Industrial Science ISSN 2347-5420
Volume-1 | Issue-10 | April-2015 Available online at www.lsrj.in

IDENTITY-BASED CONDITIONAL PROXY RE-ENCRYPTION WITHOUT RANDOM ORACLES FOR SECURING BROKERLESS PUBLISH/SUBSCRIBE SYSTEMS

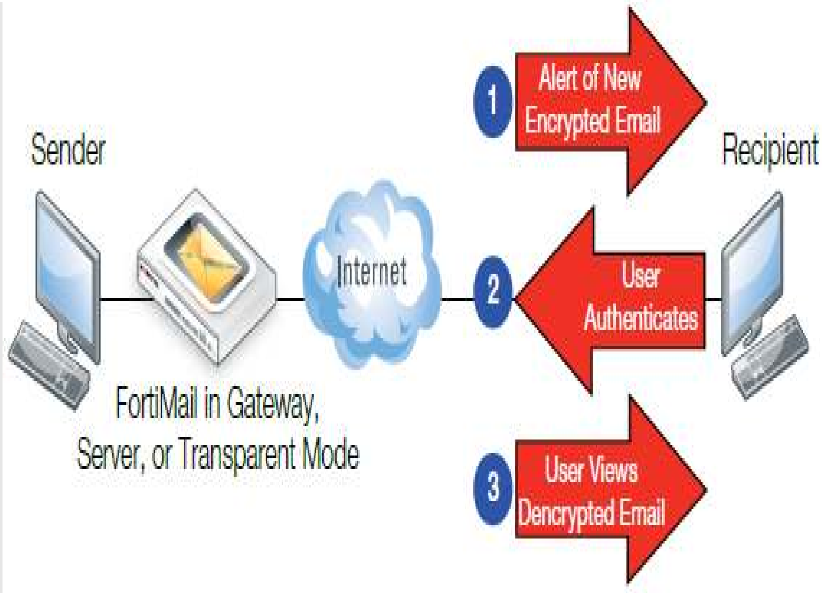
Shagoufta Taskeen
PG Scholar(M.Tech)

Short Profile

Shagoufta Taskeen is a PG Scholar(M.Tech)

Co- Author Details :

Vidyarani H. J.
Assistant Professor


Sender
Internet
FortiMail in Gateway,
Server, or Transparent Mode
1
Alert of New Encrypted Email
2
User Authenticates
3
User Views Dencrypted Email
Recipient

ABSTRACT:

Identity-based encryption (IBE) is an important primitive of identity-based cryptography. As such it is a type of public-key encryption in which the public key of a user is some unique information about the identity of the user.In an identity-based proxy re-encryption (IB-PRE) scheme, a semi-trusted proxy can convert a ciphertext under Alice's identity into a ciphertext for Bob. The proxy does not know the secret key of Alice or Bob, and also does not know the plaintext during the conversion. However, some scenarios require handle a fine-grained delegation. In this paper, by using the identity-based encryption (IBE) technique of Boneh-Boyen, we propose a new identity-based conditional proxy re-encryption (IBCPRE) scheme, which enables Alice to implement fine-grained delegation of decryption rights, and thus is more useful in many applications. Our scheme has significant advantages in both computational and communicational than Shao et al.'s IBCPRE scheme.

KEYWORDS

peer-to-peer, broker-less, security,identity-based encryption.Proxy re-encryption, Identity-based conditional proxy re-encryption, Bilinear maps.

I.INTRODUCTION

A publish-subscribe (pub-sub) network service is a wide-area communication infrastructure that enables information dissemination across geographically scattered and potentially unlimited number of publishers and subscribers. A wide area pub-sub system is often implemented as a collection of spatially disparate nodes communicating on top of a peer-to-peer overlay network.The provisioning of basic security mechanisms such as authentication and confidentiality is highly challenging in a content-based publish/subscribe system. Authentication of publishers and subscribers is difficult to achieve due to the loose coupling of publishers and subscribers. Likewise, confidentiality of events and subscriptions conflicts with Content-based routing. This paper presents a novel approach to provide confidentiality and authentications in a brokerless content-based publish/subscribe system. In a proxy re-encryption (PRE) scheme, a semi-trusted proxy can convert a ciphertext under Alice's publickey into a ciphertext for Bob. During the whole conversion, the proxy neither know the secret key of Alice or Bob nor learn the plaintext. Since the concept of PRE was introduced by Blaze et al.[1], a number of PRE protocols have been proposed in the context of public-key encryption [2]. In order to solve the problem of key-distribution, Green and Ateniese [8] proposed identity-based proxy re-encryption (IB-PRE) scheme,which allow a proxy to translate a ciphertext under Alice's identity into a ciphertext under Bob's identity. The proxy is able to re-encrypt all ciphertexts from Alice to Bob during the traditional IB-PRE scheme. As a result, it is difficult for Alice to implement any further fine-grained delegation of decryption rights. Suppose Alice wants Bob only to decrypt part of her ciphertexts. In this case, Alice can only trust the proxy to implement her policies by re-encrypting the legitimate ciphertexts. This approach is infeasiblebecause of the high-trust requirements on the proxy (for example, the proxy can be corrupted or collude with Bob). To address this issue, Weng et al. [9] introduced conditional proxy re-encryption (C-PRE). In such systems, ciphertexts are generated according to a specific condition (also canbe called keyword, attribute and any meaningful bit-string), and the proxy can transform a ciphertext only if the relevant condition is satisfied. As a consequence, Alice can flexibly appoint Bob the decryption capability based on the condition attached to the message.

In this paper, we propose an identity-based conditional proxy re-encryption (IBCPRE), which is based on Boneh-Boyen's [10] identity-based encryption (IBE) schemeand C-PRE technique. In IBCPRE scheme, the ciphertext is encrypted under the user's identity and a condition. Only the ciphertext satisfying one condition can be transformed by the proxy and

then can be decrypted by Bob. Compared with Shao et al.'s [6] IBCPRE scheme, our scheme has significant advantages in both computational and communicational costs than [5].

Proxy re-encryption has also been researched based on the identity.

Based on Boneh and Franklin's identity-based encryption system [3], Green and Ateniese [8] proposed a CPA-secure IB-PRE scheme in random oracle model. But many researchers have expressed doubts about the random oracle model. Fortunately, Chu and Tzeng proposed an IB-PRE against chosen ciphertext attack (CCA) without random oracles . Those two schemes satisfy the properties of unidirectionality, multi-use and non-interactivity.

Recently, Wang et al.[8] introduced an IB-PRE scheme by using the random padding techniques to ensure the non-malleability of the ciphertexts instead of using a hash encapsulated on a bilinear pairing, which is unidirectional, multi-use and IND-CCA2 secure in the random oracle model.

II.IDENTITY-BASED ENCRYPTION

Identity-Based Encryption (IBE) takes a breakthrough approach to the problem of encryption key management. IBE can use any arbitrary string as a public key, enabling data to be protected without the need for certificates. Protection is provided by a key server that controls the dynamic generation of private decryption keys that correspond to public identities and the key servers base root key materialIdentity-Based Encryption (IBE) dramatically simplifies the process of securing sensitive communications. For example, the following diagram illustrates how Alice would send a secure email to Bob using IBE:

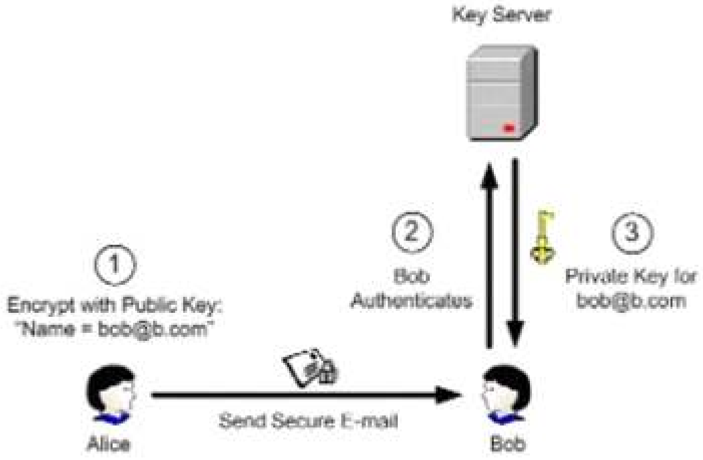

Key Server
2
Bob Authenticates
3
Private Key for bob@b.com
1
Encrypt with Public Key: "Name = bob@b.com"
Send Secure E-mail
Alice
Bob

Fig. 1. Identity-based encryption

Step 1: Alice encrypts the email using Bob's e-mail address, "bob@b.com", as the public key.
Step 2: When Bob receives the message, he contacts the key server. The key server contacts a directory or other external authentication source to authenticate Bob's identity and establish any other policy elements.
Step 3: After authenticating Bob, the key server then returns his private key, with which Bob can decrypt the message. This private key can be used to decrypt all future messages received by Bob.

Note that private keys need to be generated only once, upon initial receipt of an encrypted message. All subsequent communications corresponding to the same public key can be decrypted using the same private key, even if the user is offline. Also, because the public key is generated using only Bob's email address, Bob does not need to have downloaded any software before Alice can send him a secure message.

The Math Behind IBE

The mathematical foundation of IBE is a special type of function called a "bilinear map." A bilinear map is a pairing that has the property:

Pair(a o X, b o Y) = Pair(b o X, a o Y)

The operator "o" is multiplication of a point on an elliptic curve by integers. While multiplication itself (e.g., calculatingaoX) is easy, the inverse operation (finding a given X and aoX) is practically impossible. Two examples of bilinear maps are the Weil Pairing and the Tate Pairing.

The IBE algorithm consists of four operations:

1.Setup, which initializes a key server
2.Encrypt, which encrypts a message for a given user
3.Key Generation, which generates a private key for a given user
4.Decrypt, which given a private key, decrypts a message

Although identity-based encryption has been proposed some time ago, only recently pairing-based cryptography(PBC) has laid the foundation of practical implementation of identity-based encryption. Pairing-based cryptography establishes a mapping between two cryptographic groups by means of bilinear maps. This allows the reduction of one problem in one group to a different usually easier problem in another group. We utilize bilinear maps for establishing the basic security mechanisms in the pub/sub system and, therefore, introduce here the main properties. Let G_1 and G_2 be cyclic group of order q, where q is some large prime. A bilinear map is a function e^ : $G_1XG_1 \rightarrow G_2$ that associates a pair of elements G_1 to elements in G_2.A bilinear map satisfies the following conditions:

III.DEFINITIONS FOR IDENTITY-BASED CONDITIONAL PROXY RE-ENCRYPTION

An identity-based conditional proxy re-encryption (IBCPRE) consists of the following algorithms:

- Setup(1^k) : On input a security parameter 1^k, this algorithm generates global public parameters paramswhich is distributed to users and the master secret key msk.

- KeyGen msk ID: Takes the master secret key mskand a user's identity ID as input, this algorithm generates a private key dIDcorresponding to the identity ID.

- ReKeyGen(dID,w,ID_2): Run by user1, takes a private key dID_1corresponding to user1's identity ID_1, a condition w, and $user_2$'s identity ID_2 as input, this algorithm generates the re-encryption key $rk_{1\rightarrow 2}$

.?Enc (ID, w, m) : On input a user's identity ID , a condition w, and a message mfrom the message space. This algorithm outputs a second-level ciphertext CTassociated with condition wunder identity ID.

- ReEnc(CT_1, $rk_{1\rightarrow 2}$) Run by the proxy, on input a second-level ciphertext CT_1associated with w under identity ID_1 , and a re-encryption key $rk_{1\rightarrow 2}$. this algorithm outputs a first-level ciphertext CT_2 under identity ID_2 .

Dec(CT,dID): On input a ciphertext CT, and a private key dID corresponding to the identity ID . This algorithm outputs a message m in the message space or the error symbol $\perp$.

IV.SECURITY NOTIONS

We say that an IBCPRE scheme is semantic security under adaptive-ID and chosen-ciphertext attacks if no polynomial bounded adversary Ahas a non-negligible advantage against the challenger Cin the following IBCPRE-CCA game.

a) Second-level ciphertext.

In this paper, second-level ciphertext means original ciphertext.

Setup:The challenger C runs algorithm Setup (1k)and gives the public parameters paramsto adversary A.

Phase 1: The adversary Aadaptively issues queries q1, ..., qnwhere qi is one of:

- Private key generation query (ID): Cruns algorithm KeyGento generate private key dID and gives it to A.

- Re-encryption key generation query(ID_1,w,ID_2): On input (ID_1, w,ID_2) by A, the challenger returns the re-encryption key $rk_{1\to 2}$ ←rekeygen(dID,w,ID_2) to A , where dID is from KeyGen algorithm corresponding to ID_1 .

- Re-encryption query (ID_1, w,ID_2,CT_1):C runs algorithm reEnc(CT_1, $rk_{1\to 2}$)and returns the ciphertext CT_2 to A, where $rk_{1\to 2}$ is generated from reKeyGen (dID,w,ID_2) algorithm.

- Decryption query (CT,ID): On input (CT,ID) by A, the challenger returns m ←Dec(CT,dID) to A, where dID is the private key corresponding to ID. Challenge.Once Adecides phase 1 is over. It outputs a target identity ID* , a target condition w*, and two equal-length messages m0 m1∈M, here Mdenotes the message space. Cflips a random coin $\delta \in \{0, 1\}$, and returns the challenge ciphertext CT*=Enc(ID *,m_δ,w*) to A.

Phase 2.The adversary Acontinues to issue queries as in phase 1, challenger Cresponds the queries as in phase 1.

Guess. Finally, A outputs a guess $\delta' \in \{0, 1\}$, and wins the game if $\delta' = \delta$. Adversary Ais subject to the following restrictions during the challenge phase.

1)Acan not issue the private key generation query (ID*)to obtain the target private key dID*.
2)Acan not issue the re-encryption key generation query (ID*,w*,ID') if ID' appears in a previous private key generation query.
3)A can issue decryption queries on neither (CT*, ID*) nor (ID',ReEnc(ID*,w*,ID',CT*))
4)Acan not issue re-encryption query (ID*,w*,ID',CT*) if ID' appears in a previous private key generation query.

b) First-level ciphertext.

Since the first-level ciphertext can not be re-encrypted another ciphertext, Ais allowed to get any re-encryption keys. Furthermore, given these re-encryption keys, A can re-encrypt ciphertext by himself, and hence there is no need to provide the re-encryption query for him. The process of game is similar to the second-level ciphertext's except there is no re-encryption query.

During the challenge phase, adversary Ais subject to thefollowing restrictions.

1)A can not issue the private key generation query (ID*) to obtain the target private key dID*.
2)Acan not issue re-encryption query (ID*,w*,ID') if ID' appears in a previous private key generation

query.

3)Acan not issue decryption query (CT*,ID*).

V.APPROACH OVERVIEW

For providing security mechanisms in pub/sub, we leverage the principles of identity-based encryption to support many-to-many interactions between subscribers and publishers.

In our approach, publishers and subscribers interact with a key server. They provide credentials to the key server and in turn receive keys which fit the expressed capabilities in the credentials. Subsequently, those keys can be used to encrypt, decrypt, and sign relevant messages in the content-based pub/sub system, i.e., the credential becomes authorized by the key server. A credential consists of two parts: 1) a binary string which describes the capability of a peer in publishing and receiving events, and 2) a proof of its identity. The latter is used for authentication against the key server and verification whether the capabilities match the identity of the peer. While this can happen in a variety of ways, for example, relying on challenge response, hardware support, and so on, we pay attention mainly at expressing the capabilities of a credential, i.e., how subscribers and publishers can create a credential. This process needs to account for the many possibilities to partition the set of events expressed by an advertisement or subscription and exploits overlaps in subscriptions and publications. Subse-quently, we use the term credential only for referring to the capability string of a credential.

The keys assigned to publishers and subscribers, and the ciphertexts, are labeled with credentials. In particular, the identity-based encryption ensures that a particular key can decrypt a particular ciphertext only if there is a match between the credentials of the ciphertext and the key. Publishers and subscribers maintain separate private keys for each authorized credential.

3.

The public keys are generated by a string concatenation of a credential, an epoch for key revocation, a symbol 2 fSUB; PUBg distinguishing publishers from subscri-bers, and some additional parameters described in Section 5. The public keys can be easily generated by any peer without contacting the key server or other peers in the system. Similarly, encryption of events and their verification using public keys do not require any interaction.

The ciphertexts of the encrypted event are then signed with the private key of the publisher, as shown in Fig. 2.

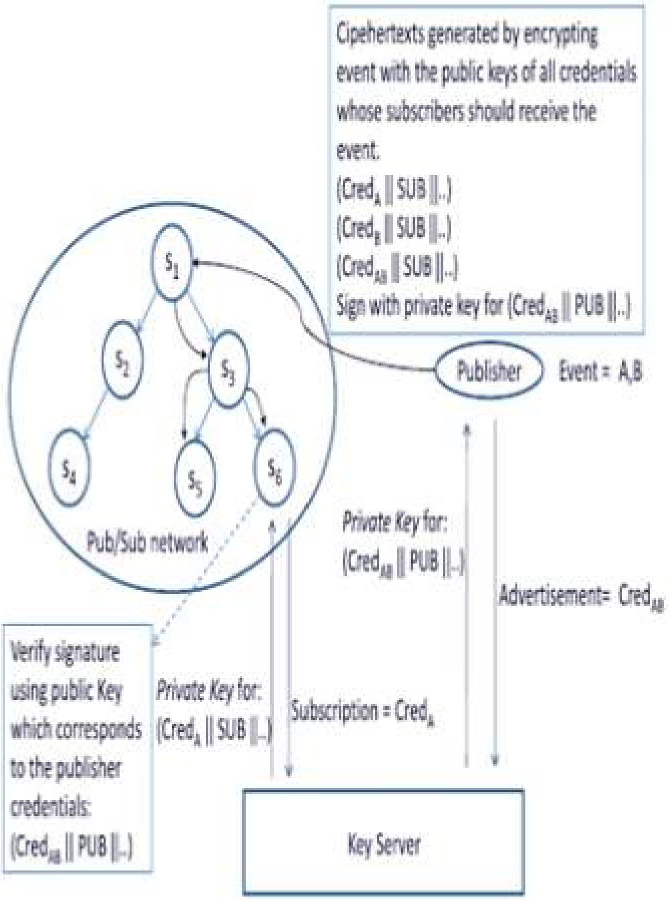

Cipehertexts generated by encrypting event with the public keys of all credentials whose subscribers should receive the event.
(Cred$_A$ || SUB ||..)
(Cred$_B$ || SUB ||..)
(Cred$_{AB}$ || SUB ||..)
Sign with private key for (Cred$_{AB}$ || PUB ||..)
S$_1$
S$_2$
S$_3$
S$_4$
S$_5$
S$_6$
Pub/Sub network
Publisher
Event = A,B
Private Key for:
(Cred$_{AB}$ || PUB ||..)
Advertisement= Cred$_{AB}$
Verify signature using public Key which corresponds to the publisher credentials:
(Cred$_{AB}$ || PUB ||..)
Private Key for:
(Cred$_A$ || SUB ||..)
Subscription = Cred$_A$
Key Server

Fig. 2. Approach overview: Publisher has credentials to publish events with two attributes A and B. Subscriber s6 has credentials to receive events with attribute A.

VI.CREATIONOF CREDENTIALS

In the following, we will first describe the creation of credentials for numeric and string attributes. Further extensions to handle complex subscriptions are discussed subsequently.

NUMERIC ATTRIBUTES

The event space, composed of d distinct numeric attributes, can be geometrically modeled as a d-dimensional space such that each attribute represents a dimension in the space. With the spatial indexing approach, the event space is hierarchically decomposed into regular subspaces, which serve as enclosing approximation for the subscriptions, advertisements, and events. The decomposition procedure divides the domain of one dimension after the other and recursively starts over in the created subspaces. Fig. 3 visualizes the advancing decomposition with the aid of a binary tree.

Subspaces are identified by a bit string of “0” and “1”s. A subspace represented by dz_1 is covered by the subspace represented by dz_2, if dz_2 is a prefix of dz_1. Subscription or advertisement of a peer can be composed of several subspaces. A credential is assigned for each of the mapped subspace. For instance, in Fig. 3, f_2 is mapped to two subspaces and therefore possesses two credentials f000; 010g. An event can be approximated by the smallest

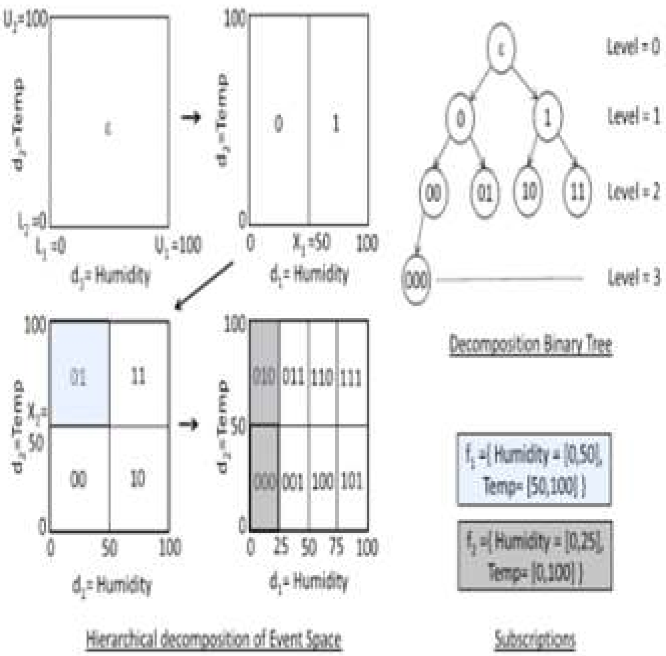


Fig 3.Numeric attribute

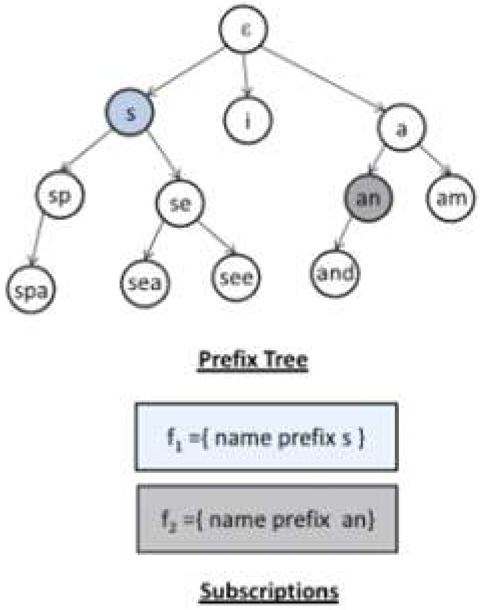


Fig 4 Prefix matching

VII.PUBLISHER/SUBSCRIBER AUTHENTICATION AND EVENT CONFIDENTIALITY

The security methods describe in this section are built upon ciphertext-policy attribute-based encryption (in short CP-ABE) scheme proposed by Bethencourt et al. . In particular, our modifications 1) allow publishers to sign and encrypt events at the same time by using the idea of the identity-based signcryption proposed by Yu et al. 2) enable efficient routing of encrypted events (from publishers to subscribers) by using the idea of searchable encryption proposed by Boneh , and 3) allow subscribers to verify the signatures associated with all the attributes (of an event) simultaneously. Our modifications do not change the basic structure of the CP-ABE scheme and preserves the same security strength.

Publisher keys. Before starting to publish events, a publisher contacts the key server along with the credentials for each attribute in its advertisement. If the publisher is allowed to publish events according to its credentials, the key server will generate separate private keys for each credential. Let Credi;j denote the credential with label j for the attribute Ai,

$$Pu^{p}_{i,j} := \left(Cred_{i,j} \,\|\, A_i \,\|\, PUB \,\|\, Epoch\right).$$

$$Pr^{p}_{i,j} := \left(g_2^{\alpha} \left(u' \prod_{k \in \Gamma_{i,j}} u_k \right)^{\gamma_{i,j}}, g^{\gamma_{i,j}} \right) =: \left(Pr^{p}_{i,j}[1], Pr^{p}_{i,j}[2]\right).$$

VIII.SECURE OVERLAY MAINTENANCE

In the following, we propose a secure protocol to maintain the desired pub/sub overlay topology without violating the weak subscription confidentiality.

The secure overlay maintenance protocol is based on the idea that in the tree, subscribers are always connected according to the containment relationship between their credentials, for example, a subscriber with credential 00 can only connect to the subscribers with credentials 0 or 00.

A new subscriber s generates a random key SW and encrypts it with the public keys Pu^{s}_{i};j for all credentials that cover its own credential, for example, a subscriber with credential 00 will generate cipher texts by applying the public keys Pu^{si};0 and Pu^{s}_{i};00. The generated cipher texts are added to a connection request (CR) and the request is forwarded to a random peer in the tree. A connection is established if the peer can decrypt any of the cipher texts using its private keys.

Filling the security gaps. By looking at the number of cipher texts in the connection request, a peer can detect the credential of the requesting subscriber s. For example, a subscriber with credential 00 can only connect to 0 or 00, and therefore, a connection request will have two cipher texts, whereas the connection request for 000 will have three cipher texts. In the worst case, a subscriber has a credential of the finest granularity. This can be covered by $\log_2$ðZ_iÞ other credentials, and therefore, a connection request contains in the worst case that many cipher texts. To avoid any information leak, cipher texts in the connection request are always kept in Oð$\log_2 Z_i$Þ (O ðL_iÞ for prefix matching) by adding random cipher texts if needed. Furthermore, the cipher texts are shuffled to avoid any information leak from their order.

A different random key SW is used for the generation of each cipher text to avoid any information leak to the peer who has successfully decrypted one of the cipher texts and, thus, has recovered the random key SW. Otherwise, the peer can try to generate cipher texts by encrypting the (recovered) SW with public keys for Oðlog$_2$ Z$_i$Þ (likewise OðL$_i$Þ) credentials and can easily determine the random cipher texts in the connection request and, thus, the

credentials of the requesting subscriber s. Finally, to avoid an attacker to generate arbitrary connection request messages and try to discover the credential of other peers in the system, the connection request is signed by the key server. This step needs to be performed only once, when a newly arriving subscriber authorizes itself to the key server in order to receive private keys for its credentials.

Secure overlay maintenance protocol

```
upon event Receive(CR of Snew from Sp)do
if decrypt_request (CR)==SUCCESS then
if degree(Sq) == available then
connect to the Snew
else
forward CR to {child peers and parent}-sp
if decrypt_request(CR)==FAIL then
if Sp==parent then
Try to swap by sending its own CR to the Snew.
else
forward to parent
```

IX.PERFORMANCEE VALUATIONS

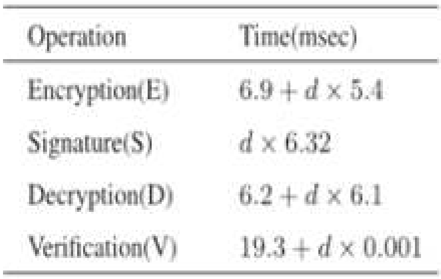
We evaluate three aspects of our system: 1) quantifying theoverhead of our cryptographic primitives, 2) benchmarking the performance of our secure pub/sub system, and3) analyzing attacks on subscription confidentiality. Here,we only discuss the first two aspects and the evaluationsrelated to the analysis of subscription confidentiality are available in the supplemental document available online.

Experimental Setup.Simulations are performed usingPeerSim [7]. Simulations are performed for up to N=2;048peers. Unless otherwise stated, out-degree constraintsof the peers are chosen aslog$_2$(N). The delays between thecommunication links are chosen in the range [24 and134 ms]. The complex subscriptions used during theevaluations contain conjunction of predicates defined on up to d=16different attributes. We evaluate the system performance under uniform (WL1) and skewed (WL2) subscription workloads, and with a uniform and skewedevent distribution. Skew is simulated using the widely used80-20 percent Zipfian distribution with three to five hot spots. The security mechanisms are implemented by the pairing-based cryptography library [14]. The implementation uses a 160-bit elliptic curve group based on the supersingular curvey2=x3+x over a 512-bit finite field.

Performance of Cryptographic Primitives

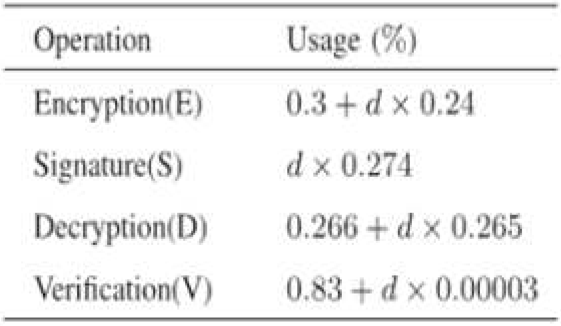
In this section, we measure the computational overhead of our security methods. All of our measurements were made on a 2-GHz Intel Centrino Duo with 2-GB RAM,running Ubuntu 9. Table 2 shows the throughput of the cryptographic primitives to perform encryption, decryp-tion, signature, and verification. All reporting values are averaged over 1,000 measurements. In our system, pairing-based encryption is used to encrypt a random key SK, which is later used to decrypt the actual event using symmetric encryption (cf. Section 5.3). Therefore, themessage size is kept 128 bytes as this key length is good enough for most symmetric encryption algorithms. Table 3shows the computational overhead (in msec) from the perspective of publishers and subscribers in our system. In general, the cost of verification is high due to the fact that it involves the computationally expensive pairing operations. Likewise, Table 4 shows the average CPU utilization for publishers and subscribers.

TABLE 3
Computation Times for Publishers and Subscribers



Operation	Time(msec)
Encryption(E)	$6.9 + d \times 5.4$
Signature(S)	$d \times 6.32$
Decryption(D)	$6.2 + d \times 6.1$
Verification(V)	$19.3 + d \times 0.001$

TABLE 4
Average CPU Utilization



Operation	Usage (%)
Encryption(E)	$0.3 + d \times 0.24$
Signature(S)	$d \times 0.274$
Decryption(D)	$0.266 + d \times 0.265$
Verification(V)	$0.83 + d \times 0.00003$

CONCLUSION

In this paper, we have presented a new approach to provide authentication and confidentiality in a broker-less content-based pub/sub system. The approach is highly scalable in terms of number of subscribers and publishers in the system and the number of keys maintained by them. In particular, we have developed mechanisms to assign credentials to publishers and subscribers according to their subscriptions and advertisements. Private keys assigned to publishers and subscribers, and the ciphertexts are labelled with credentials. We adapted techniques from identity-based encryption 1) to ensure that a particular subscriber can decrypt an event only if there is a match between the credentials associated with the event and its private keys and 2) to allow subscribers to verify the authenticity of received events. Furthermore, we developed a secure overlay maintenance protocol and proposed two event dissemination strategies to preserve the weak subscription confidentiality in the presence of semantic clustering of subscribers.we propose an identity-based conditional proxy re-encryption scheme which can provide a fine-grained level in the IBE setting. We formalize the definition and security of IBCPRE, and give a concrete scheme without random oracles. But we note that we do not give the formal security proof for our proposal, which is our future work.

REFERENCES

[1] J. Bacon, D.M. Eyers, J. Singh, and P.R. Pietzuch, "Access Controlin Publish/Subscribe Systems," Proc. Second ACM Int'l Conf.Distributed Event-Based Systems (DEBS),2008

[2] E. Anceaume, M. Gradinariu, A.K. Datta, G. Simon, and A.Virgillito, "A Semantic Overlay for Self- Peer-to-Peer Publish/Subscribe," Proc. 26th IEEE Int'l Conf. Distributed ComputingSystems (ICDCS),2006.

[3] M. Ion, G. Russello, and B. Crispo, "Supporting Publication and Subscription Confidentiality in Pub/Sub Networks,"Proc. SixthInt'l ICST Conf. Security and Privacy in Comm. Networks (Secur-eComm),2010.

[4] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "PublicKey Encryption with Keyword Search,"Proc. Int'l Conf. Theory andApplications of Cryptographic Techniques on Advances in Cryptology(EUROCRYPT),2004.

[5] D. Boneh and M.K. Franklin, "Identity-Based Encryption from theWeil Pairing,"Proc. Int'l Cryptology Conf. Advances in Cryptology,2001.

[6] S. Choi, G. Ghinita, and E. Bertino, "A Privacy-EnhancingContent-Based Publish/Subscribe System Using Scalar ProductPreserving Transformations,"Proc. 21st Int'l Conf. Database andExpert Systems Applications: Part I,2010.

[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-PolicyAttribute-Based Encryption,"Proc. IEEE Symp. Security andPrivacy,2007.

[8] Blaze, Matt, Gerrit Bleumer, and Martin Strauss. "Divertible protocols and atomic proxy cryptography." In Advances in Cryptology—EUROCRYPT'98, pp. 127-144. Springer Berlin Heidelberg, 1998.

[9]Ateniese, Giuseppe, Kevin Fu, Matthew Green, and Susan Hohenberger. "Improved proxy re-encryption schemes with applications to secure distributed storage." ACM Transactions on Information and System Security (TISSEC)9, no. 1 (2006):

[10] Canetti, Ran, and Susan Hohenberger. "Chosen-ciphertext secure proxy re-encryption." In Proceedings of the 14th ACM conference on Computer and communications security, pp. 185-194. ACM, 2007.