Efficient Key Management Protocol for Secure RTMP Video Streaming toward Trusted Quantum Network

Montida Pattaranantakul, Kittichai Sanguannam, Paramin Sangwongngam, and Chalee Vorakulpipat

This paper presents an achievable secure videoconferencing system based on quantum key encryption in which key management can be directly applied and embedded in a server/client videoconferencing model using, for example, OpenMeeting. A secure key management methodology is proposed to ensure both a trusted quantum network and a secure videoconferencing system. The proposed methodology presents architecture on how to share secret keys between key management servers and distant parties in a secure domain without transmitting any secrets over insecure channels. The advantages of the proposed secure key management methodology overcome the limitations of quantum point-to-point key sharing by simultaneously distributing keys to multiple users; thus, it makes quantum cryptography a more practical and secure solution. The time required for the encryption and decryption may cause a few seconds delay in video transmission, but this proposed method protects against adversary attacks.

Keywords: Quantum cryptography, key management, custom protocols, symmetric encryption, real-time message protocol, RTMP.

Manuscript received July 23, 2014; revised Mar. 24, 2015; accepted Apr. 15, 2015.

Montida Pattaranantakul (corresponding author, montida.pattaranantakul@nectec.or.th) and Chalee Vorakulpipat (chalee.vorakulpipat@nectec.or.th) are with the Wireless Information Security and Eco-Electronics Research Unit (WISRU), National Electronics and Computer Technology Center, Pathumthani, Thailand.

Kittichai Sanguannam (kittichai.sa@gmail.com) is with the Department of Information Technology, Triple T Broadband Public Company Limited, Nonthaburi, Thailand.

Paramin Sangwongngam (paramin.sangwongngam@nectec.or.th) is with the Intelligent Devices and System Research Unit (IDSRU), National Electronics and Computer Technology Center, Pathumthani, Thailand.

I. Introduction

The role of videoconferencing over the Internet has grown significantly to meet human requirements. Videoconferencing solutions are an alternative option to fulfill this growing demand — according to a next-generation videoconferencing white paper, 20 million workers globally will run corporate-supplied videoconferencing from their desktops by 2015, and the market for videoconferencing will reach $8.6 billion [1].

However, general data communication services such as videoconferencing have many inherent vulnerabilities and associated security risks. Attackers can detect or capture video streams when video conferences are in session by monitoring data transmission patterns and further analysis in terms of protocols or a set of rules on how those users communicate messages to other people. This is a critical issue, whose resolution can probably prevent data loss and corruption.

As a result, many research efforts have been aimed at designing a videoconferencing system based on security architecture. In general, two basic security mechanisms exist that focus on preserving confidentiality of video data — specific video encryption algorithms [2]–[4] and virtual private networks (VPNs) [5]; there are a number of VPN protocols being used for videoconferencing such as IPsec VPN [6]–[8] and SSL VPN [9]. However, there are some drawbacks and negative aspects in using either VPN mechanisms to create an encrypted tunnel or applying video encryption algorithms to scramble video contents.

Regarding the aforementioned security issues (drawbacks), most video encryption algorithms are secured based on

pseudorandom numbers that are created from mathematical functions and provide outputs as periodic sequences and patterns. This could be a security drawback if keys were trapped or the patterns broken up. The second problem arises in key management and security associations. Although symmetric key cryptography is the most popular encryption algorithm used to create a VPN tunnel, it would be difficult to share symmetric keys between two remote devices securely. As a result, the distribution and management of keys seems to be a critical problem that remains an open research area and requires further study.

To resolve the above problems, ensure end-to-end protection, and enhance security architecture for a videoconferencing system, additional support might be required, such as an authentication mechanism [10], a decentralized group key management [11], cryptographic algorithms [12]–[13], and a level of trust [14]–[15]. Therefore, the main contribution of this paper is a proposed new framework on how to transmit video contents over a public network in a secure domain. This framework comprises three different layers, the lowest of which is the quantum key distribution (QKD) layer, which provides a mechanism for securing key exchange between two parties based on the laws of quantum physics [16]. These secret keys will be used to encrypt video streams. The next layer up is the key management layer, which encompasses all activities related to the keys, such as storage, distribution, and destruction. The main function of the key management layer will be inter-operated with the QKD layer by accumulating the quantum secret keys, creating a secure channel, exchanging key information, and distributing these keys to be used for further video encryption simultaneously; thus, the proprietary key management protocols [17] have been proposed regarding this layer. The last layer is the application layer, where the videoconferencing system comes into play. The media data of audio and video streams is encrypted based on symmetric key encryption by using quantum secret keys as a part of the encryption process.

The rest of this paper is organized as follows. Section II presents the background research of QKD. Section III discusses the conceptual framework of trusted quantum networks in more detail. Our proposed key management framework is described in Section IV, while Section V presents a secure videoconferencing system based on the proposed key management protocols by using quantum key encryption to ensure robust and reliable video transmission. Section VI provides a comparative analysis between the existing structures and the proposed framework, and shows the experimental results of data encryption performance. Finally, Section VII features some concluding remarks and future works.

II. Background of QKD

Traditional computer processing is based on a foundation of binary digits represented as a set of bit string values in such a way that each bit must be either zero or one, while the occurrence of the two values simultaneously is not feasible. In fact, techniques for the creation of bit strings have been derived from mathematical equations that provide periodic patterns of key values. The drawbacks of generating keys based on mathematical equations may break down the ideal of information security; therefore, the final result can be exploited by unanticipated advances in algorithms and hardware when quantum computers [18] will become a reality.

To figure out the traditional weakness of pseudorandom number generation, the concept of QKD, usually known as quantum cryptography [16], [19]–[21] has been proposed. This technology offers a promising, unbreakable way to steal keys, as well as providing secure communications over an untrusted network in such a way that if any eavesdroppers attempt to intercept secret keys during a quantum key exchange state, then detectable changes in the system will occur through the introduction of abnormal high bit error rates of a key. This appealing characteristic has been applied to create a strong key, stopped eavesdroppers, and increased security performances in videoconferencing systems.

III. Conceptual Framework of Trusted Quantum Networks

In general, a trusted quantum network can be divided into three different layers — a QKD layer, a key management layer, and an application layer. Our approach to a practical architecture design for a trusted quantum network consists of two quantum links and three physical nodes serving as links between three different locations that are structured as a star topology (see Fig. 1).

1. QKD Layer

In the QKD layer, a number of pre-configured and pre-installed QKD devices perform quantum key generation. Each such device is linked together with its peer through a quantum channel to establish shared quantum secret keys. These quantum secret keys can only be shared between key management servers that have a directed quantum link in common.

2. Key Management Layer

After the quantum secret keys (a quantum secret key from here on in is referred to as simply a key) have been generated

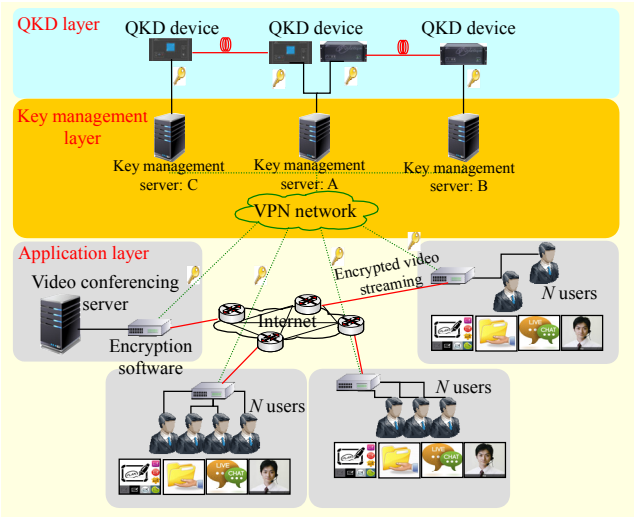

Fig. 1. Proposed framework of secure videoconferencing system based on trusted quantum network.

by the QKD layer, a pool of ordered secret bits will be forwarded to the key management layer, where key management servers are installed. The tasks associated with these key management servers play an instrumental role in the formation of a secure channel among the servers.

In our approach, a VPN connection is applied as an additional mechanism to improve system security during key management exchange processes. Thus, only information pertaining to keys, such as *key_id* and *pairing_node*, are transmitted from a key management server to a peer server through a VPN channel to check the correctness of the keys and whether paired users do in fact obtain the same keys, while the actual keys are kept secret in a key management server's storage system. Consequently, the key management servers aim to provide secure key storage, transfer keys among key management servers, perform routing, and later distribute keys to be used in video encryption. Using a key-transfer technique based on a hop-by-hop key encryption paradigm allows a couple of users who are making requests and whom do not connect or belong to the same quantum link to share the same keys.

3. Application Layer

The application layer is where videoconferencing systems and transparent encryption software reside. When a secure video conference session starts, all of its related video streams will be encrypted with keys before being sent out through the Internet; the encrypted video streams will be decrypted after being received from the Internet. Therefore, various cryptographic services require a number of keys to provide secure communication.

IV. Architecture of Secure Key Management

In large-scale communication systems, secure and efficient key management schemes require complex replication and scaling architecture, both of which are difficult to implement. Although a number of key management techniques have been submitted to the scrutiny of experts and follow industry standards such as ISO [22], ANSI [23], and NIST [24]–[26], many key management applications that contain their own unique proprietary protocols with the aim of avoiding issues relating to incompatibility have been proposed.

Presently, cryptographic key management has been utilized in many practical applications, such as quantum communication networks. For instance, the DARPA quantum network [27]–[28] relies on the IPsec protocol suite and universal hash function. The SECOQC project [29]–[30] proposed a customized architecture and protocol stack for a QKD network. The idea was inspired by an Internet model consisting of both the Quantum Point-to-Point Protocol (Q3P) [31] and the QKDTL protocol [32]. In the meantime, the SwissQuantum project [33] has been designed and deployed to demonstrate the reliability and robustness of QKD in modern enterprise network scenarios. Recently, the Tokyo QKD network [34] was built to demonstrate eavesdropping attacks over secure video transmission. The network consists of key management agents and the key management servers.

1. Structure Overview

The key management and custom protocols of this paper are state of the art. This paper proposes a simple key management infrastructure for synchronizing and managing keys among key management servers and onward distribution of symmetric keys to corresponding applications. This paper focuses on how to utilize key management services to improve user satisfaction with efficient and secure support when using a videoconferencing system. Thus, key management tasks are related to managing and distributing quantum keys in parallel based on user requests. Our proposed key management protocol has been successfully implemented and demonstrated in a real-world test by adapting keys for further video encryption. Figure 2 illustrates an overview of our key management framework.

2. Key Management Framework

Our key management framework can be divided into five different protocols — key caching protocol, key transfer protocol, point-to-point encrypted transfer protocol (PPETP), key routing protocol, and key distribution protocol. These protocols perform different operations to provide a framework

of authentication and key exchange services. Moreover, the design of our key management method aims to support a compatible module in which the different protocols are able to work together with one another to achieve key management services. The details of each protocol are described in what follows.

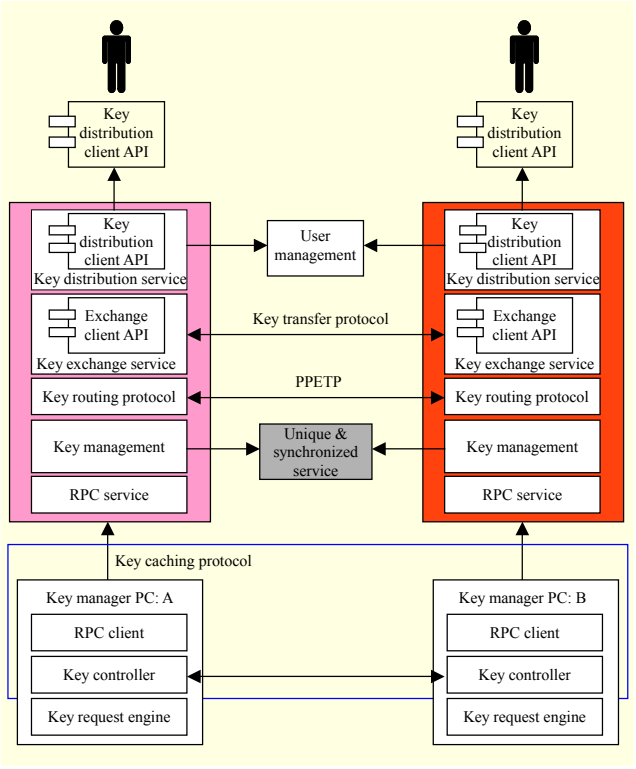

Fig. 2. Interoperability among different custom protocols over key management layer [16].

A. Key Caching Protocol

A key caching protocol, executed from a local site, is tasked with examining the correctness of the secret keys generated from the QKD devices by verifying their content. The main function of the key caching protocol of this paper is to ensure that, for all directed quantum point-to-point links, the secret-key bits appear to have the same value as those bits of their peers.

With reference to Fig. 1, there are link connections from site A to site B, and site A to site C. Thus, key management server A must contain keys that are identical to those held in key management server B; and vice versa.

In addition, this process has added both a transaction identifier number and a timeout value to each key block; the aforementioned key blocks will be temporarily locked during the key caching process to prevent illegitimate access by other operations. The sequential steps of our key caching protocol are illustrated in Fig. 3. Upon completion of a round of key verification, all of the verified keys are transferred to their corresponding key management server.

B. Key Transfer Protocol

A key transfer protocol is employed to transfer secret keys from one key management server to another. If end nodes do not share the same keys or belong to the same directed edges along the multilink connection–based quantum channels that exist within the network structure then the key transfer protocol

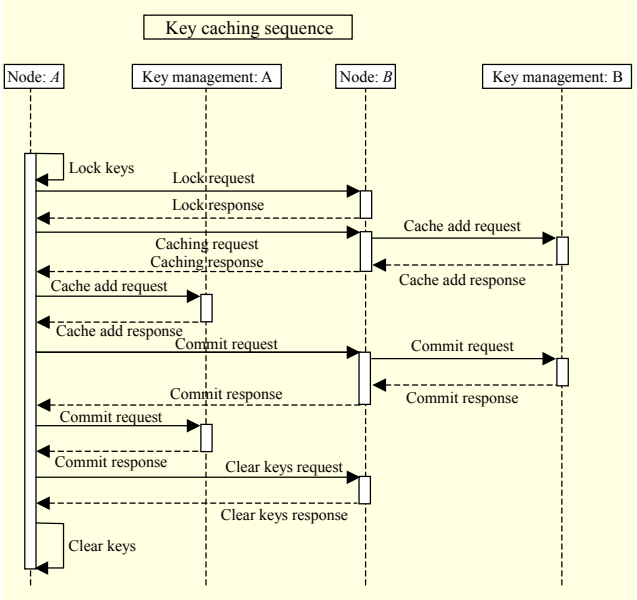

Fig. 3. Key caching protocol.

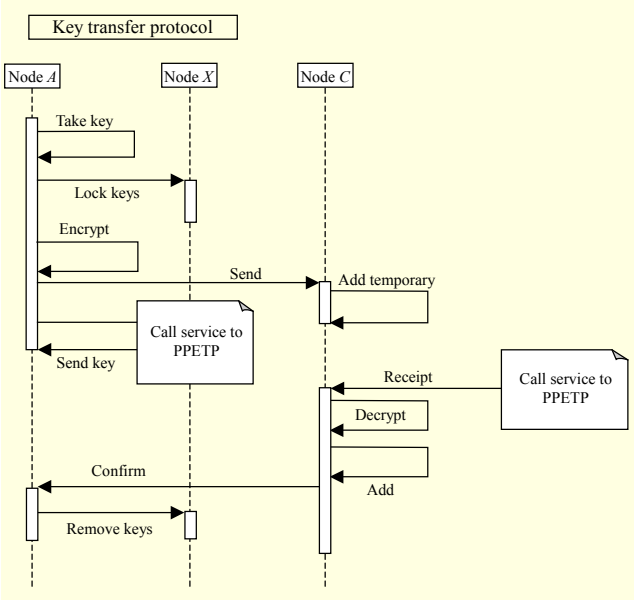

Fig. 4. Key transfer protocol.

comes into play.

Figure 4 illustrates the sequential process of our key transfer protocol; for instance, node A needs to communicate with node C based on symmetric key encryption. A set of secret bits must be exchanged before this communication can begin. Firstly, node A randomly selects such a set of secret bits, which is denoted by R, such that $r_1 \subseteq R\{0,1\}^n$, where r_1 is a subset of a secret bit contained in R and n is a number of bits comprising 0s and 1s. The aforementioned set R has been encrypted using a secret key that is shared between node A and a neighbor of node A, which we denoted by "X" in the figure. The corresponding ciphertext message can be calculated from $c_1 = r_1 \oplus k_{AX}$. The ciphertext message c_1 will be transmitted to node C over the Internet. To decipher the ciphertext message, node C has to wait for the corresponding key, k_{AX}, from node A, which is passed through the neighbor node, node X, along quantum point-to-point links using a call service from the PPETP. As a result, node A and node C will use a shared key, r_1, which is obtained from $r_1 = c_1 \oplus k_{AX}$ for further secure communication in a videoconferencing system.

C. PPETP

A PPETP is compiled and executed in the key management layer to solve the significant problem of how to share secret keys between two end nodes in a secure domain, whereby it is assumed that the two nodes rely on independent directed quantum channels. This protocol incorporates features of the hop-by-hop mechanism to transmit secret keys from a source to a destination across intermediate nodes. The PPETP protocol is designed to work together with the key routing protocol to obtain the appropriate route information that will help determine where a shared key is being sent in a secure manner.

Using the "*findnexthop*" function, a source node will look for all possible adjacent nodes along the available quantum links to search for the best route for a secure key transfer according to the routing information provided by the key routing protocol.

When the transmission path has been identified from the source node to the next hop, the corresponding key, k_{AX}, which is used for further ciphertext decryption of node C will be encrypted with a secret key shared between node A and the next hop. Let us assume that node B is the next hop between node A and node C. At node A, the corresponding key, k_{AX}, is to be first encrypted with a secret key, k_{AB}, which is the key that is shared between node A and node B. The result of this encryption is defined as $c_{AB} = k_{AX} \oplus k_{AB}$. When the ciphertext message c_{AB} reaches node B, node B has to perform two tasks — the decryption operation to retrieve key k_{AX} from c_{AB}, referred to as $k_{AX} = c_{AB} \oplus k_{AB}$, and the encryption operation to encrypt with a new key, k_{BC}, which is a key that is shared between node B and node C to generate a new ciphertext message, c_{BC}.

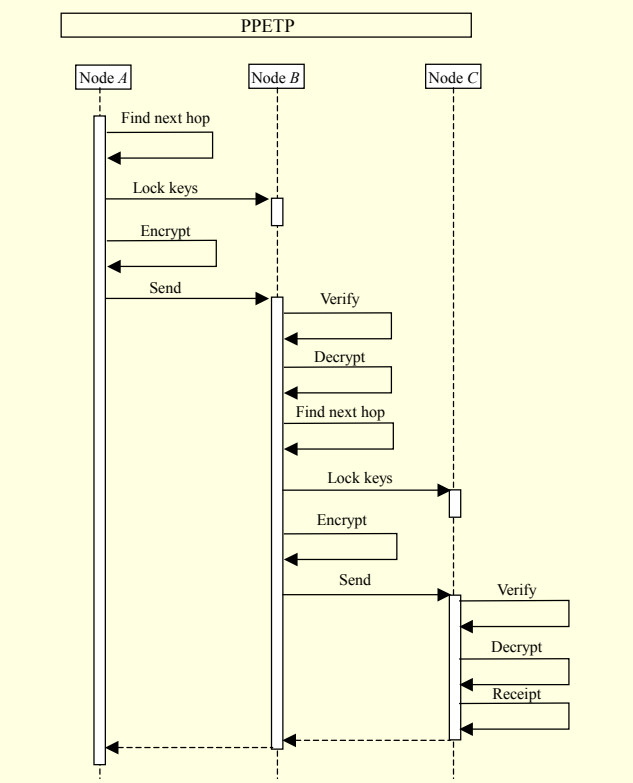

Fig. 5. PPETP.

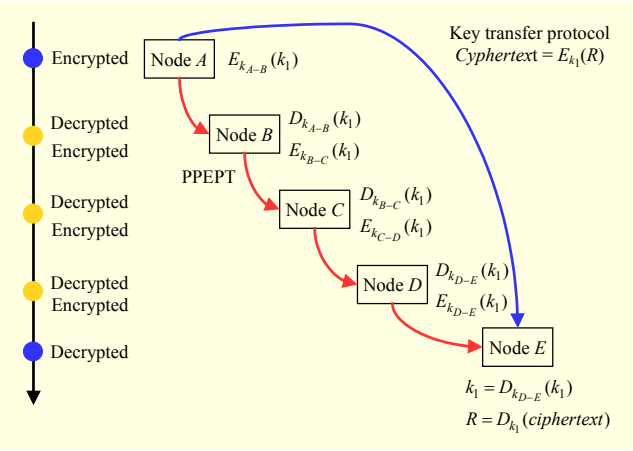

Fig. 6. Hop-by-hop key encryption technique.

To summarize, the process of transmitting keys from a source node to a destination node continues, along with the associated encryption and decryption operations, based on the route information obtained from the key routing protocol until a destination node has been reached. In the final step, node C uses the secret key k_{AX} to decrypt the ciphertext message c_1 to obtain the final secret key r_1, as well as informing the source node of its successful decryption of the ciphertext message c_1. A sequential diagram of the PPETP protocol and an overview

structure of key exchange across multiple hops is shown in Figs. 5 and 6, respectively.

D. Key Routing Protocol

A key routing protocol is used to determine optimal network data transfer and communication paths over network nodes in such a way that the shortest path from a local subnet to a destination node is always utilized. This protocol contains static route information (such as details of adjacent nodes), IP addresses (needed for sending data to the next hop), the total number of quantum links that exist in the quantum network, and the current status of routing.

There is a routing table that stores route information and contains the next hop association, in which the next hop is indicated as the hop to which the message is sent. This next hop performs the same look-up and forwarding functions, and so on until the message reaches the destination. For the key routing protocol, each node in the network knows only the IP address and the next hop information of adjacent nodes that are interconnected.

E. Key Distribution Protocol

A key distribution protocol provides a comprehensive connection that allows users to communicate with a key management server for quantum key requisitions, including requesting a new key for data encryption and decryption. The key distribution protocol was designed based on two new buffers known as In-buffer and Out-buffer. If end users request keys for data encryption, then the encrypted keys will be placed in the In-buffer, while the corresponding keys are automatically placed in the Out-buffer to serve as decrypted keys for decryption purposes.

To summarize, the main idea of our proposed secure key management is to overcome the limitations of a quantum point-to-point structure. Key management protocols not only ensure the creation of a trusted network but also offer key distribution among multiple users, even if such users do not connect or belong to the same quantum link.

V. Secure Videoconferencing System Based on Quantum Key Encryption

Quantum cryptography technology has been extensively advanced to the point that it can be used to increase the security of a videoconferencing system. The system works by establishing keys that provide instructions on how to encode and decode digital data streams, using key management services to manage the keys and distribute them to users. The keys will be discarded after successful communication.

1. Understanding Video Streaming

Many streaming media systems are based on the Real-time Messaging Protocol (RTMP) [35] for client and server communication. It is a basic data transfer protocol that works on top of the TCP/IP protocol. The main concept of the RTMP protocol is that it is designed to split payload data into fragments; the default fragment sizes are 128 bytes for video and 64 bytes for audio data. RTMP data sent by a client on port 1935 will be encapsulated by the RTMP protocol in such a way that it is going to be tunneled inside HTTP through port 80 and subsequently sent to a destination server.

2. OpenMeetings System

This paper has integrated the "OpenMeetings Videoconferencing" application [36] as an example of a real-time video demonstration with highly secure data transmission services based on quantum key encryption. Considering Fig. 7, the network structure is connected to the videoconferencing server, where it is located within a trusted network and listened to on port 1935. This videoconferencing server will wait for user requests and perform initial handshaking to establish a network connection between the videoconferencing server and clients. Therefore, all users must first login to the videoconferencing server according to the videoconferencing server's IP address. The entire contents of any data streaming will be transferred to a TCP/IP network to or from the videoconferencing server in parallel; any corresponding data transmission rates are dependent upon the network's bandwidth.

3. Transparent Encryption Software

Transparent encryption software is implemented to encrypt data streams sent by users before transmission over the Internet. The features of transparent encryption provide high levels of security in such a way that the power of keys is used for streaming-video encryption and decryption based on a one-time pad (OTP) encryption algorithm. Using quantum key encryption helps increase the security performance of secure video transmission over the Internet. The idea of transparent encryption software can be divided into four subsections.

A. Packet Filter

The packet filter works together with the kernel to collect network packets being sent to and from a network interface–based iptables configuration. In this case, the transparent encryption software examines only the videoconferencing

server's IP address with port 1935, while other IP addresses passing through this network interface will not be considered. Therefore, all packets being sent to and from the videoconferencing server must perform data encryption.

B. Packet Process

As data streams flow across the network, the packet process captures all packets passing through a network interface and analyzes each packet to identify the actual data payload according to the appropriate TCP segment format. Information about data payload has been continuously forwarded to payload encryption and decryption services to perform cryptographic operations.

C. Payload Encryption and Decryption

This is the process used to scramble the data payload before sending it out to the lower layer. All outgoing payloads must be encrypted with keys based on OTP encryption, while all incoming payloads must be decrypted. The corresponding keys will be loaded into a queue buffer, either an In-buffer or Out-buffer, which has been organized by the key manager.

D. Key Manager

This function deals with how to manage the keys and distribute them to the corresponding users within the transparent encryption domain correctly. It consists of a key provider (to import and export the keys to or from the queue buffer) and a key distribution client protocol (this works closely with the key management server to request the keys for cryptographic purposes).

In addition, transparent encryption software can install either within a client/server domain or through independent hardware encryption, but to achieve a better security result, this paper suggests the installation of a transparent encryption module as a separate part from the videoconferencing server and client computers to prevent Trojan attacks. As its functions, only the TCP payloads with port 1935 will be encrypted. Thus, the difference between the transparent encryption technique and VPN technique is that the VPN technique performs packet encryption for all incoming and outgoing message, whereas the transparent encryption technique performs packet encryption for only certain messages, based on a matching condition.

VI. Security Analysis and Performance Evaluation

In this section, we present and discuss the security analysis and performance evaluation of the proposed structure in more detail. This paper focuses on experimental results to demonstrate a videoconferencing system in real time; hence, they are performed under a trusted quantum network environment. Moreover, this paper includes a comparative evaluation at the end of this section that draws a comparison between the proposed framework and existing quantum networks.

1. Experimental Results

For the experimental setup, a videoconferencing server was assigned using http://192.168.20.2:5080/openmeetings, which was located within a private network. The transparent encryption was installed separately on different computers, which were located in front of the videoconferencing server and users' computers (see Fig. 1). It is important to keep the encryptors within visible range to maintain control and guarantee there is no attack between an intermediate contour line (that which connects a videoconferencing server and encryptor). Accordingly, the proposed framework has been

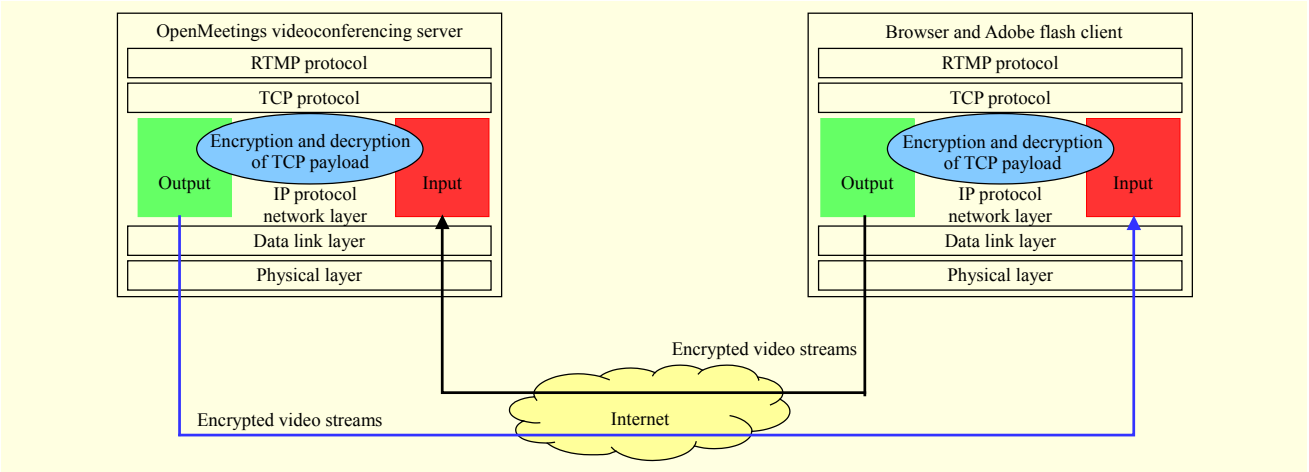

Fig. 7. Practical concepts for encrypted RTMP streams.

designed in such a way to prevent outsider attacks only, not insider attacks (as they are beyond the scope of this paper).

First-time users logging in to the videoconferencing server, known as "OpenMeetings Videoconferencing," have to fill out an authorized user account through the OpenMeetings Videoconferencing's web interface. After successfully logging in, users can communicate with other users through the OpenMeetings Videoconferencing's web interface. The program provides videoconferencing, instant messaging, document-sharing on a white board, screen sharing, and recording during meeting sessions. All encryption and decryption processes are received and responded to by the encryptors and key management servers. From our experimental results when integrating quantum key encryption with the OpenMeetings Videoconferencing system, the security has been improved by analyzing the achieved security level of the proposed scheme and assessing its performance. The results have shown that even the Wireshark program is unable to analyze and display the data content that has been transmitted over the Internet; in addition, third parties or eavesdroppers will not be able to understand the forms of user communication used.

2. Encryption Efficiency Analysis

The two main characteristics used to identify the efficiency of a videoconferencing system are security and the speed of data services. Security aspects usually deal with the encryption and decryption of video streams. In this context, an OTP-based quantum key encryption has been applied to fulfill the security requirements regarding the proposed framework. However, a time delay has been incurred due to encryption, decryption, and replacement. Figure 8 shows the times required for encoding and decoding for different message sizes with regard to the experimental setup. The setup has been tested under Ubuntu 12.04 based on an Intel (R) Xeon (R) CPU 2.67 GHz environment. According to the figure, big data sizes require more time for encryption and decryption, but in fact, the data stream transmission technique over the Internet generally relies on the maximum payload of a TCP segment. Each TCP/IP packet supports datagrams up to 64 kilobytes per second. As a result, the time required for the video encryption and decryption of a 64 KB data payload is approximately 0.098 s, which is an insignificant effect on the transmission delay in a videoconferencing system.

3. Comparative Analysis of Quantum Networks

The key management method for the secure videoconferencing system featured in this paper is based on a quantum network. One of the benefits of a quantum network is that it can offer a clear perspective of scalability and pave the way toward truly secure long-distance communication. Recently, many different types of quantum network, in terms of structure and protocols, have been created (see [27]–[34]); these networks aim to maintain the integrity of keys. A performance comparison of some of these different quantum network structures is illustrated in Table 1.

A. Security Model

A quantum network is defined to be a technology that prevents eavesdropping attacks over optical channels and establishes a trusted quantum network for secure communication. Based on the proposed framework, the VPN technique has been officially applied to establish a private network among key management servers for exchanging key information, such

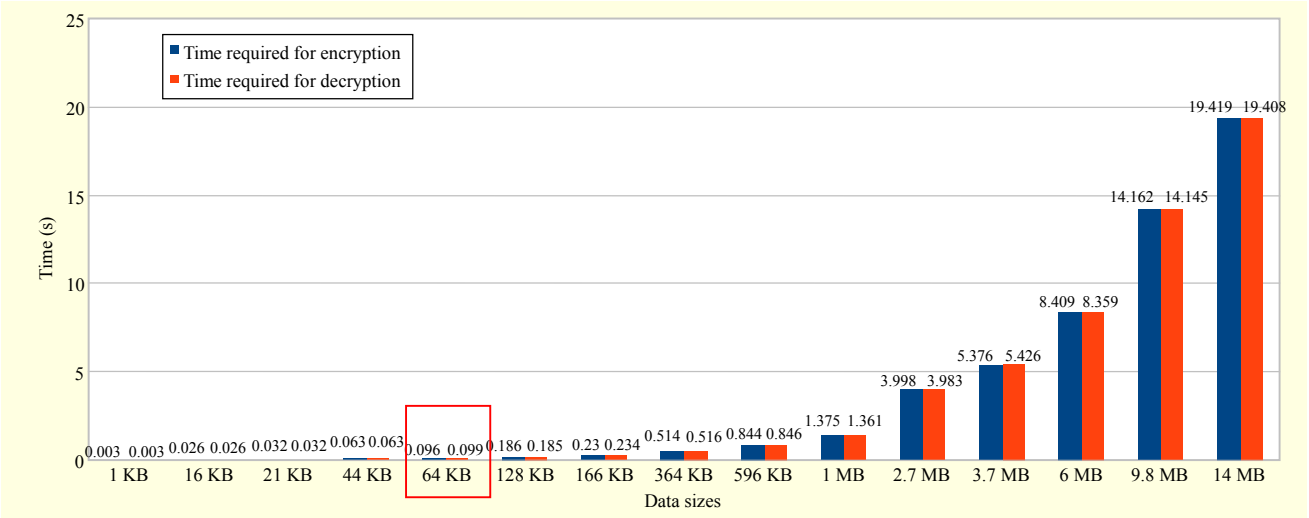

Fig. 8. Time required for encoding and decoding for various message sizes.

Table 1. Performance comparison of different quantum network structures.

Features	Key management properties based on differential QKD networks			
	DARPA network	SECOQC network	Tokyo network	Proposed network
Security model	VPN encryption, IKE	VPN encryption	VPN encryption	VPN encryption, Secure key management
Network topology	Star	Mesh	Mesh	Star
Protocol design	QKD protocol IPSec (IKE) QKD circuit management	QKD protocol Q3P QKD-NL QKD-TL	QKD protocol Key management agent Key management server	QKD protocol Key caching protocol Key transfer protocol PPETP Key routing protocol Key distribution protocol Key distribution client protocol
Key transportation	Hop-by-hop encryption			

as *key_id* and *pairing_node*, to check the correctness of keys; that is, whether a paired user obtained the same keys for cryptographic purposes. At the same time, the actual keys are kept in secure storage.

B. Network Topology

Figure 1 shows that the QKD layer comprises three physical nodes, node *A*, node *B*, and node *C*, with two optical links. Each node is connected to a central node, as in a star topology. However, there is a drawback to having such a star topology; that is, a central point of failure. Hence, the entire network is dependent upon the central node; if it fails, then the network may become inoperable. The mesh topology is fault tolerant, so it can ensure data privacy and security because every message travels along a dedicated link.

C. Protocol Design

In a traditional framework, there is still no standardization of key management protocols to provide flexible and feasible solutions across quantum networks and services. Most existing projects have established a quantum network based on their own proprietary protocols. According to the proposed framework and experimental setup, six new protocols have been developed, in Thailand, to create a field prototype of a trusted quantum network. These protocols have been operated under a key management layer, with the exception of the key distribution client protocol, which was executed under the application layer.

To summarize, the purpose of this paper is to design and develop a simple trusted quantum network to serve as a secure communication platform for a videoconferencing system.

D. Key Transportation

Basically, traditional quantum network structures have been constructed and composed from QKD systems featuring multiple point-to-point links; thus, keys are only shared among directed quantum channels. This is a limitation.

Key management services with hop-by-hop key encryption have been adapted to expand the scope of key distribution and overcome the limitation of quantum point-to-point link connections, whereby the key management services not only ensure the creation of a trusted network but also offer key distribution among multiple users, even if such users do not connect or belong to the same quantum link.

Table 2. Cryptographic primitives.

Factors	Types of algorithms			
	DES	3DES	AES	OTP
Key length	56 bits	112/168 bits	128/192/256 bits	Equal plaintext sizes
Block size	64 bits	64 bits	128/192/256 bits	Key streams
Cryptographic resistance	Assumption	Assumption	Assumption	Information theory

4. Foundation of Cryptography

Table 2 provides a synopsis of cryptographic primitives in a security context, and it illustrates that there is evidence that quantum cryptography combined with OTP encryption makes a videoconferencing system more secure and efficient, as well as enhancing privacy solutions.

VII. Conclusion

Efficient key management protocols for secure RTMP video

streaming toward a trusted quantum network present a new model for the transmission of video streams in a secure domain over the Internet.

The proposed method of key management, protocols, and well-designed video encryption algorithm belonging to this paper are all state of the art. The promise of the key management protocols along with their interoperability across a videoconferencing system based on RTMP encryption is a significant step toward data protection and privacy in electronic communication, because the protocols rely on quantum key encryption.

In addition to the strength of the key management in the proposed system, keys can be shared between two end points even though the end points do not connect or belong to the same quantum link. Furthermore, the application of hop-by-hop key encryption provides secure key transfer — a strategy that is able to overcome the limitation of quantum point-to-point link connection. A secure videoconferencing system has been considered based on a strong encryption algorithm. The system's keys and key management offer fast, secure, and reliable data, voice, and video transmission.

References

[1] D. Chin, *Next Generation Video Conferencing, Boosting Productivity of the Decentralized Workforce White Paper*, Arkadin Global Conferencing, Los Altos: CA, USA, 2011.

[2] F. Liu and H. Koenig, "A Survey of Video Encryption Algorithms," *Comput. Security*, vol. 29, no. 1, 2010, pp. 3–15.

[3] J. Shah and V. Saxena, "Video Encryption: A Survey," *IJCSI*, vol. 8, no. 2, Mar. 2011, pp. 525–534.

[4] B. Kurht and D. Kirovski, "*Multimedia Security Handbook*," Boca Raton, FL, USA: CRC Press, 2004.

[5] S. Frankel et al., "Guide to IPsec VPNs," Recommendations of the National Institute of Standards and Technology, NIST Special Publication 800-113, Gaithersburg, MD, USA, Dec. 2008.

[6] J.A. Pérez et al., "Quality of Service Analysis of IPSec VPNs for Voice and Video Traffic," *AICT/ICIW*, Guadeloupe, French Caribbean, Feb. 19–25, 2006, pp. 43–48.

[7] S. Park et al., "Characterizing the Impacts of VPN Security Models on Streaming Video," *CNSR*, Montreal, Canada, May 11–14, 2010, pp. 152–159.

[8] O. Adeyinka, "Analysis of IPSec VPNs Performance in Multimedia Environment," *IET*, Seattle, WA, USA, July 21–22, 2008, pp. 1–5.

[9] C. Du et al., "VCNF: A Secure Video Conferencing System Based on P2P Technology," *IEEE HPCC*, Dalian, China, Sept. 25–27, 2008, pp. 463–469.

[10] Z. Li et al., "Authentication to Peer-to-Peer Network: Survey and Research Direction," *NSS*, Gold Coast, Australia, Oct. 19–21, 2009, pp. 115–122.

[11] W. Fumy and P. Landrock, "Principle of Key Management," *IEEE J. Sel. Areas Commun.*, vol. 11, no. 5, June 1993, pp. 785–793.

[12] W. Diffie and M.E. Hellman, "New Direction in Cryptography," *IEEE Trans. Inf. Theory*, vol. 22, no. 6, Nov. 1976, pp. 644–654.

[13] W. Stallings, "*Cryptography and Network Security Principles and Practices*," 4th ed., New York, USA: Pearson Education, Inc., 2005.

[14] F. Wang, Z. Xiao, and J. Chen, "Research on Security of Trusted Network and its Prospects," *ETCS*, Wuhan, China, Mar. 6–7, 2010, pp. 256–259.

[15] M. Saadi et al., "Design and Implementation of Secure and Reliable Communication Using Optical Wireless Communication," *Frequenz*, vol. 68, no. 11–12, Nov. 2014, pp. 501–509.

[16] W.K. Wootters and W.H. Zurek, "A Single Quantum Cannot be Cloned," *Nature*, vol. 299, no. 5886, Oct. 1982, pp. 802–803.

[17] M. Pattaranantakul et al., "Secure and Efficient Key Management Technique in Quantum Cryptography Network," *ICUFN*, Phuket, Thailand, July 4–6, 2012, pp. 280–285.

[18] M.A. Nielsen and I.L. Chuang, "*Quantum Computation and Quantum Information*," 10th ed., Cambridge, UK: Cambridge University Press, Jan. 2011.

[19] S. Wiesner, "Conjugate Coding," *ACM Sigact News*, vol. 15, no. 1, 1983, pp. 78–88.

[20] C.H. Bennett and G. Brassard, "Quantum Cryptography: Public Key Distribution and Coin Tossing," *IEEE Int. Conf. Comput., Syst., Signal Process.*, Bangalore, India, 1984, pp. 175–179.

[21] C.H. Bennett, "Quantum Cryptography Using Any Two Non-orthogonal States," *Physical Rev. Lett.*, vol. 68, no. 21, May 1992, pp. 3121–3124.

[22] ISO/IEC 11770–5:2011, *Inf. Technol. – Security Techn. – Key Manag. – Part 5: Group Key Manag.*, ISO/IEC Standard, 2011. Accessed June 2, 2014. http://www.iso.org/iso/catalogue_detail.htm?csnumber=54527

[23] Standard Committee X9 Incorporated, *ANSI, X9.24 – Retail Financial Services Systematic Key Manag. – Part 1: Using Systematic Techn.*, ANSI Standard, 2004. Accessed June 2, 2014. http://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.24-1%3A2 009

[24] E. Barker et al., "Recommendation for Key Management – Part 1: General (Revision 3)," NIST Special Publication 800-57, Gaithersburg, MD, USA, July 2012, pp. 1–147.

[25] E. Barker et al., "Recommendation for Key Management – Part 2: Best Practices for Key Management Organization," NIST Special Publication 800-57, Gaithersburg, MD, USA, Feb. 2014, pp. 1–78.

[26] E. Barker et al., "Recommendation for Key Management – Part 3: Application-Specific Key Management Guidance (Revision 1)," NIST Special Publication 800-57, Gaithersburg, MD, USA,

Apr. 2014, pp. 1–94.
[27] BBN Technologies, "DARPA Quantum Network Testbed," Air Force Research Laboratory, New York, NY, USA, Final Tech. Rep. AFRL-IF-TR-2007-180, 2007.
[28] C. Elliott, "Building the Quantum Network," *New J. Physics*, vol. 4, no. 46, July 2002, pp. 46–55.
[29] M. Dianati and R. Alléaume, "Architecture of Secoqc Quantum Key Distribution Network," *ICQNM*, Guadeloupe, French Caribbean, Jan. 2–6, 2007, p. 13.
[30] M. Peev et al., "The SECOQC Quantum Key Distribution Network in Vienna," *New J. Physics*, vol. 11, no. 7, July 2009, pp. 1–37.
[31] O. Maurhart, *Q3P A Proposal*, SECOQC Project, 2006. Accessed June 2, 2014. http://www.secoqc.net
[32] M. Dianati and R. Alléaume, "Transport Layer Protocols for the SECOQC Quantum Key Distribution (QKD) Network," *LCN*, Dublin, Ireland, Oct. 15–18, 2007, pp. 1025–1034.
[33] ID Quantique, *Swiss Quantum Project*, 2011. Accessed June 2, 2014. http://swissquantum.idquantique.com
[34] M. Fujiwara et al., "Field Demonstration of Quantum Key Distribution in the Tokyo QKD Network," CLEO/IQEC/PACIFIC RIM, Sydney, Australia, 2011, pp. 507–509.
[35] H. Parmar and M. Thornburgh, "Adobe's Real Time Messaging Protocol," Copyright Adobe Systems Incorporated, Dec. 2012, pp. 1–52.
[36] OpenMeetings, *Open-Source Web-Conferencing*, Apache OpenMeetings Project, 2014. Accessed June 2, 2014. http://code.google.com/p/openmeetings/

Montida Pattaranantakul received her BS degree in computer science from Prince of Songkla University, Hat Yai, Thailand, in 2006 and her MS degree in computer applications from Bangalore University, India, in 2009. She is currently working as a research assistant at the Cybersecurity Laboratory, National Electronics and Computer Technology Center, a member of the National Science and Technology Development Agency, Pathumthani, Thailand. Her current research interests include key management, secure cryptographic protocol design, and network security.

Kittichai Sanguannam received his BS degree in electrical engineering from Prince of Songkla University, Hat Yai, Thailand, in 2001. His research interests include big-data processing, low level networking development and optimization. He is currently working as a senior engineer under the Research and Development Division of Triple T Broadband Public Co. Ltd., Nonthaburi, Thailand.

Paramin Sangwongngam received his BS degree in engineering from Prince of Songkla University, Hat Yai, Thailand, in 2000 and his MS degree in engineering from Chulalongkorn University, Bangkok, Thailand, in 2005. In 2006, he joined the Optical and Quantum Communications Laboratory, National Electronics and Computer Technology Center, National Science and Technology Development Agency, Pathumthani, Thailand, prior to moving on to the Photonics Technology Laboratory in 2013. His research interests span three main fields — error control coding, optical wireless communications, and security. He is currently focusing on the implementation of LDPC codes, visible light communications, quantum key management, and quantum key reconciliation. He has filed several patents in Thailand in the aforementioned fields. In addition, he has also led two research projects as a project manager.

Chalee Vorakulpipat received his BS degree in electronics engineering from King Mongkut's Institute of Technology, Ladkrabang, Bangkok, Thailand, in 1997 and his MS degree in information technology from Kasetsart University, Bangkok, Thailand, in 2000. He was awarded a scholarship from the Royal Thai Government to pursue a doctoral study. He earned his PhD degree in information systems from the University of Salford, Salford, UK, in 2008. He is currently the head of Cybersecurity Laboratory, National Electronics and Computer Technology Center, Pathumthani, Thailand. He has been involved in several projects in information security, mobile device management, social networking sites, ubiquitous computing, context-aware computing, e-health, and mobile application development. He has over thirty refereed publications in these areas that have appeared in conference proceedings and journals, such as ETRI Journal, Computers & Security, Advanced Engineering Informatics, Automation in Construction, and Knowledge Engineering Review. He also serves as a subcommittee member on issues regarding national information security of Thailand. In his academic role, he works as a lecturer for information systems courses at several universities across Thailand. He holds information security professional certificates including CISSP, CISA, and IRCA (ISMS Lead Auditor).