Authenticated Live Integration and Verification Scheme in Hybrid Content Delivery Networks

C. Chandrasekar, PhD
Assistant Professor, Department of Computer Science,
Arignar Anna Government Arts College, Cheyyar,
Tamil Nadu

Manuprasad V.
M.Phil. Scholar, Department of Computer Science,
Sree Narayanaguru Arts and Science College,
Coimbatore, Tamil Nadu

ABSTRACT
With the increase number of wireless communication environment, content distribution and streaming are became very ease. The Live video streaming over wireless network is very popular nowadays. Due to the popularity of streaming delivery of data contents over wireless networks, the development of live video streaming also has attracted much attention. This technology boosts the distribution of live video contents over CDN (Content Delivery Networks). Due to the tremendous popularity and reliability, the live video streaming suffers from huge security issues such as DOS (Denial of Services), clone attacks other malicious attacks. The proposed system protects the data from the above security issues by using a new prototype named as "ALIVE". It is a shield for data, which rescues from video crime and other misbehaving activities over CDN and P2P. Alive performs the source and destination authentication while transmitting the video, so this eliminates the entry of spoofer's and unauthenticated nodes in the network. Alive utilizes an encoding technique named as cryptmask, which performs an encoding technique against copyright forgery attacks. To achieve the above goal, alive incorporated with effective encoding techniques. The experiment has been performed using NS-2 tool.

Keywords
Content Delivery Networks, Video streaming, video source identification, wireless video streaming, video blocking.

1. INTRODUCTION
Wireless communication has seen a tremendous growth in modern era. With the booming of Internet applications, the multimedia streaming over the Internet is become very popular, and everyone can gain access to the media content on the Internet at anytime with reliable nature [1]. CDN is a group of servers or data centers, which located in several geographical areas to serve better content to end user faster. In CDN a client normally directed to download the video from the nearby server, in such cases P2P network has been used along with the CDN.

The figure 1.0 represents the process included in live video streaming, where live video streaming receives live video content as frames, and each frame will be encoded using streaming encoder and finally this will transmitted to the streaming server. Each video frame has been transmitted to the end user from the nearest server; the server selection is based on their geographical locations [2].

Though, presenting a video streaming services to significant end consumers is possibly one of the greatest displeased and challenges of the Internet. Video streaming on CDN is affected by certain issue such as scalability, quality heterogeneity and security [3]. Our study handles the security issue from the above. Such live video streaming leverages the upload bandwidth capacity of peers for the distribution of video/audio content. Distinct from traditional client-server based systems, CDN forward content to clients via several sub servers in the network.

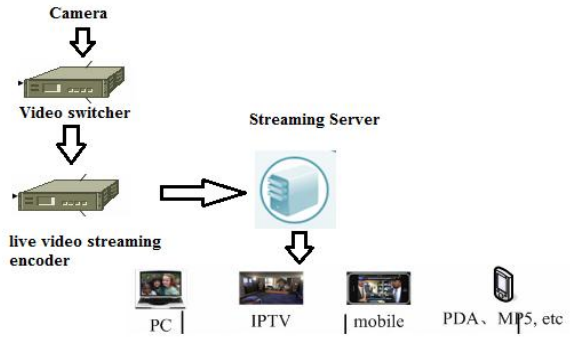

Fig1.0 Live video streaming process

Video Streaming Server (VSS) systems are designed to distribute the workload and network traffic among the servers and take advantage of the computing and storage resources of each individual server [4]. There are two aspects to this approach. The advantages of VSS system is very scalable and can` potentially serve a very large live video streaming community where the network and processing load will be a significant challenge for a centralized system. The distributed live video streaming architecture affected by various security problems and one potentially overwhelming threat is stream contamination and unauthenticated data access. In such attacks, the intruder interrupts and modifies the data stream while streaming [5]. In content distribution, the attacker corrupts the targeted content by adding noise data and performs copy right violations [5][9]. Sometimes the data is unable to differentiate the authenticated users and forged contents.

Security is important for the live VSS system to include mechanisms to authenticate content, consequently that the system is resistant to pollution attacks [6]. Such authentication can be implemented based on message digest or digital signature. In LVS (Live Video Streaming) the authentication can be done at two stages: source authentication and destination authentication, along with the above authentication process, the system also includes the integrity verification and data protection against various type of attacks over wireless content delivery networks.

2. RELATED WORK

2.1 Blacklisting Defense Mechanism

This blacklisting approach attempts to determine the peers that are present in the centralized or decentralized network that originate and relay pollution. Identified peers will be added into the blacklist. Network Peers cannot be able to send chunks or receive chunks from peers on the blacklist [10].

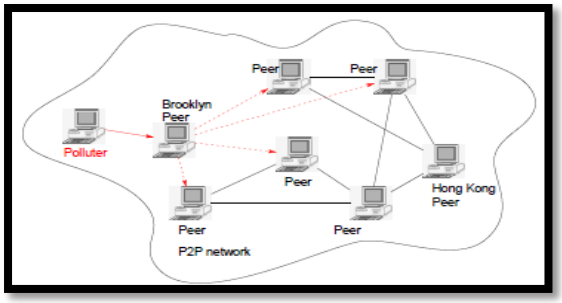

Fig 2.0 Peer to Peer Live data corruption

2.2 Traffic Encryption Mechanism

One of the main reasons that the current P2P video streaming systems are prone to pollution attack is that all of these systems have their messages transmitted in clear text. To make pollution the attacker needs to send the correct messages to the other peers with the correct header and data format [11]. One of the main disadvantages of using traffic encryption the system is subjected to a reverse engineering of the source.

2.3 Hash Verification Mechanism

A peer provides the hashes of all the chunks of the file. If a peer receives chunks from other peers, it compares the hashes of the chunks received with the corresponding hashes in the file to verify their integrity [12].

2.4 Chunk Signing Mechanism

There are many chunk Signing mechanisms have been used namely *Sign-All Approach, Signature Amortization and Sign-and-Correct [13]*.

This solution of different mechanisms incurs a delay corresponding to the processing of *n* chunks at the source and receiving and processing of *n* chunks in the best case and *n* in the worst case, at the receiver. The delay at the receiver here is slightly higher while using these signing mechanisms.

3. PROBLEM DEFINITION

There are several challenges associated with video content prevention, video authorization and video copyright forgery detection. The challenge in live video streaming is how to authorize the video distributor and restrict the degree of information that can be learned by the owner in case of video leaks. Differential security methods have been proposed for source detection and video content protection in hybrid content delivery networks. Several techniques have proposed with cryptographic primitives, even though they could concentrate only on data protection or source protection. Those techniques have been developed for perturbing the video streaming so as to preserve security while ensuring the mined frames or other analytical properties are sufficiently close to the patterns mined from original video.

4. ALIVE FRAMEWORK

To overcome the several security issues stated in previous chapter, we present a new technique named as ALIVE, which protects, authenticates and streams the data effectively. Several existing system concentrated either source or destination authentication. But our proposed technique concentrates on both source and destination authentication against data misuse attacks.

The contributions of this journal are as follows.

1) We enhance the existing live video source identification method along with destination authentication scheme which makes suitable for the videos against polluted by blocking, copy right forgery and blurring problems. This is the first work trying to address the source, destination identification authentication problem in the live video streaming.

2) We recognized several security issues in video streaming, wireless camera spoofing attack, which causes serious risk to security and surveillance systems.

3) We enhance and accelerate the video authentication and data forgery identification method by incorporating wireless channel characteristics. Our method is able to identify the source device much faster than the existing methods, and therefore can be used to detect the camera spoofing attack in a timely manner.

4) We perform the above authentication and forgery detection methods in hybrid content delivery networks such as CDN and P2P. For that we proposed a new scheme named as ALIVE.

5) We proposed a new invisible watermarking technique "CRYPTMASK", which helps to protect copyright forgery detection and unauthorized data detection.

Extensive simulated experiments are conducted to validate the effectiveness and efficiency of our method. The results show that our method largely outperforms the existing methods in the presence of video source ad destination authentication.

4.1 Network Model

The first model is initializing the simulation with network construction which has 50 mobile nodes. The system simulated the proposed scheme by using the ns-2 network simulator. In the simulation, 50 mobile nodes are placed within a square area of 1500 m × 1500 m. this use Random Mobility model to determine movements of mobile sensor nodes. In the Random mobility model, each node moves to a randomly chosen location with a randomly selected speed between a predefined minimum and maximum speed. After reaching that location, it stays there for a predefined pause time. It then randomly chooses another location after that pause time and moves to that location. This random movement process is repeated during a simulation time.

4.2 Authentication Key Generation against copyright forgery attacks

The system performs key generation scheme for secure video streaming. This helps to prevent the video content from attack. The second process of ALIVE creates an authentication key based on node analysis. In the key generation process, keys are generated dynamically using

video frame number and copyright details. The CRYPTMASK algorithm works on a public and private key system. The public key is made available to everyone. With this key a VSS can encrypt video content without the copyright key the user cannot decrypt it, the only person who can decrypt it is the one who possesses the private key and the copy right key, which can be received from their neighbor. In order to reduce the key transmission overhead and for security, the system grabs the keys from its neighbor. It is theoretically possible but extremely difficult to generate the private key from the public key; this makes the CRYPTMASK algorithm a very popular choice in signature generation.

Algorithm: CRYPTMASK

Input: video content

Output: Authentication for Masked data

Usage: solution against copyright forgery attack.

Steps:

1. Read the packet set D.
2. Begin the encode process E for every D.
    a. For each frame f in Vs do Df[]=encode(f)
3. Perform integrity verification for the transmitted packets.

In this section, this introduces the encoding scheme against source, destination and copyright forgery attacks, called CryptMask, which transforms a transaction file *D* into its encoded version *D.*The cryptmask technique is more unique and does not provide any details about the original data, and this effectively detects the modified content at the time of integration. This is a type of mask which helps to hide the data from un authorized users.

The above algorithm specifies the steps related to the CRYPTMASK. In the concept of forgery node finding and video streaming, the CRYPTMASK act as a proactive mechanism where the data has been masked from unauthorized users.

4.3 Authentication and data verification process

This step proposes a mechanism to detect any unauthenticated data access and video content attack, which results from dropping, delaying, modifying, spoofing or fabricating of video frames. This step allows a node to distinguish between its neighbors to prevent identity spoofing among them. This is used to build a video content structure of the first-hop neighbors of each node and the neighbors of each neighbor. The video authentication structure is used through ALIVE to detect malicious nodes and respective counter measures.

The message receiver should be able to verify whether a received message is sent by the node that is verified or by a node in a particular group. The adversaries cannot pretend to be an innocent node and inject fake messages into the network without being detected.

The proposed message authentication scheme performs a lightweight network code updating process. The main idea is that for each message m to be released, the message sender, or the sending node, generates a source message authenticator for the message m. The generation is based on CRYPTMASK. In order to provide effective authentication against unauthorized video content accessing each node in the network member is required to compute and add their signature for all other members who transmits the video content in the network. In the scheme, the entire process requires only three steps, which as follows. Hop by hop verification and video streaming.

This ALIVE verifies the collected video content with the respective receiver data and finds any deviation and forgery did. Finally the false video content will be filtered and malicious node will be blocked based on the authentication method.

5. EXPIREMENTS

This performed simulations using network simulator. The simulator is written in C and implements the Random Mobility Model. The events (nodes meeting, node arrival at its selected destination, and alarms time-out) are pushed to and pulled from an ideal time-line. Initially, nodes are assumed to be randomly deployed over a network area. Then, until the simulation ends, for each node, a random speed and destination location are randomly chosen (within the bounds set by the user): this implies to analyze and to order all the meeting events and the node arrival events with reference to the time-line. While the time goes by, the events on the time-line are processed. The events corresponding to node arrival are processed as previously described (choosing a destination, a node speed, and analyzing the new generated events). Along with 50 nodes 2 stream servers have created. The streaming events are processed as the main part of the ALIVE, where the lightweight integrity and source and destination authentication has been completed. Our proposed work is successfully implemented using Ns2. The performance of this proposed work ALIVE using cryptmask scheme is discussed in this chapter. The figure below shows the configuration of the simulation.

Table1.0 parameter list

Parameters	
Number of Nodes	50
Topography	1500 * 1500
Simulation time	300

6. RESULTS AND ANALYSIS

Here the performance of the routing protocol is evaluated using the NS-2(version 2.35) stimulator. Various network parameters of the proposed system is analyzed and compared with the network parameters of its predecessors. Some of the parameters analyzed are:

1. Authentication Delay
2. Throughput

To evaluate the performance of the proposed ALIVE and Cryptmask schemes, computational and execution time are considered. The results chapters prove the proposed system is outperformed than the existing techniques. This considered the authentication delay and data encoding delay for deployed data on the video streaming process. Encoding and authentication verification delay are specified below.

Table2.0 Evaluation table

Parameter	Value(Time in sec)
Authenticating delay	8.4
Encoding delay	3.6

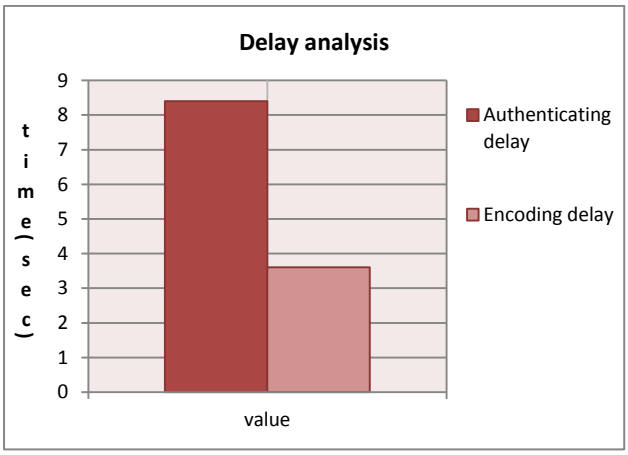

Fig 3.0: Delay Analysis Chart

The above figure shows the time delay for every process in our proposed scheme. It is defined as the average time taken by the packet to reach the server node from the client node.

Throughput

Throughput is the number of useful bits per unit of time forwarded by the network from a certain source address to a certain destination, excluding protocol overhead, and excluding retransmitted data packets.

$$\text{Throughput} = \frac{\text{Sent bytes}}{\text{Second}}$$

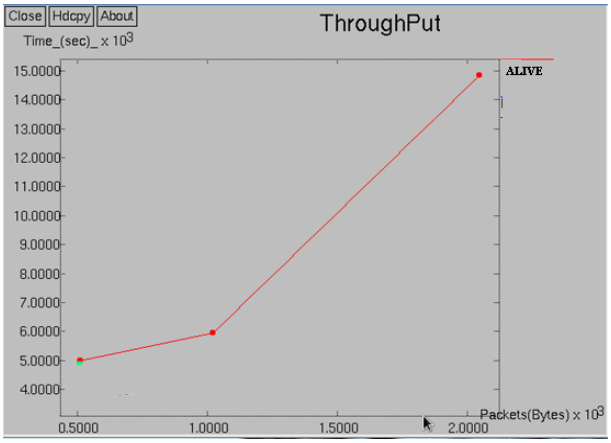

Fig 4.0 : Throughput analysis chart

7. CONCLUSION

In this manuscript we introduced an efficient method for source, destination authentications along with copyright forgery detection in scalable live video streaming over hybrid content delivery networks. We demonstrated the performance analysis even in the presence of video forgery and DOS attacks. We expanded the encoding technique for both data compression and copyright recognition, this also identifies the source and destination data using the cryptmask approach. Simulation and results shows the system outperforms in the term of time and throughput, this detects the attacks in live video streaming, so the throughput level has been increased.

8. REFERENCES

[1] Misra, Satyajayant, Martin Reisslein, and Guoliang Xue. "A survey of multimedia streaming in wireless sensor networks." *Communications Surveys & Tutorials, IEEE* 10.4 (2008): 18-39.

[2] Buyya, Rajkumar, Mukaddim Pathan, and Athena Vakali, eds. *Content delivery networks*. Vol. 9. Springer Science & Business Media, 2008.

[3] Pathan, Al-Mukaddim Khan, and Rajkumar Buyya. "A taxonomy and survey of content delivery networks." *Grid Computing and Distributed Systems Laboratory, University of Melbourne, Technical Report* (2007).

[4] Liu, Yong, Yang Guo, and Chao Liang. "A survey on peer-to-peer video streaming systems." *Peer-to-peer Networking and Applications* 1.1 (2008): 18-28.

[5] Dhungel, Prithula, et al. "The pollution attack in P2P live video streaming: measurement results and defenses." *Proceedings of the 2007 workshop on Peer-to-peer streaming and IP-TV*. ACM, 2007.

[6] J. Liang, N. Naoumov, and K. W. Ross. The index poisoning attack in P2P file-sharing systems. In *IEEE INFOCOM*, 2006.

[7] W. Conner, K. Nahrstedt, and I. Gupta. Preventing DoS attacks in peer-to-peer media streaming systems. In *MMCN*, 2006.

[8] C. Gkantsidis and P. Rodriguez. Cooperative security for network coding file distribution. In *IEEE INFOCOM*, 2006.

[9] M. Haridasan and R. V. Renesse. Defense against intrusion in a live streaming multicast system. In *P2P'06*, 2006.