In Proc. of 30th EUROMICRO Conference, Rennes, France, September 2004.

A Video-Spam Detection Approach for Unprotected Multimedia Flows based on Active Networks

Andreas Hess, Jirka Klaue
Telecommunication Networks Group, Technische Universität Berlin
Einsteinufer 25, 10587 Berlin, Germany
{hess, jklaue}@tkn.tu-berlin.de

Abstract

This paper presents a real-time protection mechanism for MPEG-4 flows against video spam. Applications like Internet telephony, video on demand or life video streaming become more and more important. Because of the increased appearance of e-mail spam during the last years, it is expected that these services will get into the focus of the spam industry.

Since it is necessary to cope with the individual QoS-requirements of media flows and the characteristics of varying transmission media, it might be advisable not to encrypt media flows such that intermediate and edge nodes could optimize the transmission of these flows, e.g., by protocol boosters. Moreover, currently used video-conferencing systems and real-time multimedia applications mostly lack integrated security features.

We have been able to demonstrate the blending of media streams with unwanted content and have implemented an active networking service that analyzes MPEG-4 media flows. Additionally, it is capable of detecting and blocking the above mentioned attack.

1. Introduction

Applications like Internet telephony, video on demand or life video streaming gain more and more in importance. For example, a video-conference system allows to discuss in real-time with colleagues of a remote branch, without having to travel. Though interactive applications pose strict requirements with regard to end-to-end latency (maximally 400 ms) and jitter, it is often not possible to integrate security services as data authentication, integrity and confidentiality.

In principle IPsec, which is an often mentioned solution, can under certain restrictions be employed to protect real-time multimedia streams. IPsec consists of a set of protocols which provide data authentication, data integrity and confidentiality to IP datagram units. But, for example, applying IPsec (tunneling mode and ESP) to protect a voice channel, using the common vocoder G.723.1, leads to an overhead up to 52.4% [7]. Furthermore, the Internet Key Exchange standard only specifies the negotiation of a Security Association (SA) between two parties, whereas a video conference might involve more participants. Beyond this, it is a huge challenge to combine IPsec with existing Internet QoS approaches, as the IPsec header does not reveal information about the type of data that is transported.

One approach that is capable to cope with the QoS issue is the Secure Real-Time Transport Protocol (SRTP), which currently is an action item of the IETF. SRTP is a security profile for RTP and RTCP, which provides confidentiality, message authentication, and replay protection to both protocols [7]. But even here two issues remain: first, it is not yet implemented in applications and second, it requires changes in the application and/or the end-user devices.

But with regards to a future fusion of many currently (still) separated networks respectively technologies, many devices of low computing power will become part of the Internet. Taken this into account, it might be necessary to transcode media streams according to the capabilities of the receiving end system and thus, the media flows must not be encrypted. With respect to mobile devices also the trade-off between computer power and power saving must be considered.

Another important aspect is transparency for users and applications. Widely spread and used applications like RealPlayer, Media Player or Netmeeting were developed without considering security issues and consequently, most of them use UDP/RTP for data transmission. Hence, a subsequent integration of security services mostly requires the modification of applications respectively end systems, but many users are not able to update their systems or do not like that.

In this paper we introduce a video spam detection approach for unprotected multimedia flows based on active networking. An active network allows to dynamically

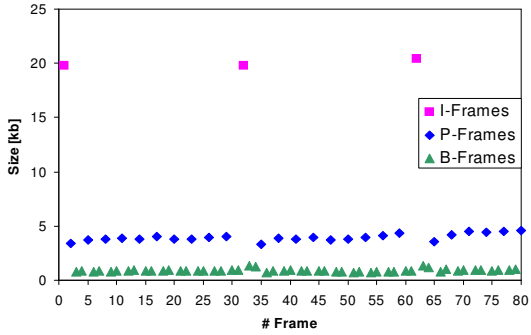
deploy task-specific services on chosen active nodes in order to specifically process a designated set of packets. We implemented an active service that creates a profile of a MPEG-4 video flow and compares the profile with current network traffic. The service triggers an alarm in the repeated event of an inconsistency between profile and network traffic. Furthermore, we show how an underlying active networking infrastructure could help to identify, perhaps involuntary, spam relays.

The paper is structured as follows: in Section 2 we shortly introduce MPEG-4 video streams, their structure and how we create the stream profiles in the network layer. Section 3 discusses active networks and their use for our purpose. Then our experiments and results are shown in Section 4 and finally we conclude our paper in Section 5.

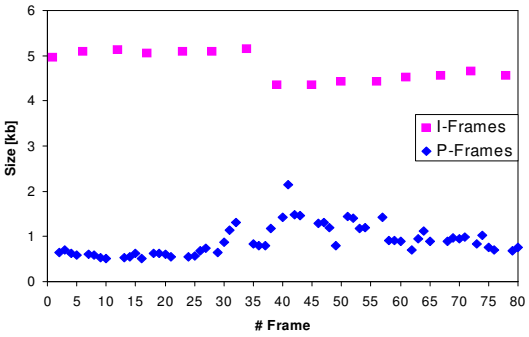
2. MPEG-4 Flow Detection

MPEG-4 video flows basically consist of different frames, namely I-, P- and B-frames [8]. Dependent on the video source and the encoder parameters, the size and succession of these frames vary. In Figure 1, the stream profile of two video sources with different encoder parameters is shown. The periodic sequence of frame types, e.g., *IBBPBBPBBPBB* is called Group of Pictures (GOP). Most real-time applications encode the video streams with fixed GOPs, at least fixed for a certain time span. In our spam detection approach we use this periodic GOP structure. If frames from a different source are interleaved with the stream, the decoder will accept and decode these frames, too. This leads to the effect of the mixed displaying of two sources. If the interleaving is done with I-frames only, the original video is almost completely domineered (Figure 2). It is easy to imagine that this could be exploited by spammers or script kiddies.

If a video stream is delivered via UDP/RTP, the GOP profiles can be extracted relatively easy by a classifier. Such classifiers are needed for Differentiated Services (DiffServ) and Quality of Service (QoS) frameworks [10, 2] anyway. They work by extracting content information from the RTP-headers or directly from the payload. We have implemented a classifier module which extracts frame type information from the UDP payload of the packets. It takes advantage of the fact that frames start at packet borders due to error concealment considerations [11]. Moreover, MPEG-4 frames can be identified by unique start codes [8]. The classifier extracts the profile by looking for start codes at the beginning of the UDP or RTP payload. The GOP structure is identified and memorized. We compare COPs to the saved stream profile whereby we are enabled to detect anomalies in the stream. At this point it must be taken into account that losses can occur, which could cause anomalies that have nothing to do with spam attacks. That's



(a) Profile 1



(b) Profile 2

Figure 1. Examples of MPEG-4 flow structure: (a) with B-frames (b) without B-frames

why occasionally occurring GOP failures are tolerated. However, frame losses are rare compared to anomalies caused by frame interleaving. If lots of GOP failures occur in a short time, the algorithm decides that it is due to an attack. Furthermore, the base GOP is reinitialized periodically to adapt to variably structured streams. Another issue is the possibility of packet reordering. This could lead to false GOP detections, but it can be solved by taking the RTP sequence numbers into account.

3. Active Networking

An active network is a network in which the nodes can be dynamically instructed to perform custom operations on a specific set of messages that pass through the node. In detail, an active node is able to execute services which are loadable on demand from a remote service module

(a) Original stream (b) Attackers stream

(c) Interleaved streams

Figure 2. Interleaved display of two sources (a) Original stream (Akiyo, video sequence, I-, P- and B-frames) (b) Attackers stream (Kimble, static image, I-frames only) (c) Interleaved display of source (a) and (b)

repository and thus enhance its functionality in a flexible manner. Application examples for this technology are media transcoding services (adaptation to the available bandwidth), protocol boosters [6], network security enhancing services [5] or overlay networks [3]. Active nodes are mainly located in proximity to the users due to a lower traffic volume and a higher distribution of the active services. A great benefit of operating active nodes is transparency. Users do not have to (re-)configure or upgrade their systems – the corresponding operations are autonomously done by the active network itself.

In the context of this project, active networks are used to analyze, block and backtrack video flows. This will be discussed in detail in the remainder of this section. But first of all, we give a short explanation, why we think that the integration of a backtracking mechanism into the system is required. The initial position is that most spammers try to hide their real identities by:

- IP source address spoofing or
- using (involuntary) spam relays.

Since more and more public and commercial channels for spreading spam are shut down, it is feared that spammers are turning their attention towards other possibilities to distribute their content. An example is the Sobig-F worm which amongst other things set up a spam relay server on each infected system. Consequently, in the future many spam relays might be compromised hosts.

For the purpose of rapidly reestablishing the original and desired video session, it is mandatory to block the disturbing traffic, on the active node which detected the pattern anomaly, if possible. Afterwards, the overlay network of active nodes is used to discover the point of origin of the spam traffic in order to block the unwanted traffic as early as possible and to identify (involuntary) spam relays. Hence, the system must be able to differentiate and handle the following three scenarios:

- two media flows with varying IP source addresses,
- two media flows with the same IP source address but with varying network paths or
- two media flows with the same IP source address and with (nearly) the same network paths

A fundamental requirement of our approach is that the video flow that is analyzed first is the requested one. Primarily, at the beginning of a video transmission the MPEG-4 profiler must be deployed on the active node which is closest to the receiver (see Figure 3). Second, the MPEG-4 profiler starts profiling the video flow as described in Section 2 and third, it memorizes the IP source address of the flow. Finally, the network path of the video flow is required. The active networking environment provides communication mechanisms among the active nodes which can be used for this task. On each active node a specific service that is able to check if the node forwards packets to a specific destination is running. Next, the router that is running the MPEG-4 profiler triggers the overlay network members in order to check which active routers lie on the network path of the video flow. The requesting node sorts the positive answers in descending TTL order

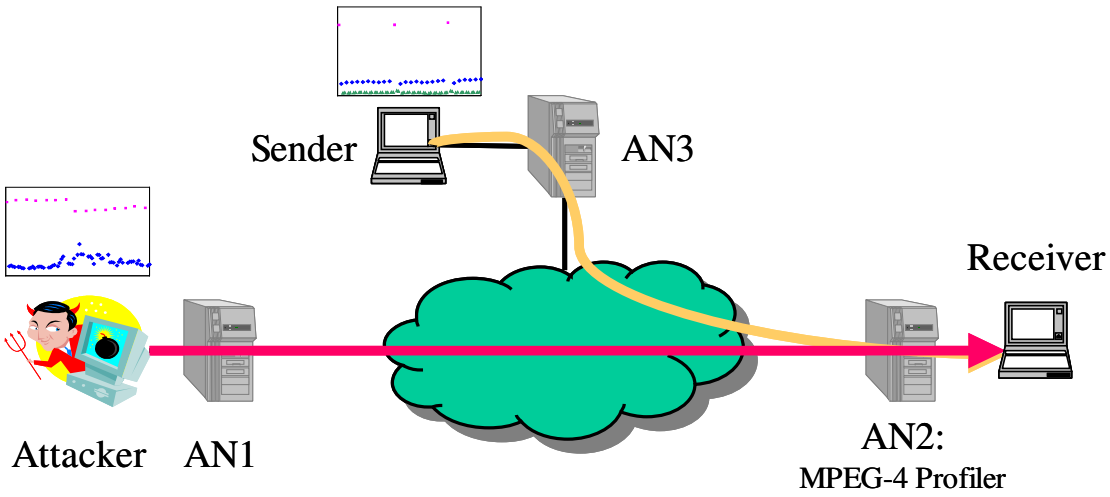


Figure 3. Setup of testbed

and memorizes those. The knowledge of the active node which is supervising the multimedia stream consists of the following components:

- video-profile,
- IP source address of video flow and
- network path of video flow.

Accordingly, in the event of a video spam attack the system performs the following set of operations. First, it checks whether the IP source addresses of both video streams differ and if so, it briefs the other active nodes of the overlay network to block traffic identified by the ”bad” IP source address and the target’s IP destination address.

In case that the IP source addresses of both video flows do not differ, the network path information is taken as the next decision criterion. Hence, the system asks the overlay network members who of them is in charge of routing traffic towards the target. Again the positive answers are sorted in descending TTL order and compared with the initial stored network path. Those active nodes which are not enlisted in the original network path are instructed to block the disturbing traffic. Consequently, a fundamental requirement is that at least one active node lies on the network path of the video spam flow but which is not simultaneously a member of the original video stream path. In case that neither the IP source addresses nor the network paths differ further criterions are required in order to decide which packets might pass an active router and which to block. For example, a possibility would be to analyze the size of the video frames but this belongs to the scope of future work. Another issue to keep in mind is to be aware of routing changes in the Internet. Currently, the original network path is analyzed and stored at the beginning of the transmission but routing changes due to an overloaded link or router can occur. A solution would be to periodically trigger the overlay network neighbors in order to evaluate the network path, but this is not integrated yet.

As a final remark, with each setting of a blocking rule on an active node a timer is also started. In the event that a timer elapses, the validity of the rule is verified, thus, that either the blocking rule is removed or the timer is reset.

4. Experiments and Results

In this section we shortly describe the experimental setup and the prototype which we used in order to prove our presented concept. Figure 3 schematically shows the configuration of the network we used for our tests.

The *Receiver* is a Windows ™-client running the video player. To demonstrate the feasibility of our video spam detection approach we used both the VideoLAN software [1] and a proprietary MPEG-4 player[1]. *Sender* and *Attacker* are Linux boxes executing our MPEG-4 software [9]. Additionally, we arranged three Linux boxes with the AMnet [4] framework and active network infrastructure near the clients. Together with our active network service the nodes *AN1*, *AN2* and *AN3* constitute the active network. Since this is a static scenario, dynamic service deployment facilities are not needed and, in our setup, we did not use them. *AN1* and *AN3* are also used to generate packet loss (equally distributed). In our experiments we used loss rates between 0 and 3%.

Different video sources came into operation, namely the well-known *Akiyo* sequence (Figure 2a) with low motion, a high-motion sequence (a clip from an action movie) and

1 This is an MPEG-4 player developed in the context of the ODYSSEUS project together with the HHI Berlin which is not publicly released.

a static sequence with no motion (Figure 2b). These video sources were MPEG-4 encoded using different parameters, namely constant/variable bit rate, different GOP structures *IBBPBBPBBPBB*, *IPPPPP* and *I*-frames only. We tested all possible combinations (except two videos with only I-frames) at different loss rates. Node *AN2* with the flow profiler detected and blocked all attacks correctly. Even if the same video was used as source and spam stream, the algorithm was able to detect the attack. This is due to the fact that an interleaved stream never has a stable GOP. Moreover, anomalies caused by UDP packet losses did not cause a false alarm.

If both the source video and the attacker's stream consist of only I-frames the GOP-based approach would not work. An idea how to deal with such scenarios is discussed in Section 5. Until now we have not faked the IP source address of the attacker. Consequently, we have not used the active backtracking mechanism in our setup yet. This is also an issue for further studies.

5. Conclusion and Future Work

We have demonstrated the possibility to insert video spam in real-time multimedia flows. The reasons why strong security is not always at hand for real-time applications were discussed and we proposed an alternative approach to detect these attacks which is based on an active network infrastructure. An experimental testbed was build whereby we showed the feasibility of our approach with many different stream parameters, taking also into account the problems caused by packet losses. Our detection module correctly identified spam attacks in all cases and never caused false alarms because of loss-generated anomalies. No changes in the existing protocol stack, applications and end user devices are necessary.

An open issue is how to deal with source streams with only I-frames. If the attackers stream also uses I-frames only, the GOP-based detection would not work. In this case, information of the traffic shape could be consulted essentially the frame size history (Figure 1). The blocking of unwanted traffic near the receiver is difficult if IP source addresses are spoofed. In this case the active network needs to backtrack the traffic streams to their respective sources in order to block them, as explained in Section 3. Another open issue is the investigation of possible exploits of RTCP and RTSP. For instance, it is imaginable that a running RTSP session could be stopped and then other content could be delivered. To assess the performance of our approach, we will compare it with IPSec and SRTP in terms of latency, jitter and packet loss.

References

[1] VideoLAN – free software and open source video streaming solution for every OS! http://www.videolan.org/.

[2] T. Ahmed and A. Mehaoua. Adaptive MPEG-4 streaming based on AVO classification and network congestion feedback. In *Packet Video*, Nantes, France, Apr. 2003.

[3] C. Bachmeir and P. Tabery. PIRST-ONs: a service architecture for embedding and leveraging active and programmable networks technology. In *IEEE 10th International Conference on Software, Telecommunications and Computer Networks, SoftCOM*, Oct. 2002.

[4] T. Fuhrmann, T. Harbaum, M. Schöller, and M. Zitterbart. AMnet 2.0: An improved architecture for programmable networks. In *Proceedings of the International Workshop on Active Networks IWAN2002*, Dec. 2002.

[5] A. Hess, M. Jung, and G. Schäfer. FIDRAN: A flexible intrusion detection and response framework for active networks. In *8th IEEE Symposium on Computers and Communications (ISCC'2003)*, Kemer,Antalya,Turkey, July 2003.

[6] C. Hoene, I. Carreras, and A. Wolisz. Voice over IP: Improving the quality over wireless LAN by adopting a booster mechanism - an experimental approach. In *Proc. of SPIE 2001 - Voice Over IP (VoIP) Technology*, pages 157–168, Denver, Colorado, USA, Aug. 2001.

[7] L. L. Iacono and C. Ruland. Vertrauliche Video-Konferenzen im Internet. In E. J. Jan von Knop, Wilhelm Haverkamp, editor, *Lecture Notes in Informatics: Security, E-Learning, E-Services*, German Informatics Society, pages 167–181, 2003.

[8] ISO-IEC/JTC1/SC29/WG11. *ISO/IEC 14496: Information technology - Coding of audio-visual objects*, 2001.

[9] J. Klaue, B. Rathke, and A. Wolisz. Evalvid - a framework for video transmission and quality evaluation. In *Proc. of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, Urbana, Illinois, USA, Sept. 2003.

[10] J. Shin, J.-W. Kim, and C.-C. J. Kuo. Content-based packet video forwarding mechanism in differentiated services networks. In *Packet Video Workshop*, Cagliari, Italy, May 2000.

[11] D. Wu, Y. T. Hou, W. Zhu, H.-J. Lee, T. Chiang, Y.-Q. Zhang, and H. J. Chao. On end-to-end architecture for transporting mpeg-4 video over the internet. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(6):923–941, Sept. 2000.