ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

Encountering Evidence of a Node with Proximity based Mobile Opportunistic Social Network

[1] Dr.S.Chakaravarthi, [2], D.Saranya, [3] T.Shanmuga Priya
[1] Assistant Professor-III, [2] (PG)Scholor, [3] (PG) Scholor
Masters of Engineering,
Department of Computer Science and Engineering,
Velammal Engineering College,Chennai, India

***Abstract*—The mobile communication are usually done by proximity based mobile opportunistic social network to communicate between the nodes. Usually the nodes are communicated with their real ID to the mobile opportunistic social network (MOSN).In current method, using FaceChange that can support both anonymizing real IDs among neighbour nodes and collecting real ID-based encountering information which also support the fine grained control over the information in encountering evidence. Only when the two nodes disconnect with each other, each node forwards an encrypted encountering evidence to the encountered node to enable encountering information collection. Here, we propose, Advanced extensions for sharing real IDs with the alias name or alias ID between mutually trusted nodes of trusted authority. The efficient encountering evidence collection are done by the exchanging encountering information between the nodes after communication are done. This exchange of information by encountering evidence shows the trustworthiness and validate the use by having the time, real ID, alias name which are in the encrypted form of cipher text with some advanced encryption algorithm and with signatures. The signature with the real ID based play the role for decryption which says about the trustworthiness and validate the user. This helps to verify the trustworthiness of the mutually trusted node of trusted authority and validity of the user nodes from malicious nodes.**

***Keywords* : Encountering evidence, mobile opportunistic social network, Tursted authority.**

I. INTRODUCTION

AS A special form of delay tolerant networks (DTNs) [1], mobile opportunistic social networks (MOSNs) [2], [3] have attracted much attention due to the increasing popularity of mobile devices, e.g., smartphones and tablets. In MOSNs, mobile devices carried by people communicate with each other directly without the support of infrastructures when they Manuscript received November 27, 2015; revised June 25, 2016 and September 6, 2016; accepted October 3, 2016; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor H. Zheng.

This work was supported in part by the U.S. NSF under Grant NSF-1404981, Grant IIS-1354123, and Grant CNS-1254006; in part by the IBM Faculty Award under Grant 5501145; in part by the Microsoft Research Faculty Fellowship under Grant 8300751; and in part by the Startup Fund of Southern Illinois University. An initial version of this paper was published in the proceedings of ICNP 2015 [46]. K. Chen is with the Department of Electrical and Computer Engineering, Southern Illinois University, Carbondale, IL 62901 USA (e-mail: kchen@siu.edu). H. Shen is with the Department of Computer Science, University of Virginia, Charlottesville, VA 22904 USA (e-mail: hs6ms@virginia.edu). Digital Object Identifier 10.1109/TNET.2016.2623521 Fig. 1. Demonstration of a privacy issue and a possible solution in MOSNs. (a) Possible privacy issue. (b) Solution: neighbor Anonymity. meet (i.e., within the communication range of each other) opportunistically.

Such a communication model can be utilized to support various applications without infrastructures, such as packet routing between mobile nodes [4], encountering based social community/relationship detection [5], [6], and distributed file sharing and Question & Answer (Q&A) [7]–[9] in a community. In each system, a node is uniquely labeled by an unchanging ID (defined real ID), which is obtained from the trust authority (TA), for the corresponding service. Since those services are built upon node encountering, nodes need to collect real ID based encountering information. For example, nodes need to know whom they have met to identify proximity based social community/relationships.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

In packet routing, nodes need to collect the encountering information to deduce their future meeting probabilities with others. Then, a packet can always be forwarded to the appropriate forwarder.

II. MOTIVATION

In current MOSN applications, nodes can collect real ID based encountering information easily since neighbor nodes communicate with real IDs directly. We define two nodes as neighbor nodes when they are within the communication range of each other. However, when using real IDs directly, the disclosure of node ID to neighbor nodes would create privacy and security concerns. For example, a malicious node can first know the IDs of some central nodes or nodes with specific interests. Then, as shown in Figure 1(a), when neighbor nodes communicate with real IDs, a malicious node can easily identify attack targets from neighbors and launch attacks to degrade the system performance or steal important documents. Further, without protection, malicious nodes can also easily sense the encountering between nodes for attacks. Therefore, neighbor node anonymity is needed to prevent the disclosure of real IDs to neighbors. Clearly, a permanent pseudonym cannot achieve such a goal since it can be linked to a node, which can still enable malicious nodes to recognize targets from neighbor nodes.

Thus, an intuitive method to realize the neighbor node anonymity is to let each node continuously change its pseudonym used in the communication with neighbors, as shown in Figure 1(b). However, when neighbor node anonymity is enforced, nodes cannot collect the real ID based encountering information (i.e., cannot know whom they have met), which disables aforementioned MOSN services. Consequently, there is a challenge on anonymizing neighbor nodes for privacy protection and meanwhile still supporting encountering information collection in MOSNs. There are rich investigations on protecting node privacy in MOSNs [10]–[17]. However, most of related works [10]–[16] focus on anonymizing interests and profiles and are not designed for neighbor node anonymity, which is a feature provided in this paper. The work in [17] supports neighbor node anonymity but fails to provide encountering information collection at the same time. Therefore, we propose FaceChange to realize both aforementioned goals based on a key observation in MOSNs. That is, disconnected nodes cannot communicate with each other directly in MOSNs, which makes attacking disconnected nodes almost impossible.

This also means that knowing real IDs after the encountering would not compromise the privacy protection. Thus, the proposed FaceChange keeps node anonymity only during the encountering and postpone the real ID based encountering information collection to a moment after two neighbor nodes disconnect with each other. Figure 2 illustrates the design of FaceChange. When two nodes meet, they communicate anonymously. However, each of them creates an encountering evidence that contains their real IDs. The encountering evidences are sent to the other node only when they separate, thus enabling the encountering information collection while keeping the anonymity during the encountering. For an encountering evidence, we call the node that creates it as the creator and the encountered node that is to receive it as the recipient

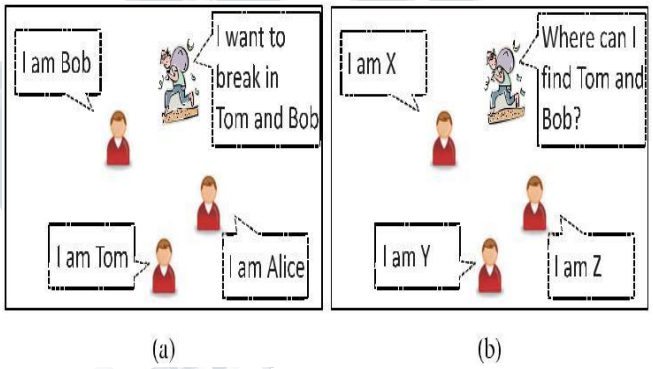


Fig. 1.Demonstration of a privacy issue and a possible solution in MOSNs.
(a) Possible privacy issue. (b) Solution: neighbor Anonymity.

III. SECURITY CHALLENGES

FaceChange needs to handle the following challenges for encountering information collection. (1) The security of the encountering evidence needs to be ensured. An encountering evidence can only be accessed by its creator and recipient and cannot be forged. (2) An encountering evidence needs to be successful delivered to its recipient even when the real ID of the recipient node is unknown due to neighbor node anonymity. (3) When creating an encountering evidence, a node can control what contents (e.g., basic encountering information and application information) to be included based on its trust on the encountering node. The calculation of the trust should be privacy-preserving. FaceChange incorporates the following schemes to handle the three challenges.

.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

3.1 ENCOUNTER EVIDENCE

Encountering Evidence Encryption and Validation Scheme: For each encountering evidence, FaceChange uses the bilinear pairing technique [18] to generate an encryption key and a pair of uniquely matched token and commitment with efforts from both encountering nodes. The property of the bilinear pairing ensures that nodes other than the creator and recipient, even eavesdroppers, cannot know the key. Further, the token is attached to the evidence and the commitment is stored on the recipient node for validation, thereby ensuring the uniqueness of each encountering evidence.

3.1.1 Encountering Evidence Relaying Scheme:

In this scheme, during the encountering, the recipient node specifies a relay node and encrypts its real ID with the public key of the relay node. It then forwards such information to the creator. Later, after the two nodes separate, the creator routes the encountering evidence to the relay node, which decrypts the ID of the recipient node and further routes the evidence to the recipient node, thereby delivering the encountering evidence. Encountering Evidence Generation Scheme: More similar attributes (e.g., affiliation and reputation) between two nodes often denote higher trust between them [12].Thus, we realize the control on the contents in an encountering evidence based on the attribute similarity. We use the commutative encryption [19] and the solution for "the millionaire's problem" [20] to calculate the attribute similarity blindly in this process, which protects node privacy. With neighbor anonymity, a node may fail to recognize the destinations of its packets even when meeting them, thereby making it hard to deliver packets. We solve this problem by letting nodes pretend to be a better forwarder for packets destined for them to fetch these packets (Section IV-G). As a result, packet routing can be conducted correctly and efficiently in FaceChange. This shows that MOSN services can be supported when FaceChange is adopted. We further design two advanced extensions to enhance the practicability of FaceChange. The first one enables mutually trusted nodes to disclose real IDs to each other during the encountering, and the second one enhances the routing effi- ciency of the encountering evidence relaying. In summary, the major contribution of this paper is to propose a novel design that supports both neighbor node anonymity and real ID based encountering information collection in MOSNs. FaceChange prevents two encountering nodes from disclosing the real IDs during the encountering, so malicious nodes cannot identify targets from neighbors for attack.

3.1.2 Encountering Evidence Relaying Scheme

In this scheme, during the encountering, the recipient node specifies a relay node and encrypts its real ID with the public key of the relay node. It then forwards such information to the creator. Later, after the two nodes separate, the creator routes the encountering evidence to the relay node, which decrypts the ID of the recipient node and further routes the evidence to the recipient node, thereby delivering the encountering evidence. Encountering Evidence Generation Scheme: More similar attributes (e.g., affiliation and reputation) between two nodes often denote higher trust between them [12]. Thus, we realize the control on the contents in an encountering evidence based on the attribute similarity. We use the commutative encryption [19] and the solution for "the millionaire's problem" [20] to calculate the attribute similarity blindly in this process, which protects node privacy. With neighbor anonymity, a node may fail to recognize the destinations of its packets even when meeting them, thereby making it hard to deliver packets

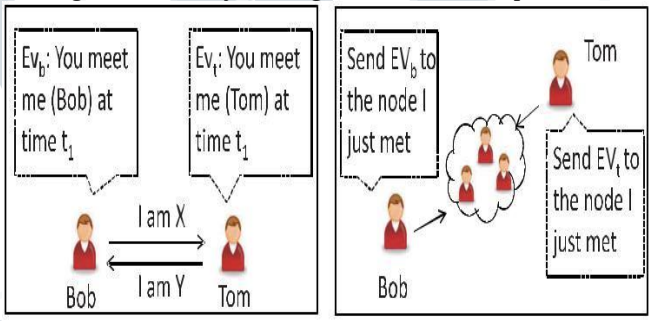


Fig. 2. General solution for encountering record collection. (a) Create the encountering evidence under neighbor node anonymity. (b) Route the encountering evidence to the other node after separation.

3.2 NODES ACTIVITY ON ENCOUNTERING INFORMATION

When nodes move away from each other, they rely on the encountering evidence to know the real IDs of nodes they have met to support MOSN services. This is acceptable since in MOSNs, a malicious node cannot communicate with a disconnected node for attacks. In the following, Section II introduces related work. Section III presents the preliminary background. Sections IV and V introduce the design of FaceChange and two advanced extensions, respectively. Section VI evaluates FaceChange through trace-driven and smartphone-based experiments. Section VII concludes this paper with future

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

work.

IV. RELATED WORK

4.1 Impact of Human Mobility on Opportunistic Forwarding Algorithms:

We study data transfer opportunities between wireless devices carried by humans. We observe that the distribution of the inter contact time (the time gap separating two contacts between the same pair of devices) may be well approximated by a power law over the range [10 minutes; 1 day]. This observation is confirmed using eight distinct experimental data sets. It is at odds with the exponential decay implied by the most commonly used mobility models. In this paper, we study how this newly uncovered characteristic of human mobility impacts one class of forwarding algorithms previously proposed. We use a simplified model based on the renewal theory to study how the parameters of the distribution impact the performance in terms of the delivery delay of these algorithms. We make recommendations for the design of well-founded opportunistic forwarding algorithms in the context of human carried devices.

4.2 SMART: Lightweight Distributed Social Map Based Routing in Delay Tolerant Networks:

Previous Delay Tolerant Network (DTN) routing algorithms exploit either past encounter records (probabilistic routing) or social network properties (social network based routing) to derive a node's probability of delivering packets to their destinations. However, they only have a local view of the network, which limits the routing efficiency. Also, when two nodes meet, they have to exchange the delivery probabilities to the destinations of all packets in the two nodes, which incurs high resource consumption. In a social network, the people a person frequently meets are usually stable, which makes them play a more important role in forwarding message for the person. Based on this, we propose a lightweight distributed Social MAP based Routing algorithm in delay Tolerant networks (SMART). In SMART, each node builds its own social map consisting of nodes it has met and their frequently encountered nodes in a distributed manner. Based on both encountering frequency and social closeness of the two linked nodes in the social map, we decide the weight of each link to reflect the packet delivery probability between the two nodes. The social map enables more accurate forwarder selection through a broader view. Moreover, nodes exchange much less information for social map update and need fewer updates due to social map stability, which reduces resource consumption. Trace-driven experiments and tests on the GENI ORBIT testbed demonstrate the high efficiency of SMART in comparison with previous algorithms.

4.3 Dynamic Social Feature-based Diffusion in Mobile Social Networks:

With the wide use of smart mobile devices and the popularity of mobile social networks (MSNs), direct marketing has been adopted by more and more companies to announce the news of their products first to a group of selected profitable customers and let them diffuse the news by "word-of-mouth" to other potential buyers to control the marketing cost. In this paper, we study the diffusion minimization problem whose goal is to select an optimal set of initial nodes to disseminate the information to the whole network as quickly as possible. We tackle the problem by taking advantage of node social features in MSNs. We define dynamic social features to capture nodes' dynamic contact behavior and use social similarity metrics to measure their social closeness. We adopt the community concept in social networks to reduce the complexity of the diffusion minimization problem. We propose novel diffusion node selection algorithms based on these new features to minimize the diffusion time. Simulation results show that our algorithms have lower diffusion times than the existing ones.

4.4 Safety Challenges and Solutions in Mobile Social Networks:

Mobile social networks (MSNs) are specific types of social media which consolidate the ability of omnipresent connection for mobile users/devices to share user-centric data objects among interested users. Taking advantage of the characteristics of both social networks and opportunistic networks (OppNets), MSNs are capable of providing an efficient and effective mobile environment for users to access, share, and distribute data. However, the lack of a protective infrastructure in these networks has turned them into convenient targets for various perils. This is the main impulse why MSNs carry disparate and intricate safety concerns and embrace divergent safety challenging problems. In this paper, we aim to provide a clear categorization on safety challenges and a deep exploration over some recent solutions in MSNs. This work narrows the safety challenges and solution techniques down from OppNets and delay-tolerant networks to MSNs with the hope of covering all the work proposed around security, privacy, and trust in MSNs. To conclude, several major open research issues are discussed, and future research directions are outlined.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

4.5 ***An Efficient and Secure ID Based Group Signature Scheme from Bilinear Pairings:***

An efficient and secure identity based group signature scheme from bilinear pairings. Group signature allows group member to sign arbitrary number of messages on behalf of the group without revealing their identity. Under certain circumstances the group manger holding a tracing key can reveal the identities of the signer from the signature. Our scheme is based on the Computation Diffie-Hellman Problem (CDHP) assumption and bilinear pairings. In the scheme, the size of the group public key and length of the signature are independent on the numbers of the group members

***4.6 A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*:**

An encryption method is presented with the novel property that publicly revealing an encryption key does not thereby reveal the corresponding decryption key. This has two important consequences: 1. Couriers or other secure means are not needed to transmit keys, since a message can be enciphered using an encryption key publicly revealed by the intended recipient. Only he can decipher the message, since only he knows the corresponding decryption key. 2. A message can be "signed" using a privately held decryption key. Anyone can verify this signature using the corresponding publicly revealed encryption key. Signatures cannot be forged, and a signer cannot later deny the validity of his signature. This has obvious applications in "electronic mail" and "electronic funds transfer" systems. A message is encrypted by representing it as a number M, raising M to a publicly specified power e, and then taking the remainder when the result is divided by the publicly specified product, n, of two large secret prime numbers p and q. Decryption is similar; only a different, secret, power d is used, where $e \cdot d \equiv 1 \pmod{(p-1)\cdot(q-1)}$. The security of the system rests in part on the difficulty of factoring the published divisor, n. Key Words and Phrases: digital signatures, public-key cryptosystems, privacy, authentication, security, factorization, prime number, electronic mail, message-passing, electronic funds transfer, cryptography.

4.*7 DTN Routing as a Resource Allocation Problem*:

Many DTN routing protocols use a variety of mechanisms, including discovering the meeting probabilities among nodes, packet replication, and network coding. The primary focus of these mechanisms is to increase the likelihood of finding a path with limited information, so these approaches have only an incidental effect on routing metrics such as maximum or average delivery delay. In this paper, we present rapid, an intentional DTN routing protocol that can optimize a specific routing metric such as worst-case delivery delay or the fraction of packets that are delivered within a deadline. The key insight is to treat DTN routing as a resource allocation problem that translates the routing metric into per-packet utilities which determine how packets should be replicated in the system.

We evaluate rapid rigorously through a prototype deployed over a vehicular DTN testbed of 40 buses and simulations based on real traces. To our knowledge, this is the first paper to report on a routing protocol deployed on a real DTN at this scale. Our results suggest that rapid significantly outperforms existing routing protocols for several metrics. We also show empirically that for small loads RAPID is within 10% of the optimal performance

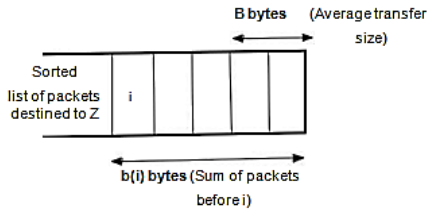
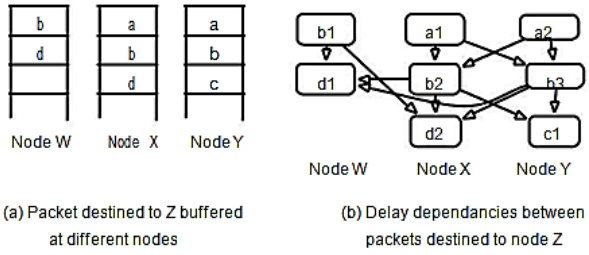


Figure 1: Position of packet i in a queue of packets destined to Z.



(a) Packet destined to Z buffered at different nodes

(b) Delay dependancies between packets destined to node Z

4.*8 Routing in a Delay Tolerant Network*:

The delay-tolerant networking routing problem, where messages are to be moved end-to-end across a connectivity graph that is time-varying but whose dynamics may be known in advance. The problem has the added constraints of finite buffers at each node and the general property that no contemporaneous end-to-end path may ever exist. This situation limits the applicability of traditional routing approaches that tend to treat outages as failures and seek to find an existing end-to-end path.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

We propose a framework for evaluating routing algorithms in such environments. We then develop several algorithms and use simulations to compare their performance with respect to the amount of knowledge they require about network topology. We find that, as expected, the algorithms using the least knowledge tend to perform poorly. We also find that with limited additional knowledge, far less than complete global knowledge, efficient algorithms can be constructed for routing in such environments. To the best of our knowledge this is the first such investigation of routing issues in DTNs.

4.9 Probabilistic Routing in Intermittently Connected Networks:

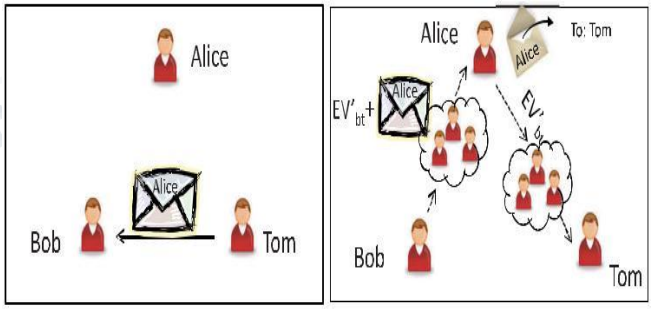
Vahdat and Becker present a protocol for epidemic routing in intermittently connected networks [8]. When two nodes encounter each other, they exchange messages being carried (subject to buffer space), thus causing the messages to spread through the network like an epidemic of a disease. This approach ensures that a message reaches its destination as soon as possible, but it also wastes a lot of resources through unnecessary message transfers. Due to the ambiguity in deciding what the best next hop is the networks discussed here, Chen and Murphy propose that applications should be able to affect that through the introduction of a utility function [3], allowing applications to specify weights of several factors influencing the function. In the proposed solution, called Disconnected Transitive Communication, a discovery protocol is used to find the best next hop within the cluster of currently connected nodes. Grossglauser and Tse approach this kind of routing from a slightly different point of view [5]. One major problem with ad hoc networks is that due to interference of concurrent transmissions between nodes they scale badly. By only doing local communications between neighbors and instead relying on the movement of nodes to bring a message to its destination, it is shown that this problem can be mitigated.

4.10 A Complex Network Analysis of Human Mobility:

—Opportunistic networks use human mobility and consequent wireless contacts between mobile devices, to disseminate data in a peer-to-peer manner. To grasp the potential and limitations of such networks, as well as to design appropriate algorithms and protocols, it is key to understand the statistics of contacts. To date, contact analysis has mainly focused on statistics such as inter-contact and contact distributions. While these pair-wise properties are important, we argue that structural properties of contacts need more thorough analysis. For example, communities of tightly connected nodes, have a great impact on the performance of opportunistic networks and the design of algorithms and protocols.

we propose a methodology to represent a mobility scenario (i.e., measured contacts) as a weighted contact graph, where tie strength represents how long and often a pair of nodes is in contact. This allows us to analyze the structure of a scenario using tools from complex network analysis and graph theory (e.g., community detection, connectivity metrics). We consider four mobility scenarios of different origins and sizes. Across all scenarios, we find that mobility shows typical smallworld characteristics (short path lengths, and high clustering coefficient). Using state-of-the-art community detection, we also find that mobility is strongly modular. However, communities are not homogenous entities. Instead, the distribution of weights and degrees within a community is similar to the global distribution of weights, implying a rather intricate intra-community structure.

To the best of our knowledge, this is the most comprehensive study of structural characteristics of wireless contacts, in terms of the number of nodes in our datasets, and the variety of metrics we consider. Finally, we discuss the primary importance of our findings for mobility modeling and especially for the design of opportunistic network solutions.



(a) (b)

Fig. 3. Relaying encountering evidence to the recipient. (a) Select the relay node. (b) Relay to the recipient.

V. TECHQUIE USED

The implementation of encountering evidence of information is done bye using maily three algorithm

(i) RSA algorithm.
(ii) H-mac algorithm.
(iii) ID- based signature algorithm.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

VI. GROUP SIGNATURE

In this section we introduce the definition and security properties of group signatures [14].

Definition- A group signature scheme is a digital signature scheme consisted of the following four procedures:

- Setup: On input a security k, the probabilistic algorithm outputs the initial group public key Y and the secret key s of the group manager.
- Join: A protocol between the group manager and a user that result in the user becoming a new group member. The user's output is a membership certificate and a membership secret.
- Sign: A probabilistic algorithm that an input a group public key, a membership certificate, a membership secret and a message m. Output is the group signature of m.
- Verify: An algorithm takes as input the group public key Y, the signature, the message m to output 1 or 0.
- Open: The deterministic algorithm takes as input the messages m, the signature, the group manager's secret key s to return " Identity or failure"

A secure group signature must satisfy the following properties:

- Correctness: Signature produced by a group member using Sign must be accepted by Verify.
- Unforgeability: Only the group members can sign messages on behalf of the group.
- Anonymity: Given a valid signature, it is computationally hard to identify the signer for any-one except the group manager.
- Unlinkability: Deciding whether two different valid signatures were computed by the same group member is computationally hard for anyone except the group manager. • Traceability: The group manager is always able to open a valid signature & identify the signer.
- Exculpability: Neither the group manager nor a group member can sign messages on behalf of other group members. Also, the group manager or colludes with some group members can misattribute a valid group signature to frame a certain members.
- Coalition-resistance: A colluding subset of group members (even if comprised of the whole group) cannot produce a valid signature that the group manager cannot open.
- Efficiency: The efficiency of group signature is based on the parameters: the size of the group public key, the length of the group signature and the efficiency of the algorithms and protocols of the group signatures

III.

6.1 Proposed ID based group signature scheme from bilinear pairings:

ID-based group signature scheme from bilinear pairing. We only need to consider that Key generation centre (KGC) is the group manager. We can't adopt the usual ID-based system. Since key escrow is fatal drawback for traditional ID-based system. So it assumed that that KGC must be trusted unconditionally. Otherwise, the system will be collapsed. If KGC act as the group manager of a group, he can forge the signature of any users. Therefore, the most important thing to design an ID-based group signature scheme is to solve the problem of key-escrow. Proposed scheme consists of six procedures: Set-up, Extract, Join, Sign, Verify, and Open. In our scheme, KGC is assumed no longer to be a trusted party.

6.1.1 Preliminary Works:

Let G_1 be a cyclic additive group generated by P, whose order is prime q, and G_2 be a cyclic multiplicative group of the same order q. Let a, b be elements of Z^*q . We assume that the DLP in both G_1 and G_2 are hard. A bilinear pairing is a map $e : G_1 \times G_1 \rightarrow G_2$ with the following properties:

I. **Bilinear:** $e(aP,bQ) = (P,Q)^{ab}$

[1] **Non-degenerate:** There exist $P,Q \in G_1$ such that $e(P,Q) \neq 1$

Computable: There is an efficient algorithm to compute e(P, Q) for all $P , Q \in G_1$

Let G1 be a Gap DH cyclic additive group generated by P, whose order is prime order q and G2 be a cyclic multiplicative group of same order q. A bilinear pairing is a map 1 1 2 e :G ×G →G .

Define Cryptographic hash function H1 { }1,0: ×G1 →G1 * , H { } Z q * → * 1,0: 2 and H G Z q → * 3 1 :

Let G_1 be a Gap DH cyclic additive group generated by P, whose order is prime order q and G_2 be a cyclic multiplicative group of same order q. A bilinear pairing is a map $e : G_1 \times G_1 \rightarrow G_2$. Define Cryptographic hash function $H_1 : \{0,1\}^* \times G \rightarrow G$, $H_2 : \{0,1\}^* \rightarrow Z^*q$ and $H_3 : G_1 \rightarrow Z^*_q$

- **Setup:** KGC chooses a generator P of G_1 and picks a random number $s \in Z^*q$ and set $P_{pub} = sP$. Thus system Parameters are

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

$\{G_1, G_2, e, q, P, P_{pub}, H_1, H_2, H_3\}$ and keep s as the master secret key, which is known only him-self.

Extract:

A user U_i submit his (or her) identity information ID_i and rP to KGC, where $r \in Z^*q$ long-term private key. Then KGC computes the user's public key

$Q_{IDi} = H_1 (ID_i \| T, rP)$.

Here T is life Span of r and sends $S_{IDi} = sQ_{IDi}$ to the user via a **secure channel.**

Thus user's private key pair is (r, S_{IDi}). The user should update his key pair after the

Secret key, since KGC is no longer trustful , it may expose them to other members.

Join: Suppose that a user U_i wants to join the group. For this, he and KGC perform Join protocol as follows:

The user U_i chooses a random number $x_i \in Z^*q$, then sends

$\{rx_i, rP, ID_i, x_i P\}$ to KGC

If KGC is convinced that the user know $S_{IDi} = sH_1 (ID_i \| T, rP)$ and

$e(rx_i P, P) = e(x_i, rP)$, KGC sends secretly

$s_i = sH_1 (ID_i, rx_i P)$ to the user U_i

Thus user's member certificates are $(s_i, rx_i P)$ and his private signing key is rx_i . KGC adds $(rx_i P, x_i P, rP, ID_i)$ to the member list.

Sign: To sign a message m, the user U_i randomly chooses number $\alpha, \beta, k \in Z^*q$ and uses her signing key and corresponding member certificate and then computes the following values:

$R = k P$..... (1)

$S_1 = rx_i Q_{IDi}$..... (2)

$S_2 = rx_i \alpha H_1 (ID_i, rx_i P) + H_3 (R) P$ ..... (3)

$S_3 = rx_i \beta H_1 (ID_i, rx_i P) + H_2 (m) P$ ....(4)

$S_4 = [H_2 (m)\alpha + \beta H_3 (R)] s_i$ .....(5)

Thus ID-based group signature on the message m is

$(R, S_1, S_2, S_3, S_4, rx_i P)$

• **Verify:** To verify a group signature $(R, S_1, S_2, S_3, S_4, rx_i P)$ on the message m. Verifier accept the signature if following equation holds

$e(S_4, rx_i P) = e(S_2, P_{pub})^{H_2 (m)}$

$e(S_3, P_{pub})^{H_3 (R)}$.....(6)

$e(S_1, P) = e(Q_{IDi}, rx_i P)$.....(7)

If it is true then $[R, S_1, S_2, S_3, S_4, rx_i P]$ is valid ID based group signature on the message m

Open: In case of dispute, the KGC can easily identify the user. The signer can't deny the signature after KGC present a proof. KGC check the following equation:

$e(S_{IDi}, P) = e(H_1 (ID_i \| T, rP), P_{pub})$

$e(s_i, P) = e(H_1 (ID_i, rx_i P), P_{pub})$

$e(S_1, P_{pub}) = e(S_{IDi}, rx_i P)$

$e(S_2, S_{IDi})^{H_2 (m)} e(S_3, S_{IDi})^{H_3 (R)} = e(S_4, S_1)$ .....(8)

6.2 Our Encryption and Decryption Methods

To encrypt a message M with our method, using a public encryption key (e, n), proceed as follows. (Here e and n are a pair of positive integers.) First, represent the message as an integer between 0 and n − 1. (Break a long message into a series of blocks, and represent each block as such an integer.) Use any standard representation. The purpose here is not to encrypt the message but only to get it into the numeric form necessary for encryption. Then, encrypt the message by raising it to the eth power modulo n. That is, the result (the ciphertext C) is the remainder when Me is divided by n. To decrypt the ciphertext, raise it to another power d, again modulo n.

The encryption and decryption algorithms E and D are thus: C ≡ E(M) ≡ Me (mod n), for a message M . D(C) ≡ C d (mod n), for a ciphertext C .

The encryption key is thus the pair of positive integers (e, n). Similarly, the decryption key is the pair of positive integers (d, n). Each user makes his encryption key public, and keeps the corresponding decryption key private. (These integers should properly be subscripted as in nA, eA, and dA, since each user has his own set. However, we will only consider a typical set, and will omit the subscripts.)

How should you choose your encryption and decryption keys, if you want to use our method? You

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

first compute n as the product of two primes p and q: n = p · q . These primes are very large, "random" primes. Although you will make n public, the factors p and q will be effectively hidden from everyone else due to the enormous difficulty of factoring n. This also hides the way d can be derived from e. You then pick the integer d to be a large, random integer which is relatively prime to (p − 1)·(q − 1). That is, check that d satisfies:

gcd(d,(p − 1) · (q − 1)) = 1 ("gcd" means "greatest common divisor").

The integer e is finally computed from p, q, and d to be the *"multiplicative inverse"* of d, modulo (p − 1) · (q − 1). Thus we have

$e \cdot d \equiv 1 \pmod{(p-1)\cdot(q-1)}$.

We prove in the next section that this guarantees that (1) and (2) hold, i.e. that E and D are inverse permutations. Section VII shows how each of the above operations can be done efficiently. The aforementioned method should not be confused with the "exponentiation" technique presented by Diffie and Hellman [1] to solve the key distribution problem.

Their technique permits two users to determine a key in common to be used in a normal cryptographic system. It is not based on a trap-door one-way permutation. Pohlig and Hellman [8] study a scheme related to ours, where exponentiation is done modulo a prime number.

VII. TECHNOLOGY USED IN CURRENT SCENARIO

In current methods, when nodes meet, they simply communicate with their real IDs, which leads to privacy and security concerns. In current MOSN applications, nodes can collect real ID based encountering information easily since neighbor nodes communicate with real IDs directly.

However, when using real IDs directly, the disclosure of node ID to neighbor nodes would create privacy and security concerns. For example, a malicious node can first know the IDs of nodes with specific interests when neighbor nodes communicate with real IDs, a malicious node can easily identify attack targets from neighbors and launch attacks to degrade the system performance or steal important documents. Further, without protection, malicious nodes can also easily sense the encountering between nodes for attacks.

VIII. PERFORMANCE OF NODES

we propose FaceChange, a system that supports both neighbor anonymity and real ID based encountering information collection in MOSNs. In FaceChange, each node continually changes its pseudonyms and parameters when communicating with neighbors nodes to hide its real ID.

Initially, the Trusted Authority will be generate the public and private key to all the user or node, along with the key's Trusted Authority will generate the signature of an node by their private key, and send it to the corresponding nodes . Each node in the network has their own set of public , private key and signature .Whenever node need to communicate with the neighbor node they usually communicate with their alias name or alias id . Node send the request to the other node before accept the request node first verify the signature that are generated by the TA. Only when the signature match user will decide to accept the request or not if signature mismatch node will drop the request .

After communication process between node . nodes will exchange their Envelop,(Envelop will contain the user real id.) before send envelop to other node each node will encrypt the content and sign the envelop, and send the relay node to whom envelop should be redirect.

When relay node receive the envelop check the sign in the envelop to verify the content is not corrupted and decrypt the content and directed to the node in the envelop.

IX. ACTIVITY

1. Network Formation & Neighbor Calculation
2. Data communication
3. Evidence forward

9.1 Network Formation:

First we can create a Trusted Authority and then create network node assume the communication range of a node is finite. By providing distance and range ie Coverage of an particular node. Node in the network would contain unique real ID and port number for communicate with other node. Node generate it own alias name or alias id for security purpose. Node can send the data to the other node directly when the destination node is willing to communicate with that node then only the node can send data or chat with the node .

Node need to find their nearby neighbor before starting any communication. Neighbor is calculated based on the coverage of each node , when the node comes the coverage range of the other node then the two node will consider as the neighbor node . neighbor node also know by alias name only.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 8, August 2017

9.2 Data communication:

Once the node enter in the network node will generate its alias name or alias id each node in the network would hide their real Id and shown only the alias name to all the near by node , Only the trusted authority know the real id as well as the alias name of an each node and trusted authority would maintain evidence of data communication between nodes.

Node first send the chat request to any one of the neighbor node when the destination node accept the request the both the will chat with each other Chatting or communication between two node is provided by the alias name only. Once the disconnect with each other node the two node will exchange the envelop with each other.

9.3 Evidence forward:

In each system, a node is uniquely labeled by an unchanging ID (defined real ID), which is obtained from the trust authority (TA), for the corresponding service. Since those services are built upon node encountering, nodes need to collect real ID based encountering information. the design of FaceChange. When two nodes meet, they communicate anonymously. However, each of them creates an encountering evidence that contains their real IDs.

when the destination node accept the request the both the will chat with each other Chatting or communication between two node is provided by the alias name only. Once the disconnect with each other node the two node will exchange the envelop with each other.

The envelop contain the real id of the particular node, each node need to forward envelop to the relay node that provided by trusted authority and maintain the evidence.

X. CONCLUSION

- The communication between the neighboring nodes with the changes in pseudonyms and parameters to hide its real ID from the malicious node of attacker provide the security and integrity.
- The evidence of envelope which has the time, real ID, alias name provides the validity of the user nodes.
- The encountering evidence of information shared between the neighboring nodes after the communication provides the trustworthiness of the mutually trusted authority.
- This exchange of encountering evidence of information provide the surety about both the neighboring nodes of the user and trusted authority.

REFERENCE

[1] S. Jain, K. Fall, and R. Patra, “Routing in a delay tolerant network,” in Proc. SIGCOMM, 2004, pp. 145–158.

[2] J. Wu, M. Xiao, and L. Huang, “Homing spread: Community home-based multi-copy routing in mobile social networks,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 2319–2327.

[3] T. Ning, Z. Yang, H. Wu, and Z. Han, “Self-interest-driven incentives for ad dissemination in autonomous mobile social networks,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 2310–2318.

[4] A. Balasubramanian, B. Levine, and A. Venkataramani, “DTN routing as a resource allocation problem,” in Proc. SIGCOMM, 2007, pp. 373–384.

[5] P. Hui, E. Yoneki, S. Y. Chan, and J. Crowcroft, “Distributed community detection in delay tolerant networks,” in Proc. MobiArch, 2007, Art. no. 7,