

Leveraging Social Contacts for Message Confidentiality in Delay Tolerant Networks

Karim El Defrawy, John Solis, Gene Tsudik
Donald Bren School of Information and Computer Science
University of California, Irvine
Irvine, CA USA
keldefra@uci.edu, jsolis@uci.edu, gts@ics.uci.edu

Abstract—Delay- and disruption-tolerant networks (DTNs) can bring much-needed connectivity to rural areas and other settings with limited or non-existing infrastructures. High node mobility and infrequent connectivity inherent to DTNs make it challenging to implement simple and traditional security services, e.g., message integrity and confidentiality.

In this paper, we focus on the problem of initial secure context establishment in DTNs. Concretely, we design a scheme that allows users to leverage social contact information to exchange confidential and authentic messages. We then evaluate the proposed scheme by analyzing real-world social network data, simulating communication scenarios, and through an informal security analysis.

Keywords—Delay Tolerant Networks; Message Confidentiality; Secure Messaging

I. INTRODUCTION

Delay- and disruption-tolerant networks (DTNs) are characterized by highly mobile nodes, intermittent connectivity and frequent disruptions. Disruptions may occur because of limited wireless communication range, sparsity of nodes, attacks and general noise. At the same time, DTNs represent an attractive solution for low-cost networking in rural areas and developing countries with no or poor communication infrastructure. Several prototypes are in use and have been reported on in the literature [1]–[3].

Consider a sample DTN scenario, as shown in Figure 1. A user located in a rural area wishes to send a message but cannot because the communication infrastructure does not exist or is too expensive to use (e.g., GSM). However, suppose a bus runs through the villages on a regular basis and is equipped with a wireless access point. As the bus travels along its route it transfers content to other passengers, other buses, and to fixed infrastructure nodes.

Upon arriving to the central station the remaining content is offloaded to a fixed node(s) connected to the Internet. Data is delivered directly to the destination if it has Internet access; otherwise, it travels to the fixed point closest to the destination and is delivered over the local DTN.

As illustrated by this scenario, DTNs provide inexpensive communication in the absence of a communication infrastructure. These networks can also be appropriate when operating on a temporary basis and investment in costly fixed infrastructure is unjustified. Furthermore, DTNs are well-suited for long-haul communication settings, such as space

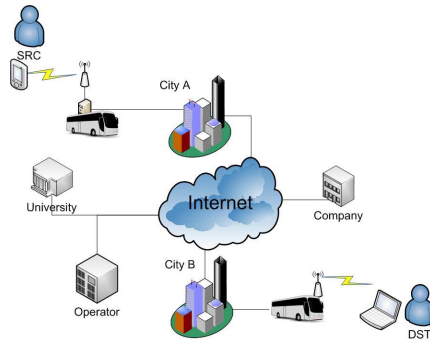


Figure 1. Sample DTN usage scenario

and inter-planetary networks [4]–[6]. In this setting distances (and delays) are very long, error rates are potentially high, and rotation of celestial bodies (and nodes themselves) can periodically inhibit communication.

While useful in many scenarios, we acknowledge that DTNs will not be a primary mode of communication in well-connected areas. We envision DTNs being deployed in remote areas as a cheaper alternative to the high-speed networking infrastructure or as a fall-back option in cases of infrastructure failure. In this paper, we focus on terrestrial DTNs characterized by the sample usage scenario above.

The sporadic connectivity of DTNs raises a number of security-related challenges due to frequent partitioning and a lack of reliable complete source-to-destination paths. One such challenge is *initial secure context establishment* since it is unrealistic to assume the existence of a global and always-available public key infrastructure (PKI).¹

To address this challenge, we suggest a simple technique for achieving secure communication in DTNs. It leverages casual information, such as knowledge of current and previous affiliations as well as social contacts of peers, in order to establish an initial security context between DTN nodes with no security “history”. We evaluate the proposed technique by analyzing real-world social network data and simulating intra- and inter-region communication scenarios. We also provide an informal security analysis.

¹Even if available we cannot assume that nodes can retrieve the public key of peers or that certificate queries are returned in a timely fashion.

II. MOTIVATION AND PROBLEM FORMULATION

Several factors motivate distinct security techniques for DTNs. Traditional protocols for establishing secure connections (e.g., SSL/TLS) require multiple rounds to exchange credentials and then agree on a set of cryptographic algorithms and security parameters. In a DTN, the round-trip delay may be excessively long (e.g., minutes or hours) and the end-to-end path might be unpredictable, since it depends on mobility patterns. Thus, messages can be lost or delivered out-of-order. However, security protocols typically require strict message ordering. Also, coping with the delay of several protocol rounds before being able to encrypt data is not viable due to the opportunistic and sporadic message transmission characteristics of DTNs. Clearly, establishing a secure context for a single message is both awkward and expensive.

Some prior works have suggested employing Identity-Based Encryption (IBE) [7], [8] to let a source derive the destination public key from some associated identity string, e.g., an e-mail address. However, IBE presents certain difficulties. Consider revocation: to revoke a public key, a compromised node's identity must be changed. This is not always feasible, e.g., changing one's e-mail address is a very cumbersome process. One way to side-step this problem is to use fine-grained identities, e.g., by including time-stamps in identity strings. The problem is thus shifted to the receiver who must often – based on the granularity of the time-stamp – contact its private-key generator (PKG) and obtain a new private key. However, this is problematic since DTNs are sporadically connected networks and constant access (by all nodes) to the PKG is by no means assured. In the context of a rural-area DTN, users might be forced to travel to a secure location for PKG access. This inconvenience can easily outweigh the benefits of IBE.

Another issue pertains to public domain-specific IBE (PKG) parameters which must be distributed so that nodes can derive public keys from identity strings. Distributing these parameters may be problematic, since users cannot be assumed to belong to the same domain and different domains might have different PKG parameters.

A solution based on Hierarchical IBE (HIBE) was proposed in [8]. It assumes that security management of the entire DTN is organized hierarchically, which is unrealistic, at least for the time being. Also, HIBE offers no forward secrecy, meaning that compromised keys can be used to violate secrecy of earlier messages. We note that [8] presents an extension to provide forward secrecy by using time-based keys issued from a secure location, e.g., the user's personal desktop. However, requiring all DTN nodes to be paired with a separate secure device is impractical.²

²Another problem is that one PKG's compromise causes compromise of all keys issued by all lower-level PKGs. Since HIBE involves a multitude of PKGs, we cannot assume that all of them are impregnable.

Finally, a trivial solution of having pre-shared secret keys for all communicating node-pairs is obviously unscalable. Albeit, in a small and fixed-size DTN, various key pre-distribution schemes [9]–[12] might be applicable.

Given the above issues, the problem at hand is as follows:

How can a DTN node (*SRC*) send a confidential message to another DTN node (*DST*) with no prior security context, i.e., *SRC* neither pre-shares a secret key with *DST* nor does it know *DST*'s public key (if one exists).

In our approach, *SRC* leverages social information, such as affiliation or common social contact(s) to send a secret message to *DST*. *SRC* routes the message encrypted under keys of affiliated intermediate nodes to ensure secure delivery to the destination.

III. NETWORK MODEL AND ASSUMPTIONS

We follow the model, jargon and definitions of the IETF DTNNG Delay-Tolerant Networking Architecture [4]. This model assumes a network with multiple operating regions. Regions are defined by geographic boundaries (e.g., cities). Gateways are fixed nodes which are part of the limited infrastructure; they interconnect regions. If needed, a gateway performs protocol translation between different regions.

Mobile nodes travel between regions and transfer messages when a gateway or another node is encountered. We assume that most nodes within a specific region travel predominately within that region. We make the following further assumptions:

- Each node is uniquely identified by a variable-length Endpoint Identifier (EID). This can take the form of (1) a single EID unique across all regions, if we assume a global addressing scheme, or (2) a tuple of the form $\{Region - ID, Entity - ID\}$, where the former is globally unique (region-specific).³
- The DTN is composed of heterogeneous nodes. However, we assume that all nodes have enough processing power and energy to perform certain symmetric cryptographic operations. More powerful nodes may perform more expensive public-key operations
- Different regions may run different routing protocols but all nodes within a single region use the same routing protocol. Nodes entering a new region are notified of the local routing protocol.
- One or more gateway(s) may interconnect two regions.
- The set of all gateways can be viewed as a DTN-wide overlay network that routes most inter-region traffic.

We address the problem of routing messages both within and across DTN regions in the following sections.

³See [13] for details.

IV. INTRA-REGION MESSAGING

As discussed earlier, it is unlikely that a node can retrieve keys on-demand (or store keys) to communicate with all other nodes. Instead, we take advantage of social information. Suppose that the destination can be linked to some (perhaps well-known) affiliated entity (AE). This entity could be as large as a company or university, or as small as a few mutual friends. Since we can not assume the existence of a global and always-available PKI we assume the AE already knows the destination's public key (through prior communication) or shares a symmetric key. With as few as two common entities a source can send confidential messages to a destination. At the same time, the destination can easily identify that the received message came through friends or AEs, making it less likely to be spam.

A. Details

Even though the message is sent through an entity affiliated with the destination we still need to ensure its secrecy. Users have little to no control over who routes messages since the basic mechanism in many DTN routing schemes [1], [14]–[16] is flooding. We now describe the operation of the intra-region scheme according to notation in Table I:

SRC	source node
DST	destination node
m	message SRC sends to DST
PK_X, SK_X	public-key and corresponding secret-key of entity X
K_X^Y	symmetric key shared by entities X and Y
AE_1, \dots, AE_k	$k > 1$ entities affiliated with DST
$E_X^Y(m)$	encryption by entity Y of message m for entity X using either: (1) a symmetric key K_X^Y , or (2) X 's public key PK_X
$PRF(K)$	a cipher-stream generated by a pseudo-random function keyed with seed K . We assume $PRF()$ has the same bit-size as m .)

Table I
NOTATION

Step 1: SRC determines that DST is affiliated with $t > 1$ entities AE_1, \dots, AE_t present in the same region. SRC can select AEs in several ways including: DST 's employer, DTN operator (service provider), alma mater (e.g., college or university), political/religious organization, hobby or sports club/association and/or DST 's (and SRC 's) common friends. This information can be learned offline and we assume that the sender is aware of at least some such social information. Intuitively, at least $t = 2$ AE-s are needed to offer a minimum level of confidentiality (discussed below).

Step 2: For each AE_i ($0 < i \leq t$), SRC already has either a public key PK_{AE_i} or a shared secret key $K_{AE_i}^{SRC}$.

The latter is likely if AE_i is a common friend, and the former – in most other cases. Either way, SRC constructs the DTN message M_1 , where:

$$\begin{aligned} M_1 &= \langle HDR_1, BODY_1 \rangle \\ HDR_1 &= \langle 1, [AE_1, E_{AE_1}^{SRC}(K_1)], \dots, [AE_t, E_{AE_t}^{SRC}(K_t)] \rangle \\ BODY_1 &= \langle m \oplus PRF(K_1) \oplus \dots \oplus PRF(K_t) \rangle \end{aligned}$$

HDR_1 represents a loose source route (similar to the IP LSRR option [17]). $BODY_1$ is an onion-like structure composed by repeatedly XOR-ing the message with cipher-stream generated by a seed K_i ($0 < i \leq t$) individually encrypted for each AE_i in the header.

Step 3: M_1 eventually reaches the first LSRR hop AE_1 . Recall that AE_1 , like other AE-s in the route, is assumed to “know” DST and either shares with it a secret key $K_{DST}^{AE_1}$ or has DST 's public key. AE_1 decrypts $E_{AE_1}^{SRC}(K_1)$ found in HDR_1 , re-encrypts K_1 as $E_{DST}^{AE_1}(K_1)$, and constructs M_2 , where:

$$\begin{aligned} M_2 &= \langle HDR_2, BODY_2 \rangle \\ HDR_2 &= \langle 2, [AE_1, E_{DST}^{AE_1}(K_1)], \\ &\quad [AE_2, E_{AE_2}^{SRC}(K_2)], \\ &\quad \dots, \\ &\quad [AE_t, E_{AE_t}^{SRC}(K_t)] \rangle \\ BODY_2 &= \langle BODY_1 \rangle \end{aligned}$$

This process is repeated until M_t reaches AE_t . At that time:

$$\begin{aligned} HDR_t &= \langle t, [AE_1, E_{DST}^{AE_1}(K_1)], \\ &\quad \dots, \\ &\quad [AE_{t-1}, E_{DST}^{AE_{t-1}}(K_{t-1})], \\ &\quad [AE_t, E_{AE_t}^{SRC}(K_t)] \rangle \\ BODY_t &= \langle BODY_1 \rangle \end{aligned}$$

AE_t decrypts $E_{AE_t}^{SRC}(K_t)$ found in HDR_t , re-encrypts K_t as $E_{DST}^{AE_t}(K_t)$, constructs M_{t+1} with its two parts HDR_{t+1} and $BODY_{t+1}$ then sends it to DST .

Step 4: Upon receiving M_{t+1} , for each AE_i , DST decrypts the corresponding $E_{DST}^{AE_i}(K_i)$ (using either SK_{DST} or $K_{DST}^{AE_i}$) to obtain K_i . Finally, it computes:

$$m = BODY_{t+1} \bigoplus_{i=1}^t PRF(K_i)$$

One benefit of using stream ciphers in constructing $BODY$ is that there need not be any fixed order of decryption and, hence, of route traversal. The same process as above holds (with minor header modifications) regardless of the order that AE_1, \dots, AE_t are traversed. Another benefit is that a message *does not* have to traverse all AEs if it is replicated by the underlying routing protocol. If the union of all messages received by the destination have traversed all AEs, then the original message can be recovered. This

minimizes the total delay and frees us from making routing decisions.

B. The Poor Man's Approach

The worst-case scenario for DTN security (at least for confidentiality) occurs when SRC cannot find any affiliated entities for DST . The following scheme attempts to make the best of the situation.

- 1) SRC generates a key $K = H(K_1, \dots, K_t)$ where $t > 1$, $H()$ is a suitable cryptographic hash function (e.g., SHA-2 [18]) and K_1, \dots, K_t are random values of appropriate size.⁴
- 2) SRC composes $t+1$ messages: $M_0 = m \oplus PRF(K)$ and $M_i = K_i$ (for $0 < i \leq t$).
- 3) SRC sends these messages in different directions and with a certain delay in between, so as to force them to travel different routes (see Section VI below).
- 4) When M_0, \dots, M_t arrive at DST , the latter recomputes K and decrypts M_0 to obtain m .

One obvious problem is that an intermediate node that receives all $t+1$ messages can easily recover m . Increasing t decreases the probability that a single node can capture all messages, but, it also increases latency.

To increase reliability, forward error-correcting (FEC) codes (e.g., erasure codes [19]) can be used to ensure that receiving $t' < t$ components is enough to reconstruct the entire message. Alternatively, secret sharing [20] can be used for the same purpose. However, neither approach results in better security: if DST can recover the message after receiving $t' < t$ components, so can any intermediate node which receives as many.

A more promising, though opportunistic, measure is to modify the above scheme such that any intermediate node – that shares a key with DST or knows DST 's public key – encrypts the message. Consider the following example. Suppose that, inadvertently, all $t+1$ message components M_0, \dots, M_t traverse the same malicious intermediate node X . However, suppose that *at least one* of the components (M_i) passed through another DTN node Y , before reaching X , and Y knows PK_{DST} or has a shared key K_{DST}^Y . In either case, Y encrypts M_i so that, by the time M_i reaches X , it cannot be used to compute m . Obviously, this method offers only best-effort security and only extensive simulations and experiments can measure its true effectiveness.

V. INTER-REGION MESSAGING

Inter-region messaging occurs primarily between powerful gateways with high-speed links (e.g., 3G, WiFi or WiMax). Gateways are capable of performing cryptographic operations and otherwise expensive multi-round protocols to establish secure channels and forward messages. The main idea is to incorporate gateways into the previous scheme

by using the local gateways of SRC and DST regions as affiliated entities on the path to DST . We assume that the number of gateways in a region is small and fixed. When a node registers with the DTN, it obtains the keys of local gateways.

Step 1: SRC determines DST 's home region by examining the *RegionName* field of the DST 's EID tuple. SRC identifies its local region gateway \mathcal{GW}_{SRC} and DST ' region gateway \mathcal{GW}_{DST} . It then selects $(t-1)$ affiliated entities (AE -s). For each AE_i ($0 < i \leq t-1$), SRC already has either a public key PK_{AE_i} or a shared secret key $K_{AE_i}^{SRC}$. In addition, SRC has $(PK_{\mathcal{GW}_{SRC}})$.

SRC constructs the message M_1 where:

$$\begin{aligned} M_1 &= \langle HDR_1, BODY_1 \rangle \\ HDR_1 &= \langle 1, [AE_1, E_{AE_1}^{SRC}(K_1)], \dots, \\ &\quad [AE_{t-1}, E_{AE_{t-1}}^{SRC}(K_{t-1})], \\ &\quad [\mathcal{GW}_{SRC}, \mathcal{E}_{\mathcal{GW}_{SRC}}^{SRC}(\mathcal{K}_{\square})] \rangle \\ BODY_1 &= \langle m \oplus PRF(K_1) \oplus \dots \oplus PRF(K_t) \rangle \end{aligned}$$

Step 2: At some point, M_1 reaches the first LSRR hop AE_1 . As before, AEs are assumed to “know” DST and either share a secret key $K_{DST}^{AE_i}$ or know DST 's public key. AE_1 decrypts $E_{AE_1}^{SRC}(K_1)$ found in HDR_1 , re-encrypts K_1 as $E_{DST}^{AE_1}(K_1)$ and constructs M_2 where:

$$\begin{aligned} M_2 &= \langle HDR_2, BODY_2 \rangle \\ HDR_2 &= \langle 2, [AE_1, E_{DST}^{AE_1}(K_1)], \\ &\quad [AE_2, E_{AE_2}^{SRC}(K_2)], \\ &\quad \dots, \\ &\quad [AE_{t-1}, E_{AE_{t-1}}^{SRC}(K_{t-1})], \\ &\quad [\mathcal{GW}_{SRC}, \mathcal{E}_{\mathcal{GW}_{SRC}}^{SRC}(\mathcal{K}_{\square})] \rangle \\ BODY_2 &= \langle BODY_1 \rangle \end{aligned}$$

This process repeats itself until the message M_{t-1} reaches AE_{t-1} . At that time:

$$\begin{aligned} M_{t-1} &= \langle HDR_{t-1}, BODY_{t-1} \rangle \\ HDR_{t-1} &= \langle t, [AE_1, E_{DST}^{AE_1}(K_1)], \dots, \\ &\quad [AE_{t-1}, E_{AE_{t-1}}^{SRC}(K_{t-1})], \\ &\quad [\mathcal{GW}_{SRC}, \mathcal{E}_{\mathcal{GW}_{SRC}}^{SRC}(\mathcal{K}_{\square})] \rangle \\ BODY_{t-1} &= \langle BODY_1 \rangle \end{aligned}$$

AE_{t-1} decrypts $E_{AE_{t-1}}^{SRC}(K_{t-1})$ found in HDR_{t-1} , re-encrypts K_{t-1} as $E_{DST}^{AE_{t-1}}(K_{t-1})$, constructs $M_t = \langle HDR_t, BODY_t \rangle$ and sends it to \mathcal{GW}_{DST} .

Step 3: \mathcal{GW}_{DST} receives $M_t = \langle HDR_t, BODY_t \rangle$ and decrypts $E_{DST}^{SRC}(K_t)$ and re-encrypts K_t with $PK_{\mathcal{GW}_{DST}}$

⁴Each K_i is at least s bits long, where s is a security parameter.

and generates M_{t+1} where:

$$\begin{aligned}
M_{t+1} &= \langle HDR_{t+1}, BODY_{t+1} \rangle \\
HDR_{t+1} &= \langle t, [AE_1, E_{DST}^{AE_1}(K_1)], \dots, \\
&\quad [AE_{t-1}, E_{DST}^{AE_{t-1}}(K_{t-1})], \\
&\quad [\mathcal{GW}_{DST}, \mathcal{E}_{\mathcal{GW}_{DST}}^{GW_{SRC}}(\mathcal{K}_{\perp})] \rangle \\
BODY_{t+1} &= \langle BODY_t \rangle
\end{aligned}$$

Routing between gateways is transparent to SRC and DST and can occur across multiple regions or over a direct link.

Step 4: When the message reaches the destination region it is received by \mathcal{GW}_{DST} which decrypts $E_{\mathcal{GW}_{DST}}^{GW_{SRC}}(K_t)$ and re-encrypts it with the key of DST . The message M_{t+1} is then forwarded using the local intra-region routing protocol. Upon receiving M_{t+1} , DST , for each AE_i and \mathcal{GW}_{DST} , decrypts the corresponding $E_{DST}^{AE_i}(K_i)$ (using either PK_{DST} or $K_{DST}^{AE_i}$) to obtain K_i . Finally, it computes:

$$m = BODY_{t+1} \bigoplus_{i=1}^t PRF(K_i)$$

Recall that, despite the description above, AE s need not be traversed in-order and works regardless of region location.

The main difference between inter- and intra-region routing is the involvement of gateways in the former. We cannot assume that gateways share keys with (or know keys of) all end-nodes. Instead, we assume that a gateway only knows keys of nodes in its own region. Gateways work together and use this knowledge to add another layer of encryption to the message. This is particularly important in cases of $t = 2$. With only one common AE , gateways add the critical second layer of encryption to keep the message secure.

VI. SIMULATIONS AND RESULTS

We simulated the proposed schemes using the ONE DTN simulator [21]. Regions were simulated by restricting movement to the downtown area of a large city. Intra-region scenarios allowed free movement around the entire downtown area while inter-region scenarios were restricted to sub-areas. Results are averaged over five iterations.

We simulated users with portable devices walking around during the course of a day sending email messages and pictures. While social contact information can be extracted from sites such as Facebook (discussed below), *user mobility* information cannot. The use of socially-aware mobility data would give further insight but we argue that a synthetic model serves as a lower bound. More intelligent routing algorithms would only improve delivery ratios and ensure – with higher probability – that all AE -s are included on the path to the destination.

Mobility: Node movement was restricted to a map of the Helsinki, Finland downtown area (about 14 km²). i.e., only existing roads and transportation lines could be traveled. 250 nodes move for 24 hours along the shortest path between

two points on the map. This allows for enough time to deliver all generated messages. Nodes move at a speed selected uniformly at random between 0.5 and 1.5 m/s. Upon reaching its destination nodes pause for a period between 0 and 120s, selected uniformly at random, before selecting a new destination. The different seeds generate different mobility patterns for each simulation run.

Connectivity and transmission: User nodes can only communicate with one other node at a time. Communication is bi-directional with a constant transmission rate of 250 kB/s, and continues until all messages have been exchanged or until nodes move outside of the 10 m communication range. Gateways can establish up to ten simultaneous connections, have an 800 m communication range, and a transmission rate of 1 MB/s. Nodes reserve 512 MB of memory for DTN traffic. Routing is handled by a simple flooding protocol, Epidemic [16], and does not use social mobility information. This will give us a lower bound on the effectiveness of our scheme.

Traffic model: Traffic is generated only during the first 12 hours. One message per hour on average is generated in intra-region scenarios and two messages per hour in inter-region scenarios. The message destination is selected uniformly at random among all nodes in all regions. Message sizes are uniformly distributed between 100 kB and 2 MB.

A. Detailed Simulation Results

Intra-Region: The first set of simulations evaluate the suitability of the ‘‘Poor Man’s Approach’’. We focus on the percentage of messages recoverable by honest-but-curious adversaries. In this model, nodes do not actively seek to retrieve messages from a particular source or destination. Instead, opportunistic nodes recover whatever messages possible. Nodes are allowed to keep copies of received messages in a separate storage buffer to recover messages that would have otherwise expired. Recall that, in the ‘‘Poor Man’s Approach’’, the encrypted message and corresponding keys are sent along different paths. We vary the delay between sending the message and key from one up to six hours. We modeled the simplest case where a single key is used to give us an upper bound for interception probability.

Delay (hrs)	Delivery Ratio	Interception Probability	Avg Hop Count	Avg Delay (hrs)
1	.86	.34	2.80	3.06
2	.84	.32	2.80	4.96
3	.83	.27	2.80	6.16
4	.81	.24	2.80	7.17
5	.81	.21	2.79	8.11
6	.81	.18	2.74	9.06

Table II
INTRA-REGION INTERCEPTION PROBABILITY

Table II shows the volume of traffic recoverable by intermediate nodes. We see that sending the message and

corresponding key along different routes offers a fair degree of message confidentiality without impacting overall delivery ratio. This is especially true when the delay is large. However, overall message delivery latency is impacted, i.e., time to receive both message and key.

As expected, inter-message delay is directly related to interception probability. As increasing delay decreases probability of interception. Security for the user becomes a trade-off between probabilistic security and overall delivery latency. We remark that this best-effort confidentiality, only applies in the worst case “Poor Man’s Approach”. The original scheme offers much stronger security guarantees.

Inter-Region: The second set of simulations test the previous parameters under the inter-region scenario. Recall that in the inter-region traffic model message destination is selected uniformly at random amongst all regions. i.e., there is still some intra-region traffic. While system delivery ratios ranged from 87%-90%, filtering out all intra-region traffic left us with delivery ratios ranging from 72%-78%. Results are recorded in Table III.

Delay (hrs)	Delivery Ratio	Interception Probability	Avg Hop Count	Avg Delay (hrs)
1	.77	.42	6.73	2.91
2	.73	.40	6.63	3.94
3	.72	.40	6.63	4.95
4	.73	.40	6.54	5.95
5	.74	.41	6.53	6.53
6	.78	.42	6.60	7.89

Table III
INTER-REGION INTERCEPTION PROBABILITY

At first glance we notice the higher interception probabilities. Comparing Tables II and III we see the clear correlation between hop counts and interception probability. Hop counts directly represent potential adversaries. An increase in the number of intermediaries who ‘see’ a particular message also increases interception probability. As messages travel further in the inter-region scenario, and thus across more intermediaries, we see a greater probability of interception, as expected.

A surprising result is that increasing the inter-message delay does not decrease the probability of recovery. However, this can be explained by the honest-but-curious adversarial model. Any node can keep a message indefinitely by copying it to a separate buffer. Since messages do not expire from this secondary buffer, an adversary just has to wait until both messages are seen. As an extension, we can vary TTLs in addition to inter-message delay. Intuitively, TTLs can be used to reduce the number of nodes who receive messages (since messages will expire from the network), but will also negatively impact delivery ratios since messages may not have enough time to traverse the network.

Overall, the “Poor Man’s Approach” has the potential to be used if no other options are available and preferably in

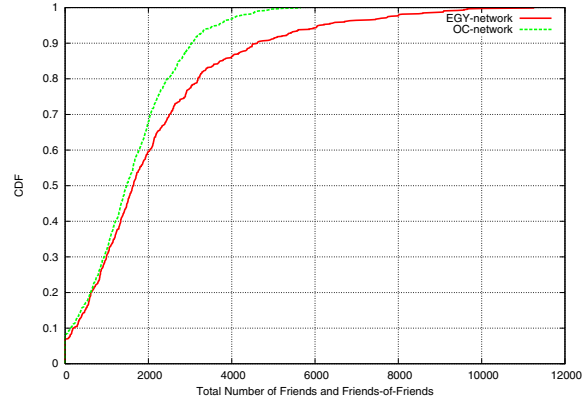


Figure 2. CDF of Social Network Reach for Facebook Users (OC and EGY Networks) intra-region scenarios. When possible, users are advised to minimize hop counts needed through TTLs or other means.

B. Social Network Coverage

To determine the social coverage of friends and of friends-of-friends (FoFs) inside a region we analyze social connections from a well known social network, Facebook. Facebook is a social network that contains 150 million [22] users from all over the world. Users can join smaller networks inside Facebook which correspond to countries, cities, organizations and institutions. We select two networks that corresponds to two geographical areas (Orange County (OC) and Egypt (EGY)) and crawl their users. We choose these two networks as representatives of small geographical areas (counties) and developing countries where DTNs may provide a cost efficient solution ⁵. We are interested in answering the following questions in our analysis: *How many friends (and FoFs) does each user have on average? and if the size of the social graph depends on the number of immediate (one hop) friends?*

It is important to study the two characteristics above because they directly affect the applicability of our schemes. If users have few friends and few FoFs then the reachability of their social network will be limited and the number of stored keys large (one for each destination). On the other hand if the number of friends is one (or more) orders of magnitude smaller than the number of people they can reach through them (i.e., FoFs), then only a few keys will be needed for high social reachability.

Figure 2 shows the cumulative distribution function(CDF) of the total number of friends and FoFs for 870 and 900 random Facebook users in the OC and EGY networks respectively ⁶. On average half the users in both networks can reach less than 1700 users, the other half more than 1700. In the OC network users the highest number of

⁵We do not claim that this small study of the social graphs of these two networks is exhaustive, but it serves to prove our points.

⁶Closed Facebook profiles are visible only to friends. The numbers presented are conservative and do not include such closed profiles; thus results can be considered a lower bound on the social reachability of users.

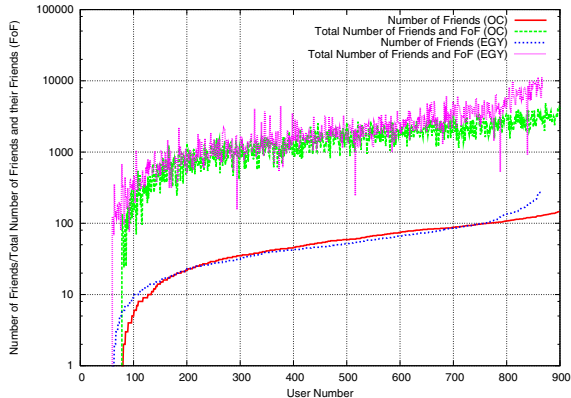


Figure 3. Number of Friends and the FoFs for a Sample of Users (OC and EGY Networks) reachable people was 5881 while the EGY it was around 11000. Our analysis indicates that by storing only several hundred of keys at max, one can reach several thousands of users. This can be seen in more detail in Figure 3. The figure shows the number of immediate friends of the sampled users from the OC and EGY networks. Clearly the number of direct friends for 800 of them is below 100 and the number of reachable FoFs is in the order of thousands for over 400 users. A small number of users (less than 10%) have more than 100 friends and up to 500 and their reachable FoF network goes as high as 10000. It is interesting to note that although we selected two completely unrelated networks on Facebook, and randomly selected the set of users to crawl, the distribution of the number of friends and FoF look similar, although people in the EGY network seem to have a larger total social graph on average. More exhaustive measurements need to be conducted to come up with strong conclusions about the exact average distributions, but this is out of scope of this paper.

Note that we have only considered the analysis of the social network coverage of users by considering only their friends. As we have described in previous sections one can also use the affiliations of users (e.g. their work, study or organizations) for confidential messaging. Unfortunately such a concept does not exist in the social networks and is hard to map into these networks. For example users in Facebook can belong to groups, but these groups are formed randomly between friends and are very transient (with some exceptions of course). On the upside, the capability to use such affiliation will strictly increase the applicability of our proposed scheme and allow users to reach more people.

VII. SECURITY ANALYSIS AND LIMITATIONS

In this section, we present a preliminary – and quite informal – security assessment of the proposed schemes. (A more formal treatment of security is underway.) Recall that the goal is secrecy of messages sent between two users who have no prior security context.

Security Model: We assess the security of our schemes under the *honest-but-curious* adversarial model. In this

model, nodes do not mount active attacks to learn messages and do not modify or drop messages. Violations of message integrity is the result of corrupted transmission or corrupted memory/storage. However, nodes can retain copies of messages beyond their TTL expiration in separate storage. This allows them, for example, to correlate message components sent with variable delays in our inter-region scheme.

Collusion: The biggest threat to message confidentiality is collusion between the intermediate affiliated entities AE_1, \dots, AE_t . We argue that this attack is unlikely, since these entities are selected by *SRC* who would avoid entities naturally prone to collusion (e.g., all government agencies or all cell phone service providers). Another issue is that many of these entities are public and operate on a long-term basis; therefore, they have little incentive to sacrifice their reputation by engaging in collusion attacks.

If *AEs* are friends we assume that *SRC* trusts its friends not to collude. *SRC* is also well-advised to pick friends who do not know each other, but are socially connected to both *DST* and *SRC*. (The counter-argument is that two people who have two friends in common are apt to know each other. Further investigation is needed to verify this claim).

MitM Attacks: Man-in-the-Middle (MitM) attacks are not applicable to our setting since public keys for affiliated entities (selected by *SRC*) are obtained from public key certificates signed by trusted Certificate Authorities (CAs).

Authentication and Integrity: Origin authentication and data integrity are not our primary goals. Moreover, they are not within the scope of the *honest-but-curious* adversary model. However, we note that *SRC* can easily authenticate itself to *DST* by enclosing its public key certificate with the message and signing the message before encryption. Whereas, if *SRC* has no public key certificate, meaningful sender authentication is impossible. However, a weak form of message integrity is possible: *SRC* can compute and append a message hash before encryption.

Limitations: While the proposed schemes represent a step forward the underlying assumptions may be problematic. In particular, we assumed that *AEs* know or can easily obtain the *DST*'s public key (or share a secret with it). This may not be the case nor does this assumption scale. Alternative mechanisms should be explored to relax this assumption. One potential consideration is the availability of a pervasive cellular telephone network (e.g., GSM), alongside the envisaged DTN. We can use SMS to transmit keys used for bulk encryption. Another possibility is to send keys on an explicitly circuitous route to ensure that they do not travel close to the encrypted message.

VIII. RELATED WORK

DTNs are an active research area with numerous recent results. An architecture for “challenged” networks characterized by very large delay paths and frequent network disruptions is presented in [13]. This design uses messages as the

underlying unit of transmission. The architecture proposes a two-tier naming structure, which we briefly described earlier in the paper. [13] also proposes a hop-by-hop security model, addressing issues such as authentication, confidentiality and integrity by requiring end-hosts to have certificates which bind identities to public keys. This is problematic due to key management and certificate revocation issues. We cannot assume always available online entities due to frequent disconnections nor can we assume timely query responses. In our approach, nodes are not required to obtain public keys of peers. They only need to obtain (or already have) keys for affiliated entities.

The idea of using social information in DTNs was first proposed in [23] which describes a publish-subscribe routing framework based on information of social interactions among users. The principal idea is that socially-related people frequently co-locate. Information of interest to people sharing a common interest can be quickly spread to these users using social routing. This strengthens the applicability and motivation of our schemes because we too use social information/social contacts as intermediaries for confidential messaging. If socially-related people frequently co-locate, then using such people as affiliated entities increases the probability of message delivery. However, as mentioned earlier, care must be taken to ensure *SRG* picks friends who, although likely to co-locate, are unlikely to collude.

There have been a few research results in DTN security. For example, [24] discusses various threats and issues but focuses primarily on so-called *bundles* and their security. It briefly addresses security services on an end-to-end basis (e.g. confidentiality), but does not go into specifics nor considers the case of initial communication between two nodes without any prior security context.

Note: Related work utilizing Identity Based Cryptography (IBC) [7], [8], [25], [26] has been discussed in Section II.

IX. CONCLUSION

This paper motivated and presented techniques for secure messaging in large-scale DTNs. They allow DTN users to leverage social contact information in order to exchange confidential and authenticated content. We showed how to address the problem in both intra- and inter-region settings and assessed performance and security of our techniques. Future work includes a more formal security analysis and extensive simulations to help identify performance and security bottlenecks and limitations.

REFERENCES

- [1] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," *INFOCOM 2006. IEEE International Conference on Computer Communications. Proceedings*, April 2006.
- [2] A. Pentland, R. Fletcher, and A. Hasson, "Daknet: Rethinking connectivity in developing nations," *IEEE Computer*, vol. 37, no. 1, pp. 78–83, January 2004.
- [3] "Wizzy digital courier," <http://www.wizzy.org.za>.
- [4] V. C. et al, "Delay-tolerant network architecture," April 2007.
- [5] S. B. et al., "Delay-tolerant networking : An approach to interplanetary internet," *IEEE Communications Magazine*, pp. 128–136, June 2003.
- [6] C. Peoples, G. Parr, B. Scotney, and A. Moore, "A reconfigurable context-aware protocol stack for interplanetary communication," *Satellite and Space Communications, 2007. IWSSC '07. International Workshop on*, Sept. 2007.
- [7] K. Fall, "Identity based cryptosystem for secure delay tolerant networking," December 2003.
- [8] A. Seth and S. Keshav, "Practical security for disconnected nodes," *Secure Network Protocols, 2005. (NPSec). 1st IEEE ICNP Workshop on*, pp. 31–36, Nov. 2005.
- [9] A. Aiyer, L. Alvisi, and M. Gouda, "Key grids: A protocol family for assigning symmetric keys," in *ICNP '06: Proceedings of the Proceedings of the 2006 IEEE International Conference on Network Protocols*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 178–186.
- [10] C. Castelluccia, "Securing very dynamic groups and data aggregation in wireless sensor networks," *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE International Conference on*, pp. 1–9, Oct. 2007.
- [11] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2002, pp. 41–47.
- [12] N. Mittal, "Space-efficient keying in wireless communication networks," *Wireless and Mobile Computing, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on*, pp. 75–75, Oct. 2007.
- [13] K. Fall, "A delay-tolerant network architecture for challenged internets," in *SIGCOMM '03*. New York, NY, USA: ACM, 2003, pp. 27–34.
- [14] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.
- [15] T. Sphyropoulos, K. Psounis, and C. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in *Proceedings of ACM SIGCOMM'05*, August 2005.
- [16] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke University, Tech. Rep. CS-200006, April 2000.
- [17] D. B. Johnson, "Mobile host internetworking using ip loose source routing," Tech. Rep., 1993, available at: <http://www.cs.rice.edu/~dbj/pubs/CMU-CS-93-128.pdf>.
- [18] "National institute of standards and technology (nist), fips publication 180-3: Secure hash standard," June 2003.

- [19] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman, and V. Stemann, "Practical loss-resilient codes," in *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. New York, NY, USA: ACM, 1997, pp. 150–159.
- [20] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.
- [21] A. Keränen and J. Ott, "Increasing reality for dtn protocol simulations," Helsinki University of Technology, Tech. Rep., July 2007, available at: <http://www.netlab.tkk.fi/tutkimus/dtn/theone/>.
- [22] "Facebook statistics," <http://www.facebook.com/press/info.php?statistics>.
- [23] P. Costa, C. Mascolo, M. Musolesi, and G. Picco, "Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks," *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 5, pp. 748–760, June 2008.
- [24] S. Farrell and V. Cahill, "Security considerations in space and delay tolerant networks," in *IEEE SMC-IT '06*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 29–38.
- [25] C. Gentry, "Certificate-based encryption and the certificate revocation problem," 2003. [Online]. Available: citeseer.ist.psu.edu/gentry03certificatebased.html
- [26] A. Seth, S. Fung, and S. Keshav, "A secure tetherless computing architecture," University of Waterloo, Tech. Rep.