection is composed of two steps such as threshold detection process and leakage detection.

During this exponential approximation, video segment of varying length with its corresponding traffic patterns is generated. The degree of similarity is determined, compared with the original traffic pattern. Thus we present the exponential approximation with the distributed sampling result is expressed as $f(x)=\exp(\alpha.x+\beta)$, where α=n.C-B.D/n.A-D2 and β=A.B-C.D/n.A-D2.. This exponential curve is calculated based on the least-square method and the curve represents the degree of similarity with the original video of length x. The equation for tuning the degree of similarity is expressed as $Y'=Y/f(x)'$ Where x is the size

Volume : 4 | Issue : 4 | April 2015 • ISSN No 2277 - 8179

of the pattern and Y is the degree of similarity, i.e. The degree of similarity resulted in the matching technique is divided by the exponential approximation. On the other hand, the original traffic pattern is evaluated with the video portions of different videos. Threshold = (Fmax+Tmax) /2, Tmax is the minimum value retrieved after the exponential adjustment and Fmax is the maximum value retrieved by adjusting the degree of similarity. In order, to detect the content leakage, the adjusted degree of similarity to the decision threshold specific to the original video is compared. Therefore, the accurate leakage detection is achieved.

The environment contains NV number of videos with each of its corresponding traffic patterns. The size of the traffic pattern L is defined by the number of slots. A set of traffic patterns with different sizes obtained from the original pattern size L is Ω. The matching process is done by comparing the size of the original pattern (L) and size of a specific pattern (l). Let M be the number of patterns of specific length existing in the environment retrieved from the original pattern. During the comparison between the number of pattern size of a specific length (M) and the original traffic pattern (L), the decision threshold with the size l is calculated. Matching process among the videos of specific length l is match(l). The previous scheme has been improved by considering the video size less than or equal to L for an effective comparison between the patterns of varying size. Consider a specific or particular video Vx, whose traffic pattern is compared with the set of patterns of specific size l retrieved from the original patter of any other video existing in the environment. Thus, this process is repeated for all videos present in the environment. In the proposed system, exponential approximation curve is estimated depending on the distributed pattern size and its degree of similarity. To find out the approximation curve, pattern size less than or equal to L is considered. Furthermore, the decision threshold for each video is determined based on the computed curve. The exponential approximation scheme (proposed system) requires less computation cost than enhancement of previous schemes.

The network performance, effectiveness, robustness and the accuracy of the dynamic decision threshold is evaluated. By implementing the dynamic decision threshold approach into the DP-TRAT performs high robustness against the network changes. DP-TRAT utilizes both the packet size-based traffic generation algorithm and the DP- matching algorithm.

Overview of the System Design

The user receives the streaming video contents from the server, where the traffic is being monitored at the user-side and the server-side. Packet observation points are employed for the traffic pattern generation at both the sender and receiver. NetEm-bridge connecting the server and the user handles variation such as packet delay, packet loss and jitter.

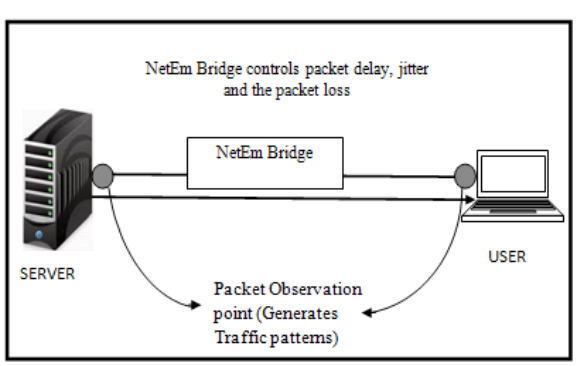

Figure 3: System Design

Varying video length performance

P-TRAT and DP-TRAT are the conventional methods that differentiate various videos of same length. Let us use the video portions of varying length from the set. A certain amount of videos is sent from the server to the user from the generated portion. The volume of traffic is monitored to generate traffic patterns for matching process. Videos of different length may cause performance degradation. Conventional methods (P-TRAT and DP-TRAT) perform comparison approach. When the number of videos in the environment is increased, the accuracy gets decreased. If the decision threshold with videos of different length is not set properly, then it leads to the flawed decision during the DP-TRAT detection performance.

Robustness

Here two kinds of experiment are done for the purpose of the estimating the robustness against the changes in the network environment. For example, consider 30 videos of varying length from 30 to 300 seconds in the environment.

During the first experiment, delay is generated at the NetEm-bridge, ranging from 0 to 200 milliseconds for every 25 milliseconds. Generated delay hasn't caused any problem since the traffic pattern generated by these methods has employed the packet size-based generation algorithm that avoids jitter and packet delay.

During the second experiment, packet loss is generated with NetEm ranging from 0.1% to 5%. The accuracy of the proposed method and the conventional methods are not affected by packet loss. Hence, P-TRAT is affected by the varying traffic amount due to packet loss and D-TRAT is also slightly affected, whereas the proposed system provides high detection performance without being affected by the packet loss.

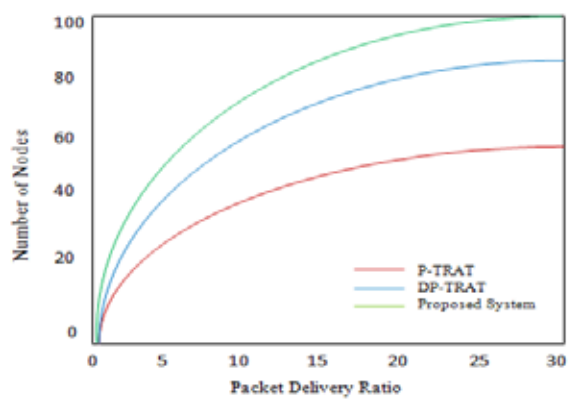

Figure 4: Performance of Packet Delivery

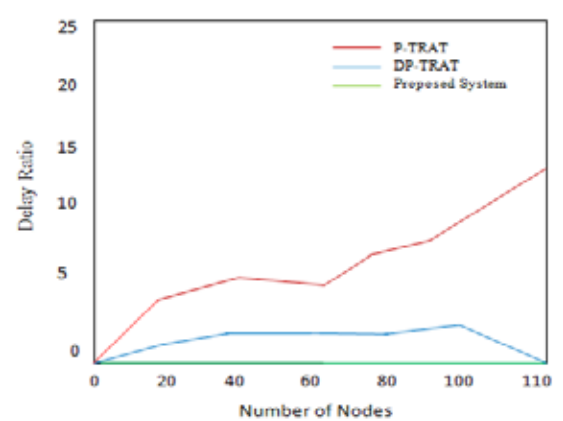

Figure 5: Occurrence of Packet Delay in the network

REFERENCE

[1]E.I. Lin, A.M. Eskicioglu, R.L. Lagendijk, and E.J. Delp, (2005) "Advances in digital video content protection," Proc. IEEE, 93(10), 171-183. | [2]S. Craver, N. Memon, B. L. Yeo, and M. M. Yeung, (1998) "Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks, and implications," IEEE J. Sel. Areas Commun., 16(4), 573-586. | [3]K. Su, D. Kundur, and D. Hatzinakos, (2005) "Statistical invisibility for collusion-resistant digital video watermarking," IEEE Trans. Multimedia, 7(1) , 43-51. | [4]Y. Liu, Y. Guo, and C. Liang, (2008) "A survey on peer-to-peer video streaming systems," Peer-to-Peer Networking and Applications, 1(1), 18-28. | [5]M. Dobashi, H. Nakayama, N. Kato, Y. Nemoto, and A. Jamalipour, (2006) "Traitor Tracing Technology of Streaming Contents Delivery using Traffic Pattern in Wired/Wireless Environments," in Proc. IEEE Global Telecommunications Conference, San Francisco, USA, 1-5. | [6]K. Matsuda, H. Nakayama, and N. Kato, (2010) "A Study on Streaming Video Detection using Dynamic Traffic Pattern," IEICE Transactions on Communications (Japanese Edition), J19-B, 2. | [7]S. Amarasing and M. Lertwatechakul, (2006)"The Study of Streaming Traffic behavior," KKU Engineering Journal, 33(5), 541-553. | [8]D. Geiger, A. Gupta, L. A. Costa, and J. Vlontzos, (1995) "Dynamic Programming for Detecting, Tracking, and Matching Deformable Contours," in Proc. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(3), 294-302. | [9]Ram M. Narayanan, (1994) "First Order Exponential Approximation for Small Arguments," IEEE Magazine on Aerospace and Electronic Systems, 9(2), 33-35.