IBC Secured Key Partition for A Peer-To-Peer Network in Delivery of Message

Dushyant B. Sisode

Department of Computer science and Engineering
Lord krishana College of Technology,Indore, Mp.,452003,India

Dr.Sanjay Thakur

Department of Computer science and Engineering
Lord krishana College of Technology, Indore, Mp.,452003,India

Abstract- Key partition scheme focuses on the confidentiality maintained in using the secret key for delivery of message in p2p networks. Identity based cryptography (IBC) was introduced into peer-to-peer (P2P) networks recently for identity verification and authentication purposes. A current IBC-based solutions could not address the problem of secure private key issuing.

In this paper we present an IBC infrastructure setup phase, a peer registration solution using Shamir's (k, n) secret sharing scheme, and a secure key issuing scheme, which adopts key generate center (KGC) and key privacy authorities (KPAs) to issue private keys to peers securely in order to enable the IBC systems to be more acceptable and applicable in real-world P2P networks. We propose IBC based Secured Key Partition For peer-to-peer network in delivery of message.

Keywords – Secret sharing scheme, Secret key Partition,Peer-To-Peer,Identity based cryptography

I. INTRODUCTION

Due to distributed, self-organization and self maintenance nature,P2P networks are extremely vulnerable to a large spectrum of attacks, mainly due to the lack of a certification service responsible for peers identity verification and for authentication purposes. By using traditional certificate-based public key infrastructure we can solve some of the problems by verifying the authenticated nodes identities and by issusing public keys to the nodes for certification.The node churn is highly frequent in the P2P network, many nodes that stored certificates may quickly become invalid, hence PKI based security protocol is difficult to be deployed. Each node requires large amounts of space to store public key certificates, which can be difficult to implement in practice.

The secured P2P overlay communication is efficient if the overlay nodes have a common, shared key for securing the communication. This is difficult to achieve in dynamic P2P overlay networks, as a new key must be generated every time a overlay node membership change occurs in order to preserve forward secrecy.

From the 1980s, To make entities' keys available to others in a trusted fashion, thereby enabling a qualitative improvement in the assurance of communications and protection and transactions carried out over the Internet, public-key infrastructures (PKIs) have been widely anticipated as a primary. Certificate-based authentication has become common practice in certain contexts, particularly in conjunction with SSLprotected web sites. In recent years, however, many commentators have lamented the fact that PKI has not achieved more pervasive adoption and deployment. Some have concluded that PKI is a failure or does not address users' primary security needs. Opinions differ on the reasons for these results, but most can be distilled into a few general categories.

Lots of research and studies have focused on introducing IBC into P2P security applications, but the proposed schemes suffered from attacks against key issuing phase. In real-world P2P networks, it is very important to keep in secret whether the private key corresponding to a certain identity has been requested. Hence, it is important to have an anonymous key issuing scheme without secure channels.

The identity based cryptography (IBC) can simplify the key management process in P2P networks significantly as compared with the PKI technique,. The identity of a peer in P2P overlay networks is used to create its public key, thus avoiding the use of any certificates. These IBC-based systems are scalable, simple to administer, and each user can carry out anytime/anywhere encryption, establish secure communication channels, prove its identity to other nodes, verify protected messages and produce a form of signature with non-repudiation property.

II. PROPOSED WORK & DESIGN SYSTEM

A. Related Work-:

IBC uses the user's identity as the public key. The private keys of the users are issued by a key generate centre

(KGC) after verifying the user's credentials. IBC was introduced in 1984 by Shamir ; however, the first practical encryption scheme (IBE) was not available until 2001 which was developed by Boneh and Franklin. Though IBC overcomes the problems of the traditional PKI, it suffers from some inherent problems, one of which is the secure channel requirement: key issuing requires secure channel to avoid eavesdropping. In 2001, Boneh and
Franklin addressed secure key issuing problem using multiple key issuing authorities. After that, many key issuing protocols without secure channels were proposed.

Several studies have been focused on introducing IBC into P2P security applications. Lu et al. in combined distributed hash tables (DHTs) and identity based encryption (IBE) to defend against man-in-the-middle attacks, however, the scheme assumed that each node has had a pre-assigned unique identifier, and has obtained the corresponding private key through a secure offline channel. This is expensive and difficult to achieve in a large scale P2P overlay network. In [9], Lua proposed a hybrid security protocol using IBE to resist the Sybil attacks,Ryu et al. in proposed ID assignment protocols based on IBC to permit the acquisition of node IDs to be tightly regulated in order to mitigate the Sybil attacks, but these two schemes still suffered from the attack against key issuing phase. Likir presented by Aiello et al. signs messages with IBS in Kademlia-based P2P networks, however the authors supposed every system user had already obtained a private key and did not consider the key issuing problem. In real-world P2P networks, it is important to have a key issuing scheme in order to keep in secret whether the private key corresponding to a certain identity has been requested. In this paper, a secure key issuing scheme for P2P networks, which addresses the shortcomings of and makes IBC more applicable in the real world is presented.

B. Proposed Work-:

In this paper we propose a IBC Secured Key Partition For A Peer-To-Peer Network in Delivery of Message. We design the system which is intended to be more secure, reliable and efficient in message transfer between peer nodes. System include a setup of IBC infrastructure system, in which we mention how each peer will interact with each other and their responsibilities. We introduce a peer authentication protocol which can register peers using Shamir's secret sharing scheme. We also propose a secure key distribution protocol which issue private keys securely without the need of secure channels. Afterward we present an optimal compression technique known as deflate which is to be applied to encrypted message that is cipher form due to private key applied on message.
Deflate will provide an additional security to encrypted message and consume less bandwidth to get transfer among peers. The protocol enables IBC more acceptable and applicable in real-world P2P networks.

By using BFT protocol we use an algorithm to key privacy authorities (KPAs) using which can remove the malicious KPAs and find out newly sub situational KPAs. Afterwards, we will analyze how to assign and adjust the threshold of Shamir's secret sharing scheme in our peer registration protocol to prevent the system from collusionattacks and denial-of-service (DoS) attacks . Finally, we conclude about proposed system that is more efficient in term of bandwidth and provide security to encrypted message by using compression technique.

C. Design of system-:

We propose our security scheme in four section setup of system, peer registration, secure keying and maintenance of system.

In system setup phase we describe how KGC and KPA work in the beginning of the system. In Peer Registration phase and secure keying phase we describe how a peer joins the system . Adopting the threshold cryptography to register users, and using secure key issuing scheme to issue private keys.In system maintenance phase the maintenance of KPAs takes place.

ID_A :	Peer A's identity (ID)
K_A :	Peer A's private key
$Proof_A$:	Peer A's proof of the registration
.	Concatenation
$SS(x,k)$	Secret share of secret x in Shamir's (k,n) threshold secret sharing scheme
$MAC(x,K)$	Keyed message authentication code of data x and key K
$\{X\}K_A$	A string X signed by peer A
$Thres_{KPA}$	Minimum number of KPAs system possesses
$PK_A(ID)$	Partial key of peer ID issued by A
$Pzl(x)$	A puzzle generated using Seed x
$Sln(x)$	Solution of Puzzle x

Terminology and assumptions of system setup

KGC: It is a trusted core node which would acts as the centre of the system architecture; it provides peer registration and key distribution service. We assume that it has been highly fault tolerant, secure and always available.

KPA: In order to provide the key privacy service in distributed manner, n nodes are selected as Key Privacy Authorities (KPAs) in the key distribution phase; these are not as reliable as KGC. Malicious attackers can compromise some of these nodes act as insider attacks.

Peer: A peer is an ordinary node in P2P networks, which is could be subject to all kinds of attacks.

D. Requirements-:

There are four requirements for system:

i. Secure peer verification and registration: It should provide a method to deal with attacks such as man-in-the-middle attacks, collusion attacks and DoS attacks during the peer registration phase.

ii. Secure key assigning: It should provide a method to issue keys securely without secure channels during the key distribution phase, and defend against replay attacks, manin- the-middle attacks and insider attacks.

iii. KPA Verification : It should provide method to identify malicious KPAs, remove them and add new alternativeKPA.

iv.Robust system maintenance : The system must provide a online method to add new substitution of KPAs remove, identify malicious KPAs.

E. Types of Attack-:

We design the security scheme to defend against three kind of possible attacks for which:

Insider attack: In P2P networks, KPAs may also be malicious. The system should deal with insider attacks, in which a minority of KPAs has been compromised by attackers.

DoS attack: Malicious peers in P2P network can simply drop the messages between KPAs and the requesting peer, which makes the peer difficult to collect sufficient secret shares.

Collusion attack: An adversary can launch a collusion attack by compromising many paths between KPAs and the requesting peer, then compute peer's ID and the proof of registration.

F. Setup of system-:

There is one KGC node and n bootstrap KPA nodes at the setup phase. First, KGC selects a master key, publishes its identity (ID) and specifies the system parameters; Secondly, KGC assigns to each bootstrap KPA node an ID and a corresponding private key based on IBC scheme via a secure offline channel. Note that, the secure offline channel is only required in the system bootstrap phase, since with its ID and private key, a KPA can communicate with the KGC through a secure channel established based on IBC.

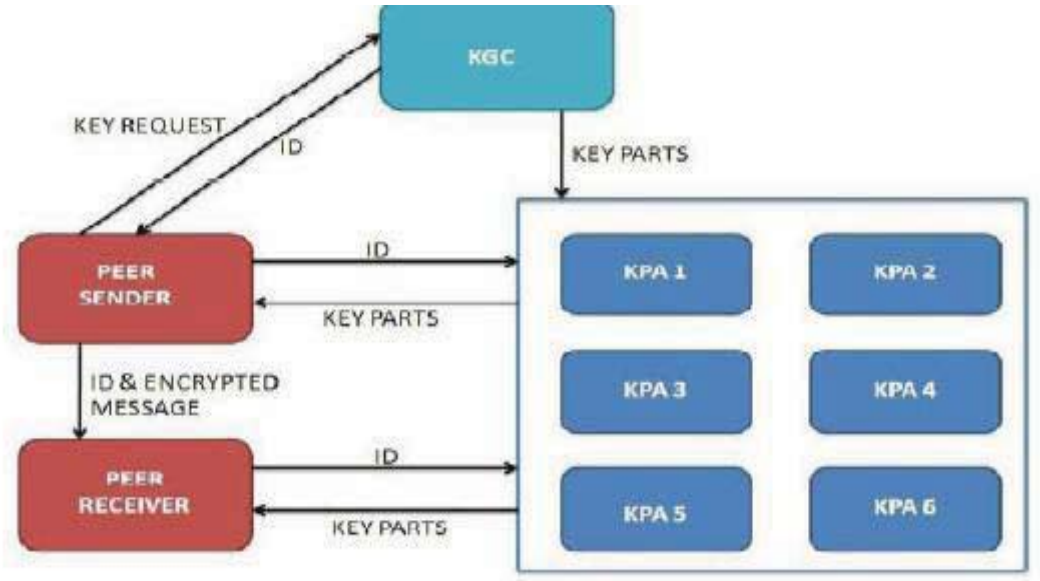

Figure1 : System Architecture

G. Peer registration-:

Before joining the network, at first a peer A should get registered to the KGC . We adopt Shamir's (k, n) threshold secret sharing scheme to secure this process.

The protocol is described as follows:

Step1: A ! KGC: N
Step2: KGC ! KPA : SS(IDA · ProofA, k),N
Step3: KPA ! A : SS(IDA · ProofA, k),N

Request: When the peer A wishes to join the network, it must first get registered from KGC by sending a request to KGC.

Distribution: After KGC receives the request, it generates IDA and ProofA for A. In particular, ProofA can be a keyed message authentication code of IDA. After that, KGC divides IDA and ProofA into n secret shares using Shamir's (k, n) threshold secret sharing scheme. Then KGC distributes those n secret shares to n KPAs respectively. It is very difficult for the adversary to obtain sufficient secret shares in a P2P network if we divide the registration data and set an appropriate threshold k.

Reconstruction: After receiving the secret shares from KGC, KPAs send them to A. After A gets at least k different secret shares, IDA and ProofA can be reconstructed. If the peer does not get sufficient secret shares, it may run the peer registration protocol

H .Secure key issuing-:

After end of registration phase, a peer obtains its ID. The next step is to describe how KGC issues a private key to a peer securely without the requirement of secure channels, and how a peer constructs its private key securely from the KGC and KPAs. Shamir's secret sharing scheme we used can also be utilized here, however, with KGC and KPAs in the system, we can make the key issuing phase more secure. We present a protocol which utilizes IBC secure key partition schemes below. Those schemes use one KGC and multiple KPAs for issuing the private keys to the users. KPAs participate in the key generation phrase, they assign the joining peer partial private keys. A registered peer can obtain its private key securely by collecting partial private key from KGC and KPAs. Those schemes avoid the need for secure channels, and the adversary who wants to obtain the private key must

compromise not only KGC but also many KPAs. In our scheme, Saxena's scheme is followed therefore it can easily be extended to other schemes. Our scheme is described as follows:

Step1: A ->KGC: Request, IDA, ProofA,N
Step2: KGC ->A : Partial key from KGC, N
Step3: A ->KPA : Request, IDA, ProofA,N
Step4: KPA ->A : Partial key from KPA, N

System setup: KGC selects its private key and specifies the system parameters. KPAs collaboratively run a key generation and distribution protocol , and share a secret s such that any k KPAs can construct it with their own secret shares.

Peer registration: As the system setup process is updated, in the peer registration process, IDA and ProofA are generated in a new way, but we can still utilize the protocol Request: A sends a request with its proof of registration as well as a nonce to KGC to obtain the partial private key.

KGC response: On receiving A's request, KGC checks the proof to verify whether A has been registered or not, if the result is positive, KGC responses with a partial private key.

Blind KPA request: After receiving the partial private key from KGC, A randomly selects some KPAs and requests them in parallel to provide key privacy service by sending a request; KPA response: Each KPA authenticates A and issues a partial private key to it.

Key retrieval: On receiving at least k partial private keys from different KPAs, A combines them and then unbinds the resulting value to produce the private key; The scheme above is secure against replay attacks, man in-the-middle attacks and insider attacks, and more details can be found in . It can easily be incorporated with other secure key issuing schemes such as that use KPAs to protect the private key.

III. SYSTEM MAINTENANCE

In P2P real-world networks, KPAs may also be may be potentially compromised to perform insider attacks and malicious with relatively low probability and to address this problem we adopt a scalable Byzantine fault tolerant authentication scheme.KGC dynamically maintains a relay group (RG) to perform distributed challenge-response authentication. RG members are randomly selected in the setup phase, thus only a limited number of RG members can, with high probability, be compromised by manin- the-middle attacks. Our KPA authentication scheme, as described formally below, can be executed in three steps: claim announcement, distributed authentication and result generation.

Step1: KGC ! Pi : {IDi, PKKGC(IDi)}KKGC
Step2.1: Pi ! KPA : IDi, PKKGC(IDi)
Step2.2: KPA ! Pi : {PKKPA(IDi)}KKPA
Step3: Pi ! KGC : {{PKKPA(IDi)}KKPA}KPi

1. Claim announcement: When the authentication process begins, KGC announces the claim to all its RG members and asks them to verify if KPAi indeed possesses the secret si, which is generated in the system setup phase. KGC sends to RG member Pi a randomly selected peer's ID IDi and its partial key from KGC.

2. Distributed authentication: According to the received peer's ID IDi, each RG member Pi independently challenges KPAi by sending a request that simulates the secure key issuing phase. KPAi has the capacity of generating the corresponding partial key if and only if it holds the corresponding secret. Afterwards, KPAi returns the partial key to Pi. At the
end of this stage, each RG member Pi obtains a partial key from KPAi.

3. Result generation: Each RG member Pi responds to KGCs authentication request with its partial key from KPAi. Afterwards, KGC can verify these received partial keys by checking the equation described in . If at least [N
]+1 partial keys can be successfully verified, KPAi indeed possesses the secret si; otherwise, KPAi is not the

genuine owner of si, and should be removed from the set of KPAs. Here, N denotes the total number of peers contained in KGC's RG. After these authentication and removing operations, the number of KPAs may fall below a threshold which is minimum number of KPAs system possesses, thus we should utilize KGC's RG to find new authenticated KPAs until the threshold is satisfied. We utilize client puzzle to verify KPA candidates. A peer wishing to be a KPA is challenged by the RG members. KPA candidates completing the puzzles of all RG members are accepted as a new KPA.

IV. ANALYSIS AND PERFORMANCE

We propose our key partition scheme to address the inside attacks and to prevent the system from collusion attacks and DoS attacks. By using three primary performance metrics we characterize the system and performance:

i.Threshold of the peer registration scheme is defined as the minimum number of secret shares a peer needs to collect from KPAs. ii.The metric reflects the system's effectiveness under collusion attacks and DoS attacks. iii.Maximum number of peers that system can support is the number of peers a KGC or KPA can support within 1 second response time period. This metric indicates the systems efficiency and scalability.

V. CONCLUSION

In our work we focus on key exchanging phase of a cryptographic schema,where the keys are given to the authenticated nodes involved in communication. we have mainly concentrated on key issuing part of cryptography as previously proposed techniques do not address the issue effectively. Public Key Infrastructure(PKI) can be used for key issuing however there is problem of managing certificates in PKI as it becomes cumbersome. Hence we go for ID based cryptography, which is efficient compared to PKI.

In this paper we have proposed a IBC secured key partition for a peer-to-peer network in delivery of message ,It can provides a peer registration service using Shamir's (k, n) secret sharing scheme. We utilize a secure private key assigning protocol, which adopts KGC and KPAs to issue keys to peers securely. We maintain the security of KPAs, and authenticate KPAs using BFT protocol it also remove malicious ones and find out alternate ones to join in the system using.

REFERENCES

[1] E. Sit and R. Morris, "Security considerations for peer-to-peer distributed hash tables," *in* IPTPS, *2002, pp*. 261–269.
[2] A. Shamir, "Identity-based cryptosystems and signature schemes," *in* CRYPTO, 1984, pp. 47–53.
[3] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in CRYPTO, 2001,pp. 213–229.
[4] B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang, and S. Yoo, "Secure key issuing in id-basedcryptography*," in ACSW Frontiers*, 2004, pp. 69- 74. pp. 674–678.
[5] R. Gangishetti, M. C. Gorantla, M. L. Das, A. Saxena, and V. P. Gulati, "An efficient secure key issuing protocol in idbased cryptosystems*," in ITCC (1*), 2005.
[6] A. Saxena, "Threshold ski protocol for id-based cryptosystems," in IAS, 2007, pp. 65–70.
[7] Z.-L. Lu, G.-H.; Zhang, "Wheel of trust: A secure framework for overlay-based services," ICC, pp.1148–1153, 2007.
[8] I. Stoica, R. Morris, D. R. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-topeer lookup service for internet applications," in SIGCOMM, 2001, pp. 149–160.
[9] E. K. Lua, "Securing peer-to-peer overlay networks from sybil attack," in ISCIT'07, Sydney, Australia, 2007.
[10] S. Ryu, K. R. B. Butler, P. Traynor, and P. D. McDaniel, "Leveraging identity-based cryptography for node id assignment in structured p2p systems," in AINA Workshops (1), 2007, pp. 519–524.
[11] L. M. Aiello, M. Milanesio, G. Ruffo, and R. Schifanella, "Tempering kademlia with a robust identity based system," in Peer-to-Peer Computing, 2008, pp. 30–39.
[12] A. Shamir, "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613, 1979.
[13] I. Baumgart and S. Mies, "S/kademlia: A practicable approach towards secure key-based routing," in ICPADS, 2007, pp. 1–8.
[14] R. Chen, W. Guo, L. Tang, J. Hu, and Z. Chen, "Scalable byzantine fault tolerant public key authentication for peer-to peer networks," in Euro-Par, 2008.
[15] M. J. Freedman and R. Morris, "Tarzan: a peer-to-peer anonymizing network layer," in CCS, 2002, pp. 193–206.
[16] B. Parno, D. Wendlandt, E. Shi, A. Perrig, B. M. Maggs, and Y.-C. Hu, "Portcullis: protecting connection setup from denial-of-capability attacks," in SIGCOMM, 2007, pp. 289– 300.
[17] P. Maymounkov and D. Mazi`eres, "Kademlia: A peer-to-peer information system based on the xor metric," in IPTPS, 2002.
[18] An Examination of Asserted PKI Issues and Pro-posed Alternatives -John Linn, RSA Laboratories,Bedford, MA, USA Marc Branchaud, RSA Security Inc., Vancouver, BC,Canada.