International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-6, Issue-7)

Research Article

July 2017

Lightweight Authentication Scheme for Live Video Streaming Security

Resmi.A.M
Scholar: Department of Computer Science,
NGM college, (Autonomous),
Pollachi-642001, India

Dr.R.Manicka Chezian
Associate Professor:Dept. of Computer Science,
NGM College (Autonomous),
Pollachi-642001, India

Abstract—

The majority of the traffic in the internet is dedicated to video streaming and its applications. Content Network Networks are good enough in live video streaming. But, implementing live video streaming is not an easy task. Delivery of live video streams can be complicated because of the variety of platforms, network types and streaming formats competing in today's online media services. Every user's expecting reliable high quality performance with data availability using heterogeneous choices. And it is now tougher than ever to ensure an agreeable, problem less viewing experience on live streaming. Another problem of video streaming is handling security issues, which need fast verification and effective attack resistance. CDN suffers from various security issues, here, a new prototype to perform secure content delivery especially on live video streaming. A lightweight video streaming authentication technique is developed, this authenticates and transmits the data and eliminates the unauthorized access.

Keywords: Content Delivery Network, Video Streaming, Security, Privacy, Attacks.

I. INTRODUCTION

Content Delivery Networks are consisting with set of web servers, which helps to deliver the content based on the user's geographic locations. The main use of the CDN is offering content from the web servers by calculating the distance [1]. CDNs facilitate rapid page loads and offer other important benefits including the several advantages such as CDN eliminate Heavy Traffic and it is more suitable for multimedia video streaming. With the use of numerous concepts it minimizes packet Loss and user can get improved streaming quality and faster loading facilities. The proposed system concentrated on the video streaming process over content delivery networks.

Video streaming on CDN faces several problems, first, end-to-end delay is more important for live streaming than VoD (Video on Demand) streaming. In live streaming, the shorter the end-to-end delay is, the more lively the stream is perceived by the users. In on-demand video streaming, video stream is already prerecorded and that is not effective liveness [2]. It is simply irrelevant due to its prerecorded. Second, a user joining an on-going live streaming session is only interested in the stream starting from his/her joining time, while in the VoD streaming case the whole video must be delivered to the new user, and VoD streaming allows users to execute VCR-like commands over the video sharing networks.

A. Security Issues in Live Video Streaming Security

Securing live video streams is a complicated task; there are abundant techniques to sneak live streams. The live content streaming suffers by several attacks. Such as RTMP attacks, injection attacks, hacking stream name and performing clone and general DOS attacks [3].

RTMP attacks on live video streams: RTMP attacks on live video streams are usually performed by using rtmpdump programs, in general RTMP (Real-Time Messaging Protocol), which developed to stream audio, video and data on internet. Rtmpdump type of attacks is tough to detect, and protecting data from such attack is very challenging. This is due to several reasons, such as it can spoof any common security parameter, and it appears to be a legitimate stream user.

Injection attacks: Another clever attack in live video streaming is injection attack. The attacker runs a certain application, which their subscribers install. This will inject HTML into their site, so that the browser satisfies flash that their page is user's site, avoiding security. This type of attacks may like a real time user's access.

Video Forgery and disclaimer attacks: Forgery attacks break the condition of confidentiality and integrity of data streaming systems and can collude in pollution, membership, neighbor selection, Sybil and DoS attacks and intrusions [4]. The source of attacks is usually any peer node. In some cases super peers can do more damage to the system than average peers mentioned in the previous section as a requirement of CDN streaming systems.

Hacking stream names: Attackers determine "secure" origin stream names by reading network traffic. Various multimedia and video streaming servers leak the secret un-protected origin URL in the network traffic stream to hack the stream names and to perform the clone attacks. The origin stream name is kept confidential, and not published in the users resource, because it cannot run with relay servers if it requires security checking. Authors proposed several effective defenses against this attack. The client's streams never show on other sites, unless the client allows it.

B. Security goals in live video streaming:

While considering live video streaming in CDN, there are some specific properties that should be effectively handled to achieve secure streaming.

Video Content Authenticity and integrity: The data transmitted must be guaranteed and not tampered with and it must be guaranteed that it was released by the intended transmission unit. The non-disclaimer refers to the situation when the nodes that received a certain piece of data cannot deny that they received it. Non-disclaimer is of interest only for video on demand applications, while for Television like broadcasting it may be a negligible feature.

Confidentiality/ Privacy: The content that is transmitted during the streaming process can only be used or retransmitted to other nodes involved in the protocol. Achieving privacy is connected with access control process. In fact an access control system that prevents unauthorized participation to a streaming, but is not supported by a CDN that can prevent recording and later replication of the content becomes useless. Recent studies on commercial television streaming solutions have shown that they do not perform encryption, which makes the protocol lose not only confidentiality but also authenticity and integrity.

Anonymity: This is one of the most controversial properties, since in many contexts the capability of a user to remain anonymous is associated to potentially unlawful activities. However, specifically in television systems, the right of a user to watch a program without disclosing his identity is a key to privacy protection and should be guaranteed by broadcasting systems. This property should be assured by CDN streaming systems, not only in face of external observers, but also with respect to the other users of the same system, and the broadcaster as well.

Haridasan and van Renesse argue that not all applications need anonymity and confidentiality, but the features that matter most in frequent cases, are authenticity, integrity and non- negation [5]. Still anonymity becomes a key issue of privacy protection in live video streaming systems. Non- disclaimer, in the same systems, may be of secondary concern; unless a user can build claims on the fact that some information has not been delivered.

The motivation of the study is to provide the security for the video streaming and verification for the live videos in the CDN. In the previous studies the mechanisms used are concentrated on either one in the factors of receiver authentication or sender authentication. So they were failed to provide authentication against both source and destination at the same. Hear, a new live video security framework is proposed for content authentication and thwarting unauthorized user access. The secure communication eliminates many data corruption attacks and data misuse attacks by using the authentication protocol on live video streaming, it also provides high privacy and security for the data and content user.

II. PROBLEM DEFINITION

After compromising one or multiple nodes, an adversary may launch various attacks to disrupt the CDN. The attacks are commonly divided into two common problems, which are video forgery and data misuse. Many confronts associated with video content avoidance, video authorization and video copyright forgery finding.

There is an important task of secure video streaming is the way of authenticating both video distributor and content receiver. And it should restrict the users in case of video leaks. Variety of techniques and tools proposed with cryptographic primitives, but that concentrated on data protection else source protection. This is often preserves the security by ensuring the frames is sufficiently close to the prior mined frame patterns.

In CDN, different types of attacks may arise [6] and the possible countermeasures and detection strategies are mentioned in [4][7].

Intrinsic weakness of video encoding is that it is particularly vulnerable to DOS (Denial of Service) attacks and delay variance attack [8]. In CDN, malicious nodes can inject corrupted video packets into a network at the time of live streaming, which get combined and forwarded with usual frames, thus causing a large number of forged packets propagating in the network.

Proposed Scheme implements the novel crypto primitive scheme along with integrity, source and destination authentication for effective attacks detection and successful secure video streaming. This allows a node to verify, if the received video packets belong to specific rule criteria's and this construct the dynamic key index in every proxies.

III. PROPOSED SYSTEM

CDN achieves tremendous growth in modern era with effective content delivery mechanisms. With the booming of those CDN, the multimedia streaming over the network is become very popular, and everyone can gain access to the media content remotely at anytime with reliable nature. These security issues are tough to overcome due to its reliable and adaptive nature. So providing authentication and integrity check is very important. The scope of the work is to give guarantee for secure authenticated data transmission between networks with the prototype. The render and receiver hops identities and verifies the streamed data at the time of integration to provide a secure communication and to eliminate the network attacks.

A lightweight live video streaming security protocol which considers distributor, receiver side authentication technique named as LVSA (Light weight Video Streaming Authentication). LVSA can authenticate video streaming servers and video streaming receivers, which efficiently identifies falsely streamed videos. With the use of cryptographic primitives, the system improves the LVSA method, making it suitable for detecting wirelessly streamed videos with blocking and blurring along with the source, destination and content authentication.

This process needs a successful identification by applying an encoded authenticated key. Live video streaming affects the streaming process in the CDN_P2P sensor in two ways. Initially within the network i.e. source forgery and another one is in authenticated users.

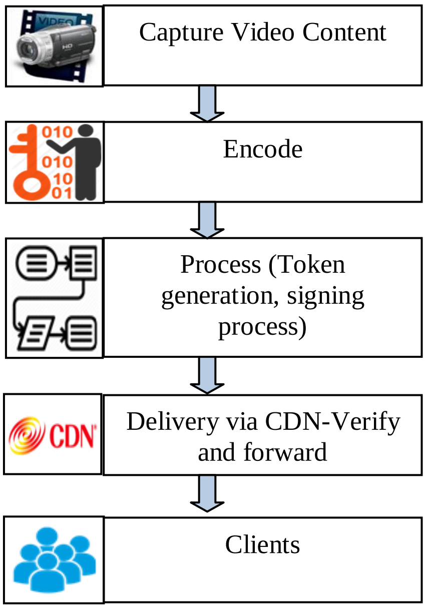
Capture Video Content
Encode
Process (Token generation, signing process)
Delivery via CDN-Verify and forward
Clients

Figure 1.0 LVSA framework Architecture

In LVSA framework, earlier a peer begins to download a video file; it obtains a streaming data which provides the hashes of all the chunks of the file. When a node receives video frames from other nodes, it compares the security/authentication key of the frames received with the corresponding hashes in the torrent file to verify their integrity and hop authentication. This considers applying the same technique for CDN live video streaming. The LVSA approach with cryptographic for this would be for each receiver to get the invisible watermarked lightweight key of each frame form the source itself. This would allow each hop to verify the integrity of each frame before forwarding it to other nodes. It will check the source and destination before delivering the streamed data. But, the load on the verification process on each hop will be very high for a large number of receivers, the proposed LVSA with cryptographic overcomes the data misuse problem by lightweight mechanism. The previous work allows an attacker to easily replace an original chunk from the source with a fake chunk and replace the corresponding valid hash with a hash for the fake chunk. When an unsuspecting node receives the duplicate/fake frames it verifies the chunk with the hash.

A. LVSA Process:

1. LVSA Initialization and Token Generation

a. Token Generation and Distribution

Initially the LVSA framework system collects the videos and remote user's details and encrypts the video contents using a token which has created in the initialization process. This process Encrypts and stores the video and transmits the data to the authorized remote client with receiver verification process. At the receiver side, while receiving the video content, our system verifies the data content and spoofed activity if any using the invisible watermarking technique.

2. Signing and Verifying Data Packets

a. *Signing Procedure*-After encryption the system performs the signing process, which helps to eliminate the forgery in the video streaming.

b. *Verification Procedure*

3. Token Refreshment

a. DATA ACCESS:

The decryption process is similar to that in the basic scheme. When the data attributes satisfy the user's access structure S, the user obtains. Then, the user decrypts the message. In this advanced scheme, T is able to update the master secret key y embedded in the user secret key SK by broadcasting to the users, where is the incremental of.

b. USERREVOCATION:

Fundamental functionality of CDN s is user management. In particular, the network operator should be able to revoke the user's access privilege when necessary. In the LVSA scheme, the LVSA framework follows cryptographic approaches to revoke users from the network. The approach is to define some time attributes and embed an expiration date to each user's access structure based on the time attributes. Users can then associate a time stamp to each streamed data using the time attributes. If users always associate the current time stamp to cipher texts, users will be automatically revoked after their designated expiration dates.

4. Content Decode and content delivery

To achieving secure data access control in hybrid content management network specify individual users (and hence the data collected by them) through a set of predefined attributes each user may be responsible for collecting specific types of data. This process collects the keys form its neighbor for data decode process. To specify data access privileges of users based on these attributes.

Effective source and receiver authentication using neighbor details:

Stage 1: **Initialization**

***Stage** 2: Claiming Neighbor's information:* Upon receiving an action message, a node verifies if the message is greater than last nonce and if the message signature is valid.

Stage *3:* **Processing claiming messages:** A claiming message will be forwarded to its destination node via intermediate nodes. Only those nodes in the network layer need to process a message, whereas other nodes along the path simply route the message to temporary targets.

Stage 4*:* **Data synchronization:** Based on the clock settings at the server, the nodes send their location and time stamps. This helps to identify the Sybil and fake nodes in the network. Time synchronization is needed by almost all spoofer detection Nevertheless; it is still a challenging task to synchronize the time of nodes in the network, even though loose time synchronization is sufficient for the detection purpose. That time synchronization process currently needs to be performed periodically to synchronize the time of each node in the network.

Stage 5: Key Hash Table for key verification: This maps streamed data items, key value pairs, on node IDs. Key is computed and stored in the KHT, which a fully decentralized, key-based neighbor caching and checking system is constructed to catch cloned nodes. The protocol's performance on memory consumption and a critical security metric are theoretically deducted through a cryptographic model, and the resulting equations, with necessary adjustment for real application, are supported by the simulations. In accordance with the analysis, the comprehensive simulation results show that the KHT- can detect node clone with high security level and holds strong resistance against receiver forgery attacks.

Algorithm:

Notations: Ts-time stamp, M-data content

1. Generates a prime number (p) and a number (g) which is between 1 and (p-1):
   - ex(p= 2467,g= 1731)
2. **selects a random number(x) as private key (x=149)**
3. Y=G^{x}modP (y= 1222)
4. send ([P,G,Y] to every nodes)
5. send key (91)
6. Attach Ts.
7. selects a random value (k), and calculates two new values (a and b):
   - where k is a random number (k= 80)
   - a=GkmodP (a= 1763)
   - b=ykMmodP (b= 367)

2. verification
   - M(verfiy)=b/(a^{x})modP (match key-91)

The algorithm represents the overall key generation and verification of the proposed work. LVSA performs key generation, management and key verification process on the video data.

Key Hash Tree Management

The first phase creates the time based key management based on session concept. In LVSA, keys are generated dynamically using local time. This is addressed in the Key Hash Management phase. When a Streaming server has data to send to the receiver, it uses its local clock value as the key.

Cryptographic Phase

The **cryptographic** phase addresses the security part of LVSA. The **cryptographic** phase obtains the dynamic key from the KHT phase and performs the necessary security service. This is also the phase where the key from the KHT is verified. If the key value received from the KHT phase is not correct then a new key is obtained from the KHT phase. This process continues until the correct key is found or the packet is marked as malicious to be discarded in the filtering-forwarding-synch phase when all attempts to find the correct key are exhausted within the tick window.

5. Authenticated video streaming

Finally this phase performs the filtering process based on the phase 2 and 1. The system eliminates the data if the node failed to produce proper key. The AVS phase filters the incoming packet out of the network if it is classified as a bad packet or forgery packet by the cryptographic phase or otherwise forwards it to the upstream nodes. In LVSA, this phase is also responsible for the synchronization process of the forwarder node with the Streaming server along a data delivery path toward the receiver with the invisible watermark modes of operation. At this phase, the forwarder node gets the source's local clock value from the cryptographic phase and updates its local clock value accordingly.

IV. EXPERIMENTAL RESULTS

A. Experiement:

The live video streaming security process is experimented using network simulator. The table 1.0 shows the parameter used for the experiment.

Table 1.0 experiment parameters

Parameter	Value
Simulation tool	NS2
Version	NS 2.35
Antenna	Omni antenna
Channel	Wireless
Number of nodes	40
Communication agent	TCP
MAC type	802-11
LL	Link layer type

The first model is initializing the simulation with network construction which has 50 mobile nodes. The system simulated the LVSA 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.

B. Results:

The developed video streaming process with security process is experimented and analyzed. Video streaming with security considerations are always need a high resource and time. But in the proposed system, using the lightweight encoding scheme for data and streaming security.

Table 2.0 RAM utilization performance analysis table

Node Count	RAM (Gbytes)
5	0.1
10	0.3
15	0.4
20	0.7
25	0.7
30	1.2
35	1.4
40	1.9
45	2.2
50	2.9

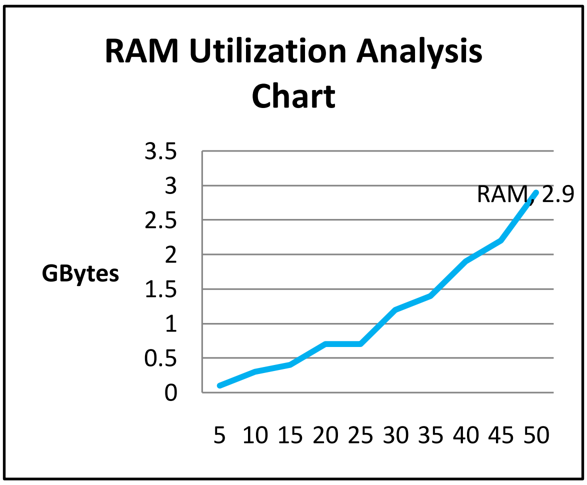

Figure 1.0 RAM utilization analysis chart.

The performance analysis showed that proposed system is highly effective in RAM utilization

Table 2.0 RAM utilization performance analysis table

Time (in 10s)	CPU usage (%)
0	0
1	2
2	3
3	6
4	8
5	11
6	13
7	15
8	16
9	17
10	18.1

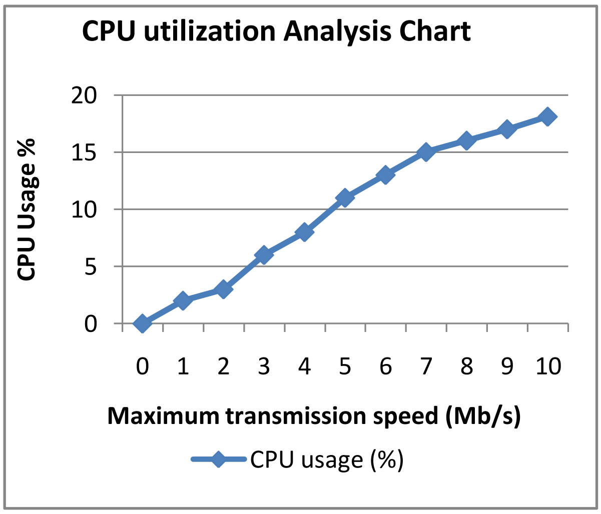

Figure 2.0 CPU utilization analysis.

As per the overall analysis, LVSA system takes 14% CPU when the data size 10 MB per second. This shows the CPU acceleration is increased when the data sie increases step by step.

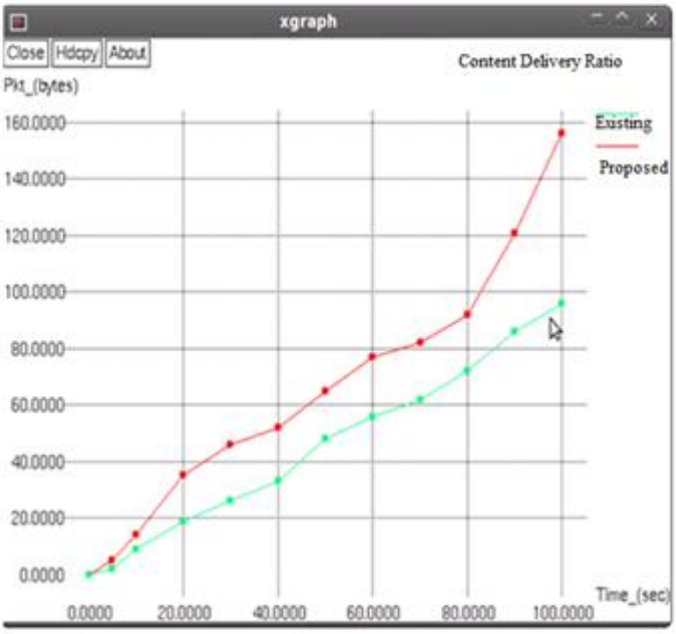

Figure3.0 Content Delivery Ratio Comparison Chart

To have a comprehensive evaluation of the proposed LVSA prototype with existing traffic encryption mechanism in different data size. For simulation, 50 numbers of nodes are used, which has considered as end users and the single distributor and multiple receivers. LVSA system performed continuous lightweight video streaming authentication process which consist of source, intermediate and destination authentication. This reduces the un-authorized data access and improves the content delivery. Figure 3.0 shows the content delivery ratio comparison between existing traffic encryption and proposed LVSA cryptographic process.

V. CONCLUSION

In this paper, an efficient method for source, destination authentications along with copyright forgery detection in scalable live video streaming over content delivery networks is developed. The results and experiments demonstrated the performance analysis even in the presence of video forgery and DOS attacks with various parameters. The expanded the crypto primitive and video encode technique for data compression and copyright recognition is designed, this also identifies the source and destination data using the LVSA approach. Simulation and results shows the system outperforms in the term of time and memory and CPU utilization, this detects the attacks in live video streaming, so the content delivery ratio level has been increased. The proposed system can be extended with real CDN implementation. This can be improved the performance and detection accuracy along with appropriate countermeasures. In the proposed system the security measures are concentrated to achieve streamed data integrity after authentication in the network communication. Whenever the security protection systems are involved in the system then, the performance must be measured to make sure that the system is giving the best performance speed in the real time scenario.

REFERENCES

[1] Newton, Christopher, Laurence R. Lipstone, William Crowder, Jeffrey G. Koller, David Fullagar, and Maksim Yevmenkin. "Content delivery network." U.S. Patent 9,456,053, issued September 27, 2016.

[2] Stoica, Ion, Hui Zhang, and Aditya R. Ganjam. "Managing synchronized data requests in a content delivery network." U.S. Patent No. 9,264,780. 16 Feb. 2016.

[3] https://www.scaleengine.com/streamsecurity.

[4] Resmi, A. M., and R. Manicka Chezian. "An extension of intrusion prevention, detection and response system for secure content delivery networks." *Advances in Computer Applications (ICACA), IEEE International Conference on*. IEEE, 2016.

[5] Haridasan, Maya, and Robbert van Renesse. "Defense against intrusion in a live streaming multicast system." *Peer-to-Peer Computing, 2006. P2P 2006. Sixth IEEE International Conference on*. IEEE, 2006.

[6] Chen, Jianjun, et al. "Forwarding-Loop Attacks in Content Delivery Networks." *Proceedings of the 23st Annual Network and Distributed System Security Symposium (NDSS'16)*. 2016.

[7] Resmi, A. M., and R. Manicka Chezian. "Network Security Architecture with Active and Dynamic Response Selection" International Journal of Advanced Research in Computer and Communication Engineering, 2017.

[8] Kumar, Sunil. "Implementation of delay variance attack using video streaming in MANET." *Optik-International Journal for Light and Electron Optics* 127.6 (2016): 3303-3307.

ABOUT AUTHORS

Resmi.A.M receivedMCA from Madurai Kamaraj University, Madurai. She completed her M.Phil Degree from Bharathiar University, Coimbatore. Currently she is doing Ph.D in Computer Science at NGM College, Pollachi. India. She has 10 years of Teaching Experience. She has published 5 papers national level/international conference and journals. Her research interest includes in the areas of Advanced Computer Network, Data Mining and Image Processing.

Dr. R. Manickachezian received his M.Sc., degree in Applied Science from P.S.G College of Technology, Coimbatore, India in 1987. He completed his M.S. degree in Software Systems from Birla Institute of Technology and Science, Pilani, Rajasthan, India and Ph.D degree in Computer Science from School of Computer Science and Engineering, Bharathiar University, Coimbatore, India. He served as a Faculty of Maths and Computer Applications at P.S.G College of Technology, Coimbatore from 1987 to 1989. Presently, he has been working as an Associate Professor of Computer Science in N G M College (Autonomous), Pollachi under Bharathiar University, Coimbatore, India since 1989. He has published one-fifty papers in international/national journal and conferences: He is a recipient of many awards like Desha Mithra Award and Best Paper Award. Recently he received the award "Best Computer Science Faculty of the Year 2015" from Association of Scientists, Developers and Faculties. His research focuses on Network Databases, Data Mining, Distributed Computing, Data Compression, Mobile Computing, Real Time Systems and Bio-Informatics.