International Journal of
Computer & Software Engineering

Original Article Open Access

Special Issue: Computational Analysis and Modeling

Proposal for Multicast Cryptography and Its Prototype Cipher

Tomofumi Matsuzawa
Department of Information Sciences, Tokyo University of Science, Japan

Abstract

With the recent development in communication technologies, the field of cryptography for information confidentiality has significantly developed, and various ciphers have been proposed, implemented, and utilized. Most modern cryptography is based on end-to-end IP communication for one-to-one communication, and many cryptographic algorithms have been developed. In IP communication, although broadcast and multicast communication exist as one-to-many or many-to-many communication, cryptographic algorithms for multiparty communication are still underdeveloped, and research on Broadcast Encryption, which is a cryptographic scheme with a fixed sender, is actively being conducted. In this paper, we propose the requirements of multicast cryptography as a feasible multi-person cryptosystem over IP multicast, in which any user can be a sender at any time. In addition, a prototype cipher is proposed.

Publication History:

Received: November 01, 2021
Accepted: November 08, 2021
Published: November 10, 2021

Keywords:

Artificial Intellegence, Backend, Blockchain, ConvolutionalNetural Network, Database, Frontend,

Inroduction

In the field of cryptography, which is an ancient military technology, the development of computers in recent years has made it possible to perform complex cryptographic calculations at high speed, leading to the appearance of DES [1] and RSA [2] during the 1970s. With the advent of DES and RSA, modern cryptography, which applies cryptographic algorithms, has dramatically developed.

Although the Internet was initially designed for military purpose, it has become a huge communication network in which many ordinary people participate and numerous commercial services exist. As an increasing amount of information is being exchanged on the Internet, it has become necessary to protect personal information and other information confidential to third parties unrelated to the party to whom the information is being transmitted. As information began being exchanged on the Internet, the need arose for a technology to securely exchange personal and other types of information that should be kept secret from third parties. With the development of the Internet, modern cryptography has entered an era of practical application. Most modern cryptographic techniques are for one-to-one communication, and the main cryptographic schemes are symmetric key ciphers such as DES and AES [3], in which the sender and receiver have a common secret key, and RSA and ElGamal, in which the common secret key can be securely transmitted. In addition, broadcast cryptography, also known as footnote cryptography, has been attracting attention in recent years for one-to-many communications. Broadcast cryptography is a cryptographic scheme [4] proposed by Fiat et al. in 1993, in which users with different secret keys can decrypt the same data with different secret keys, whereas users without secret keys cannot. In broadcast cryptosystem, the user's secret key length, the header size, and the computational complexity are the evaluation items, and various schemes [5-8] have been proposed when considering the balance among them. In addition to cryptographic algorithms, studies have also been conducted to consider fraud tracing, key revocation efficiency, and a reduction in the communication volume [9-14].

In these broadcast ciphers, the sender is fixed and the sender holds the receiver's secret key. Therefore, in an environment where there are multiple senders, the receiver must prepare a separate secret key for each sender. Even in public-key broadcast cryptography, which solves these problems, the system administrator knows the secret keys of all recipients. In addition, even in public-key broadcast cryptography used to solve these problems, because the system administrator knows the secret keys of all recipients, the security of all communications is compromised if the system administrator has malicious intentions or the keys are leaked from the system administrator.

By contrast, IP multicast has attracted significant attention as a method for delivering information simultaneously and efficiently over the Internet in recent years. IP multicast is realized by defining a set of receivers as a group and assigning special IP addresses to them. However, because IP multicast delivery routing differs from existing IP routing, each router needs to have a separate IP multicast delivery function. Multicast currently has limited use for service providers; however, if it is to be used on a global scale, all routers in the delivery path must support IP multicast routing, and various issues such as scalability and compatibility remain. In addition, the number of supported applications is still low, which has affected the delay of IP multicast support on routers. As one of the reasons for the small number of supported applications, the sender cannot arbitrarily select the recipient on an IP multicast. Therefore, the sender can only deliver to the group chosen by the recipient, and the recipient cannot be selected by the sender. This has therefore hindered the birth of commercial applications.

In this paper, we define a multicast cryptosystem in which the sender can arbitrarily select a decryptable recipient on an IP multicast, any user can be a sender owing to the nature the IP multicast, and the sender can select a recipient to be decrypted using the public key of any recipient without having his/her own secret key. In addition, a prototype cryptosystem is proposed.

Related Work

Common-key cryptography

In common-key cryptography, encryption and decryption are conducted by the sender and receiver using a common key that is

[*]**Corresponding Author:** Prof. Tomofumi Matsuzawa, Department of Information Sciences, Tokyo University of Science, Japan; E-mail: t-matsu@is.noda.tus.ac.jp

Citation: Matsuzawa T (2021) Proposal for Multicast Cryptography and Its Prototype Cipher. Int J Comput Softw Eng 6: 168. doi: https://doi.org/10.15344/2456-4451/2021/168

secret to all but the sender and receiver (hereafter referred to as the secret common key). Although it has the advantage of high-speed computation, it is often used in combination with public-key cryptography (described below) because of problems in the delivery of the secret common key.

Public-key cryptography

Public-key cryptography is a cryptosystem that solves the problem of sharing a secret key, which has been a problem of conventional common-key cryptosystems that use a secret common key. The cryptographic key *PK* is made public, and only the decryption key *SK* is kept secret, making it difficult to obtain *SK* from *PK*. The message sender sends an encrypted message to the recipient using the recipient's public key *PK*, and the recipient decrypts the message using his/her own private key *SK*. Although RSA cryptography and ElGamal cryptography are popular public key ciphers, they assume one-to-one communication as well as common key cryptography.

Broadcast cryptography

Broadcast cryptography is a cryptographic scheme that applies the conventional cryptography realized for one-to-one communication to one-to-n communication. In the case of one-to-n communication using common-key cryptography or public-key cryptography, it has been necessary to send a message encrypted n times using the public key or secret common key of each recipient separately. In broadcast cryptography, the sender only needs to encrypt the message once, and only one ciphertext is generated; however, the message can be decrypted with the decryption key of each recipient, and the same plaintext can be obtained.

Broadcast cryptography has four main processes i.e., key generation (e.g., a secret key) (*Gen*), recipient registration with the center (*Reg*), encryption (*Enc*), and decryption (*Dec*), respectively. The broadcast cryptosystem is classified into two types: a symmetric key scheme, in which only a specific sender can send a message, and a public key scheme, in which any sender can send a message.

Symmetric key broadcast cryptography

Symmetric key broadcast cryptography is a method in which only a specific sender, such as a broadcasting station, transmits a message, and the sender shares a secret key with each recipient. The sender shares a secret key with each receiver. The message is encrypted using the secret key of the receiver who is allowed to decrypt the message at the time of transmission. The main approaches are Goodrich et al.'s [7] and Naor et al.'s [15] schemes.

In symmetric-key broadcast cryptography, the sender knows all of the secret keys, and thus if there is more than one sender, the receiver must keep a secret key for each one.

Public key broadcast cryptography

With this method, the system administrator generates a public key and a private key, and distributes the private key to the recipient. The sender encrypts the message using the public key of the recipient. Public-key broadcast cryptography is mostly based on pairing on elliptic curves, such as with Boneh et al.'s method [16] (BGW method) and Park et al.'s approach [17], which reduces the communication cost by 30% compared to the BGW method. However, in this scheme, the sender does not have any secret information, and thus Kanazawa et al. proposed a broadcast cipher with sender authentication [18] based on the BGW scheme.

With these schemes, a system administrator such as a center generates a public key and a private key for each recipient, thus allowing an entity (such as a system administrator) to decrypt all communications.

By contrast, Wu et al. proposed ad hoc broadcast encryption (AHBE) [19] and contributory broadcast encryption(CBE) [20], which are public key broadcast cryptosystems without a system administrator using pairing.

With AHBE, both the sender and receiver need to maintain the number of people and the public key of the whole system in addition to their own secret information (the public key of the receiver not allowed to decrypt is also needed), which is difficult to achieve practically. With CBE, the recipients in a group securely exchange security parameters, and each recipient generates its own private key and a common public key from the parameters of all recipients. Any sender encrypts the message using the public key, and all group members can decrypt the message using their own private keys. However, this method requires a separate means of securely exchanging security parameters among all group members, changes in the private key (and its public key) whenever a member changes, and the secure transmission and reception of security parameters whenever a new member is added. Wu et al. cite the use among friends in social networks as an example.

Three-party public key delivery

Joux [22] extended the DH public key delivery scheme to three parties using a pairing map on an elliptic curve. With this scheme, the three users are *A*, *B*, and *C* respectively, their (private and public) keys are (*a, aP*), (*b, bP*), and (*c, cP*) (where *P* is a fixed point on the elliptic curve), and *e* is a pairing.

$$
\begin{aligned}
A &: K = e(bP, cP)^{a} = e(P, P)^{abc}\\
B &: K = e(aP, cP)^{b} = e(P, P)^{abc}\\
C &: K = e(aP, bP)^{c} = e(P, P)^{abc}
\end{aligned}
$$

By applying the above calculation, we can obtain the common secret key $e(P, P)^{abc}$.

This scheme, in which no other entity has the secret key of each user, extends to three parties and is not applicable to n-to-n communication.

Multicast Cryptography

In this section, we describe the concept and requirements of our newly proposed multicast cryptography.

Concept

Multicast cryptography, like public key broadcast cryptography, assumes an environment in which the sender is not fixed, and is defined as a method in the upper layer of the IP layer, independent of IP multicast routing protocols such as PIM-SM[23] and CBT[24].[1]The encrypted message to be sent is not generated separately for

[1] Support for IPSec as IPv6 multicast will be considered in the future in consideration of the affinity.

each recipient, as in broadcast cryptography; however, the message received by all recipients is the same, and each recipient obtains the same plaintext with a different secret key.

The sender can choose any recipient to whom he/she wants to deliver the message, and uses the recipient's public key in combination with his/her own to encrypt the message. Any recipient with the public key chosen by the sender decrypts the message with its own private key, and everyone receives the same plaintext.

This differs from public-key broadcast cryptography in that the private and public keys are generated by the recipients themselves (and can be updated by the recipients themselves at any time), there is no system administrator who knows the private key, and there is no *Reg* process in broadcast cryptography. In addition, the sender can arbitrarily select the recipients, only the sender knows the list of the selected recipients, and the selected recipients need not to know the other selected or unselected recipients.

Requirements

When considering cryptography over an IP multicast, it is necessary to consider the following characteristics of an IP multicast.

1. The recipient can join and leave the group at any time.
2. No one entity can know all members of a group in real time.
3. The messages that can be received are the same for all recipients.
4. Recipients do not necessarily have a trustworthy relationship with each other.
5. It is desirable to have a sender authentication function because basically anyone can be a sender except for a sender-specified group.

Definition

Multicast cryptography is defined by a group of polynomial-time algorithms (*Gen, Enc, Dec*): key generation algorithm *Gen*, encryption algorithm *Enc*, and decryption algorithm *Dec*.

Gen (**Key generation algorithm**) Taking 1^{λ} (λ is a security parameter) as input, recipient *i* outputs its own public key PK_i and private key SK_i.

Enc (**Encryption algorithm**) The public key PK_i and the plaintext *M* of the recipient are input, and the ciphertext *C* is output. The ciphertext *C* is output.

Dec (**Decryption algorithm**) It takes as input the public key PK_i and private key SK_i of any recipient *i* who is allowed to decrypt by the sender, and the ciphertext *C*, and outputs the plaintext *M* or ⊥, which indicates that a decryption is not allowed.

Prototype Method

In this section, we introduce a cryptosystem extending the RSA as a prototype of a cryptosystem satisfying the requirements of our proposed multicast cryptosystem.

Algorithm

Preparation

Let each recipient be *1, 2, ..., n., n* for each recipient.

Each recipient selects its own secret key $\{p_i, q_i\}(i = 1, 2, ..., n)$ from large prime numbers.

Calculate the public key $PK_i = p_i q_i (i = 1, 2, ..., n)$, respectively, and publish them.

Sender

Determine the secret common key s that satisfies the following conditions for any $i(i = 1, 2, ..., n)$ that satisfies the following conditions.

$$s < p_i q_i \qquad (1)$$

Determine the encryption key *e* (preferably a prime number). Calculate the ciphertext *c* as follows:

$$c = s^e \bmod \prod_{i=1}^{n} PK_i \qquad (2)$$

Send *c* and *e* as a header and send message encrypted with *s*.

Receiver

The receiver *i* computes its own decryption key d_i satisfying the following:

$$ed_i = 1 \bmod \ (p_i - 1)(q_i - 1) \qquad (3)$$

Next, it computes *s* using the decryption key d_i.

$$s = c^{d_i} \bmod p_i q_i \qquad (9)$$

Finally, it decrypts the message with *s*.

Additional note

Here, *e* should be an integer value that is prime to $\prod_{i=1}^{n}(p_i - 1)(q_i - 1)$; however, because p_i - 1 and q_i - 1 are unknown to the sender, choose a prime value with a high probability of being prime to each other. If *e* is determined in advance, it is possible to select a secret key p_i, q_i^2 in which *e* and $(p_i$-1$)(q_i$-1) are mutually elementary at the time of the recipient's secret key generation, and d_i can be calculated at the preparation stage.

The method of *c* can be combined with the public key of the receiver to be decrypted, and the sender can arbitrarily select the decryptable receiver. The modulo of the exponential part of *c* follows $\prod_{i=1}^{n}(p_i - 1)(q_i - 1)$. The algorithm for encrypting the plaintext *M* with *s* is independent of our method and uses existing symmetric key cryptosystems.

Legitimacy of the decryption algorithm

Let any integer be $N = \prod_{i=1}^{n} P_i^{r_i}$ (*i = 1, 2, ..n*, where P_i are prime numbers that are prime to each other), and for any integer $a \in Z_n^*$, the following holds from Euler's theorem:

$$a^{\prod_{i=1}^{n} P_i^{r_n - 1}(P_i - 1)} = 1 \bmod N \qquad (5)$$

Equation(2) can be transformed as follows:

[2] For example, let $e = 65537 = 2^{16} + 1$ as the general public key of RSA.

$$\begin{aligned} c &= s^e \bmod \prod_{i=1}^{n} PK_i \\ &= s^e \bmod \prod_{i=1}^{n} p_i q_i \end{aligned} \tag{6}$$

Now, consider the case in which the secret keys (p_i, q_i) of each recipient are not prime (they use the same prime number as a factor). Note that $m \leq 2n$.

$$\prod_{i=1}^{n} p_i q_i = \prod_{i=1}^{m} r_i^{h_i} \tag{7}$$

For $m = 2n$, all factors of the secret key are prime to each other, and any $h_i = 1$.

From Equations (5) and (7), Equation (6) becomes Equation (8).

$$\begin{aligned} c &= s^e \bmod \prod_{i=1}^{n} p_i q_i \\ &= s^{\prod_{i=1}^{m} r_i^{h_i-1}(r_i+1)+e} \bmod \prod_{i=1}^{m} r_i^{h_i} \end{aligned} \tag{8}$$

Now, considering the decryption of receiver j, the right-hand side of Equation (4) becomes Equation (9) from Equation (8).

$$\begin{aligned} c^{d_j} &= s^{\left(\prod_{i=1}^{m} r_i^{h_i-1}(r_i-1)+e\right)d_j} \bmod p_j q_j \\ &= s^{\prod_{i=1}^{m} r_i^{h_i-1}(r_i-1)+ed_j} \bmod p_j q_j \end{aligned} \tag{9}$$

Because $\prod_{i=1}^{m} r_i^{h_i-1}(r_i-1)d_j$ has (p_j - 1)(q_j - 1) as a factor, we can use Equation (10).

$$\prod_{i=1}^{m} r_i^{h_i-1}(r_i-1)d_j = k(p_j-1)(q_j-1) \tag{10}$$

From Equations (10) and (3), we can see that Equation (9) is as follows and Equation (4) holds.

$$\begin{aligned} c^{d_j} &= s^{k(p_j-1)(q_j-1)+ed_j} \bmod p_j q_j \\ &= s^{k(p_j-1)(q_j-1)} \times s^{ed_j} \bmod p_j q_j \\ &= 1 \times s^{ed_j} \bmod p_j q_j \\ &= s \bmod p_j q_j \end{aligned} \tag{11}$$

Discussion of Prototype Cryptography

Security

Deriving the private keys p_i and q_i from the public key PK_i of any recipient i is as difficult as the RSA assumption. Because our method is completely equivalent to an RSA cryptosystem when the number of recipients $n = 1$, the security of our method is equivalent to RSA when the number of recipients.

In the case of multiple recipients, the number of decryption keys increases as much as the number of recipients, so unfortunately the difficulty of decryption is slightly lower.

Collusion resistance

In this section, we consider the security of the case in which there are multiple receivers who are allowed to decrypt the message by the sender, and the receivers collude with each other. Let $n(n \geq 2)$ be the number of receivers that the sender permits to decrypt. In addition, let $k(1 \leq k < n)$ be the number of recipients who collude. From these k private and public keys, e, c, and s, determine the private key of one of the remaining n-k recipients.

Let $P = \prod_{i=1}^{k} PK_i$ be the product set of public keys of the colluders, among which the colluders know all but N in the following equation:

$$c = s^e \bmod NP$$

Transforming the above, there exists an integer k satisfying the following:

$$s^e - c = kNP$$

Although the colluder can find the value of kN, finding the prime factor of N (the secret key of the recipient who does not participate in the collusion) from this is as difficult as the prime factorization problem. However, if the secret keys p_i, q_i, and kN held by the colluders are not prime to each other, then the private key p_i or q_i is likely to be the same as the private key of a recipient who does not participate in the collusion.(The key is different if k is not prime to p_i or q_i or to each other by chance.) In this case, if the colluder maintains a list of recipients who are allowed to decrypt the message by the sender, it will be possible to identify the key of any of the recipients.

Because this check can be conducted even for k = 1, the recipient who can identify the key can also confirm that there is a recipient who has the same private key as his/her own among the recipients who have been allowed to decrypt the message, and thus it is desirable for each recipient to check and, if necessary, update his/her own private and public keys after each reception.

Value of secret symmetric key s and cryptographic key *e*

When the secret symmetric key s and the public cryptographic key e are small, at least for $e \leq n$ (where n is the number of recipients allowed to decrypt), the values of s^e and s^e mod $\prod_{i=1}^{n} PK_i$ will be the same, and thus s can be obtained using the usual power root operation.

In addition, if the same s is used for a long time and at least e ciphertexts using that s are collected, s can be decoded using the Chinese remainder theorem, as in the RSA identical plaintext problem, and thus it is desirable to determine the value of s variably to a certain extent.

Header size

Because the modulo of c is $\prod_{i=1}^{n} PK_i$, it increases according to O(n) as the number of recipients increases.

However, because the value of the modulo $\prod_{i=1}^{n} PK_i$ is not included in the header, the value of c is extremely small if e is taken well, and the header size can be reduced. The size of c is an issue to be discussed in the future.

Signature

The same procedure as used for RSA signatures can be applied for sender authentication. The sender encrypts M with his/her own d, and the receiver decrypts it with e. By Contrast, the recipient decrypts M with e. The decryption result is M, which is accepted.

Let i be the sender. The sender calculates d_i using the encryption key e used for message encryption and its own secret keys p_i and q_i.

$$ed_i = 1 \bmod\ (p_i - 1)(q_i - 1)$$

The sender computes D using the plaintext M applied for sender authentication, and sends it to the receiver.

$$D = M^{d_i} \bmod p_i q_i$$

Each recipient obtains M' using the sender's public key $PK_i = p_i q_i$.

$$M' = D^e \bmod PK_i$$

If $M = M'$, it is accepted. The validity and security of this authentication is the same as for RSA signatures.

Comparison with public-key broadcast cryptography

Unlike public-key broadcast cryptography, multicast cryptography does not require an entity to manage all keys. Therefore, any user can become a sender at any time. In addition, because the sender can specify any group of recipients at the time of transmission, the proposed method can be used in an IP multicast and a secret inter-group communication. In the proposed prototype scheme, the secret and public keys of RSA cryptography can be used as they are, and there is no need to prepare a separate key for multicast cryptography.

Conclusion

In this paper, we proposed the concept of multicast cryptography, which is a cryptographic method for any sender to communicate with any known receivers simultaneously. As a prototype of the multicast cryptosystem, we proposed an extension of RSA. Although the prototype is a simple extension of the existing RSA cryptosystem, we hope that it will become a pioneering approach in the field of many-to-many communication.

Competing Interests

The author declare that there is no competing interests regarding the publication of this article.

References

1. Matsui M (1993) Linear cryptanalysis method for DES cipher. EUROCRYPT'93.
2. Rivest RL, Shamir A, Adleman L (1978) A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM 21: 120-126.
3. Anderson R, Biham E, Knudsen L (1998) Serpent: A proposal for the advanced encryption standard. AES algorithm submission.
4. Fiat A, Naor M (1994) Broadcast encryption. CRYPTO '93.
5. Halevy D, Shamir A (2002) The LSD broadcast encryption scheme. CRYPTO.
6. Attrapadung N, Kobara K, Imai H (2003) Sequential key derivation patterns for broadcast encryption and key predistribution schemes. Asiacrypt.
7. Goodrich MT, Sun JZ, Tamassia R (2004) Efficient tree-based revocation in groups of low-state devices. CRYPTO.
8. Jho NS, Hwang JY, Cheon JH, Kim MH, Lee DH, et al. (2005) One-way chain based broadcast encryption schemes. Eurocrypt.
9. Boneh D, Sahai A, Waters B (2006) Fully collusion resistant traitor tracing with short ciphertexts and private keys. Eurocrypt.
10. Ogawa K, Hanaoka G, Imai H (2007) Traitor tracing scheme secure against adaptive key exposure and its application to anywhere TV service. IEICE Transaction on Fundamentals of Electronics, Communications and Computer Science.
11. Wang X, Liao Z (2010) A secure encryption protocol for ad hoc networks. Third International Symposium on Information Science and Engineering.
12. Zou X, Xiang J (2013) Dynamic broadcast encryption scheme with revoking user. Wuhan University Journal of Natural Sciences 18: 499-503.
13. Canarda S, Phan DH, Pointcheva D, Trinh VC (2018) A new technique for compacting ciphertext in multi-channel broadcast encryption and attributebased encryption. Theoretical Computer Science 723: 51-72.
14. Balakrishna C (2021) Hybrid broadcast encryption and group key agreement protocol with precice cipher texts. Turkish Journal of Computer and Mathematics Education 12: 984-988.
15. Naor D, Naor M, Lotspiech J (2001) Revocation and tracing schemes for stateless. Receivers. Crypto.
16. Boneh D, Gentry C, Waters B (2005) Collusion resistant broadcast encryption with short ciphertexts and private keys. CRYPTO.
17. Park JH, Kim HH, Sung MH, Lee DH (2008) Public key broadcast encryption schemes with shorter transmissions. IEEE Transactions on Broadcasting 54: 401-411.
18. Kanazawa F, Okamoto T, Okamoto E, Ohkawa N, Doi H, et al. (2007) Boardcast encryption with sender authentication and its duality. Proceedings of Intenational Conference on Convergence Information Technology.
19. Wu Q, Qin B, Zhang L, Ferrer JD (2010) Ad hoc broadcast encryption. CCS '10 Proceedings of the 17th ACM conference.
20. Wu Q, Qin B, Zhang L, Ferrer JD, Farras O (2011) Bridging broadcast encryption and group key agreement. Asiacrypt.
21. Gentry C, Waters B (2009) Adaptive security in broadcast encryption systems (with short ciphertexts). EUROCRYPT.
22. Joux A (2000) A one round protocol for Tripartite diffie-hellman. ANTS.
23. Fenner B, Handley M, Holbrook H, Kouvelas I, Parekh R, et al. (2016) Protocol independent multicast - sparse mode (PIM-SM): Protocol specification. IETF RFC7761.
24. Ballardie A (1997) Core-based trees (CBT version 2) multicast routing - Protocol specification. IETF RFC2189.

This article was originally published in a special issue:

Computational Analysis and Modeling

Handled by Editor:

Prof. Shigeo Akashi
Department of Information Sciences
Tokyo University of Science
Japan