FAN: Friendship Based Anonymous Routing in Wireless Social Mobile Networks of Malicious Communities

Antonio Davoli
Computer Science Dept.
Sapienza Univ. of Rome, Italy
davoli@di.uniroma1.it

Alessandro Mei
Computer Science Dept.
Sapienza Univ. of Rome, Italy
mei@di.uniroma1.it

Julinda Stefa
Computer Science Dept.
Sapienza Univ. of Rome, Italy
stefa@di.uniroma1.it

ABSTRACT

Pocket Switched Networks (PSN) whose main feature is the social-guided movement of users/devices, have attracted the attention of many researchers in the last years, mostly for the common belief to be a key technology in providing innovative services without the need of a fixed network infrastructure. However, the opportunistic and intermittent nature of the contacts among users make it very difficult to design secure and trustworthy services for these networks. In particular, anonymous communication remains among the most difficult services to achieve.

In this paper we present FAN (Friendship based ANonymity), a primitive that exploits strong friendship relationships among users to provide source anonymity and sender-receiver unlinkability in Pocket Switched Networks. The primitive is independent of the forwarding mechanism underneath, and therefore, can be coupled with any routing protocol. As shown from our large experimental results with different real data traces, FAN outperforms the TPS scheme, its only rival that provides the same anonymity properties for Pocket Switched Networks, in terms of delay, cost, and network throughput.

Categories and Subject Descriptors

C.2.1 [**Computer-Communications Networks**]: Network Architecture and Design—*Distributed networks, Network communications, Wireless Communication*; C.2.2 [**Computer-Communications Networks**]: Network Protocols—*Routing Protocols*

Keywords

DTNs, pocket switched networks, anonymous routing.

1. INTRODUCTION

The intermittent connectivity in Pocket Switched Networks (PSN) [10] turns even end-to-end communication between devices into a challenging task to achieve. Messages are routed in store-and-forward fashion, by trying to push them towards relays that are more likely to provide eventual delivery. Though flooding is a possibility [24], many research work in the area is dedicated to designing forwarding protocols that assure acceptable network throughput by trying to overload as little as possible the network with message replicas [23, 6, 11, 9, 14, 17]. Most of the aforementioned protocols rely on the possibility to exploit information such as who met who, how often did such meeting happen, and what are the odds of it to happen again in the future. And often, making use of this kind of information is the only possible solution to keep a reasonable network throughput, by still keeping message replicas number low. However, this mechanisms clearly make communication paths between nodes easily traceable.

In this work we focus on obtaining efficient and anonymous communication among peers in Pocket Switched Networks. We first introduce a new type of attack–the malicious community attack–where nodes of a single (possibly big) social community aim at compromising anonymity of other network nodes. We show how the state-of-the-art anonymity scheme for PSNs fails in providing source anonymity and source destination unlinkability under such attack. Then we present FAN (Friendship based ANonymity), a primitive that is resistant to the malicious community attack, that provides route anonymity and source-destination unlinkability in PSNs, by still being way more efficient than its rival TPS [12]. The primitive is independent of the forwarding mechanism underneath, and therefore, can be coupled with any routing protocol. We evaluate our scheme by extensive experiments with three different real data-traces.

The rest of this paper is organized as follows: Section 2 reports on the related work in the area. Sections 3 and 4 respectively define our system model and show how TPS fails providing the claimed anonymity properties under the malicious community attack. In Section 5 we present our anonymity scheme, FAN, whereas in Section 6 we show the results of its experimental evaluation. Finally, we conclude the paper with Section 7.

2. RELATED WORK

Anonymous routing schemes are not new in the area of wireless ad-hoc networks [2, 20, 22, 25]. Most of these schemes are based on the concept of mixing [5], where messages are sent along a chain of proxy nodes–called mixes–that accumulate and forward source-encrypted messages in batches. Tor [7],perhaps the most popular deployed mix network, achieves mixing by layer-encrypting a message at the

ExtremeCom '11, September 26-30, 2011, Manaus, Brazil.

source and decrypting once at each hop of a source-selected chain of proxy relays (called also Tor nodes). The last Tor node of such chain sends the unencrypted message to the destination specified by the client.

These approaches either rely on source-routing, or assume a reliable network where full connectivity among pears is available, which makes them unaplicable in delay tolerant networks of intermittent connectivity such as the PSNs. To the best of our knowledge [16] and [12] are the only approaches that cope with the intermittent nature of PSNs. In [16] the authors present ALAR, an anti-localization routing protocol aiming at not revealing the data-source while maximizing the destination's probability to receive the message. However, it does not protect the identity of the destination of the message. From the other side, TPS [12], combine threshold-based cryptography along with randomly selected pivot nodes in order to provide resistance to traffic analysis, source anonymity, and sender-receiver unlinkability in the network. Nevertheless, as we will show in the next session, TPS fails providing the claimed properties under a very natural type of attack in social mobile networks–the attack of malicious communities–where nodes of a single (possibly big) social community aim at compromising anonymity of other network nodes.

3. SYSTEM MODEL

Our network setting is made of last generation smartphones, able to communicate by using short-range communication technology, like blue-tooth and/or Wi-Fi, aside classic GSM. When two people meet, their personal devices establish a link by using one of the above mentioned short-range technologies. In the network we consider, nodes are devices carried by people, and links appear and disappear as people move and meet. Smart-phones are not-so-small devices that can easily handle video/audio streaming, 3D games, web surfing and SSL sessions, and other applications. Therefore, we can safely assume that nodes are able to perform public key cryptography that is used to sign messages and to establish secure communication sessions among peers. The nodes are equipped with public/private key pairs, and the former is signed by a trusted authority CA.

Lastly, we assume that nodes are also equipped with another public/private key pair, that of the social community they belong to. These keys are also given to the nodes from the CA, when this latter is "securely convinced" that a node, say node A, really belongs to community C. A mechanism to achieve secure authentication of a user as a member of a social community is described in [1]. Furthermore we assume that nodes of the same community are "real friends", and do not betray other community members.

3.1 The Malicious Community Attack

In real systems, the anonymity of an individual (user) is relative to other individuals (users) in the same system: The degree of anonymity depends on the set of system users and on the probability of de-anonymizing a certain user. A communicating system is said to have sender anonymity (receiver anonymity) if the source of any message cannot be distinguished from other senders (receivers) in the system. However, mostly of the deployed systems often achieve anonymity by making it impossible to link sender-receiver pairs as communication partners. This property is also known as unlinkability.

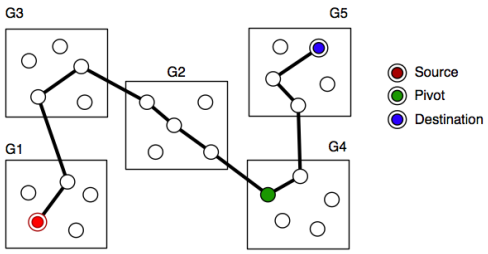


Figure 1: The TPS protocol.

The adversary attacking our network is somewhat stronger than the one considered in [12]. Our adversary can control a large number of colluding network nodes, the only restriction being for these nodes to belong to the same community–the malicious community. These nodes, other than with opportunistic contacts, are also able to communicate with other and faster communication channels (e.g. GSM or 3G). Such a scenario is more than real: One can imagine the malicious community to be a mafia organization in a city. The goal of the malicious community is to de-anonymize sender/receiver couples of network individuals. The goal of our scheme is to achieve sender/receiver anonymity and source-server unlinkability under such attack.

4. TPS VS THE MALICIOUS COMMUNITY ATTACK

The main idea behind the TPS [12] scheme is as follows: The n network nodes are divided into k random groups of the same size n/k. Each node is equipped with a private/public key pair, as well as the private/public key pair of the group it belongs to. When a node A wants to send a message to another node B, first a pivot node P is selected by sampling uniformly at random over the network nodes. Then both the message and $B's$ identity are encrypted with $P's$ public key. Afterwards, A generates a one-time threshold secret (τ, k) using Shamir's scheme [21]:

$$s = (S_1, .., S_k),$$

where k is the group's number, and τ is the minimum number of $S_i's$ necessary to recreate s. Then A encrypts $P's$ identity with s, encrypts each of the $S_i's$ with the respective's group public key PK_i, and concatenates everything into a unique final bundle β. Such bundle then starts traveling within the network, passing from node to node. Each time it enters in a non previously encountered group i the respective share S_i is decrypted with that group's private key (held by any node of that group). As soon as τ shares are decrypted, the secret s is revealed, and so is the identity of the pivot node P. At that point the remaining of the bundle is forwarded towards P, which, in turn, decrypts $B's$ identity and then forwards the rest towards the destination B (see Figure 1).

The scheme works as charm when all the attacker can do is compromise one node of the network. But what happens when we consider a malicious community of colluding nodes? First, note that, if the cardinality of such community is m, than: (1) Since the pivot node P is randomly selected over the network, the chances for it to belong to the malicious community are m/n; (2) since the groups are created at random, then, in average, m/nk nodes of each group are malicious. Now, if the first relay of the bundle created by

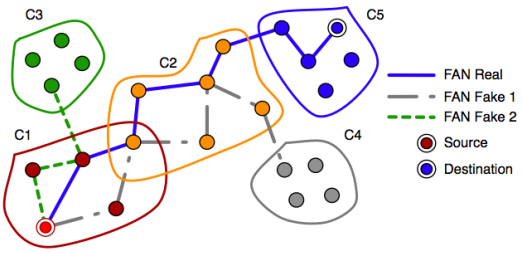


Figure 5: FAN protocol. Here the packet is forwarded towards the *fake* destination communities $C2$ and $C3$ and towards the *real* destination community $C5$.

A goes to a malicious node R_m, after decrypting its own group's share, R_m sends it directly (through e.g. GSM) to another malicious member of its community belonging to a group different from its. The process goes on till the secret s is revealed, and so is the pivot's identity P. At that point, if also the pivot P happens to be a member of the malicious community, then also the destination's identity B is revealed the communication between A and B is de-anonymized.

The power of the malicious community attack in a system where the TPS scheme is deployed clearly depends on the size of the community itself, as well as on the meeting patterns among nodes. We study it by investigating the rate of the de-anonymized messages this attack on three real datatraces: Dartmouth [13], Reality [8] and Infocom06 [19]. For this, we create 1000 messages with a uniform traffic pattern among the legitimate nodes, and forward them into the network. Each time the malicious attack succeeds (both the first relay of the message is a malicious node and the pivot is a malicious node) we consider the message as de-anonymized. For the forwarding, we use 3 different protocols: (1) The one-copy routing used in [12] where the first node encountered is selected as relay; (2) Bubble forwarding protocol [11] where groups are considered as communities, and the forwarding policy is to send the message outside the current relay's group; (3) Delegation Forwarding [9] where as forwarding quality we adopt (a) the rate of different groups one node has meet till the pivot is reached, and (b) Destination last contact afterwards. As group number we use divisors of the total number of nodes in each scenario. For each forwarding protocol we repeat the experiment 10 times, and then average the results. The final experimental outcome is presented in Figures 2, 3, and 4. As you can notice, in all scenarios the rate of de-anonymized messages grows with the growing of the malicious community's cardinality. However, malicious communities as small as 15% of the network already de-anonymize 1 message over 10. The de-anonymizing rate goes up to 1 message over 5 (more than 20%) with malicious communities of 30% of network nodes when Delegation is used as routing protocol (see Figure 4). This means that in a system where TPS is deployed, 1 every 5 messages' source-destination are de-anonymized, which is unacceptable for an anonymity scheme. Thus, the TPS scheme clearly fails to provide anonymity in PSN networks of malicious communities.

5. THE FAN ANONYMITY SCHEME

Now we present our anonymity scheme FAN (Friendship based ANonymity), which leverages source and destination anonymity even under the malicious community attack. The idea under such scheme is as follows: The source node A that wants to send a message m to a destination B, first encrypts it with $B's$ public key by creating $\langle m \rangle_{PK_B}$. Then A selects t communities uniformly at random, and generates $t+1$ different packets by encrypting $\langle m \rangle_{PK_B}$ with the public key of each of the t previously selected communities, and that to which B belongs. Than, the packets are routed in the network towards the "destination community" (see Figure 5).

Whenever one of the packets reaches its *real* destination community, the outer encryption layer is taken away, and the inner packet is forwarded inside of the community with the hope to get to the destination node. Other nodes (members of the community) do not understand who the packet is for, since $B's$ ID is encrypted within the inner layer of the packet, with $B's$ public key. However, by forwarding it inside the community, eventually it gets to B. Since the inner packet is encrypted with the destination's public key PK_B, B is the only node able to decrypt it.

The other packets sent towards fake communities eventually reach them. The outer encryption layer is taken away and the packet begins to being forwarded inside the community hoping to get it to destination, till it expires.

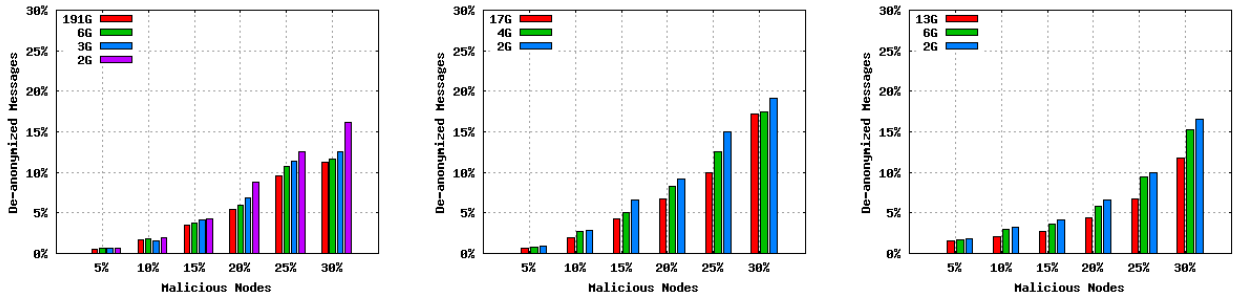
In our scheme communities are made of nodes that are socially-related to each other. To get the own's community credentials a node has to prove that it belongs to that community. This can easily be done by using a scheme like in [1]. In such a scenario the malicious nodes will fall within the same community.

It is straightforward to see how our scheme provides source and destination anonymity in the case in which the malicious community does not overlap with other communities. Now, let us suppose that the malicious community overlaps with the source community. Since the source node's ID is only included in the inner packet, encrypted with $B's$ public key, and since the different packets created by A are encrypted with public keys of $t-1$ communities, even if a malicious node happens to be a good relay to all of these communities, it cannot tell whether A is just a relay or the source node. From the other side, if the malicious community overlaps with the destination community (say community C) the overlapping member is likely to either get the packet and take away the outer encryption layer, or, to get the inner packet after another member of C has decrypted the outer layer. In both cases, since the packet is encrypted with $B's$ public key, and the routing inside the destination community is plain epidemic, the malicious relay node is not able to tell who's the destination of the packet. Moreover, since A sends $t+1$ packets, there are $t+1$ destination communities. Thus, the malicious node is not even able to tell whether the destination is within community C or not.

Clearly the security of the FAN scheme depends on the tunning parameter t of fake destination communities. The bigger such parameter the better the security, but, at the same time, the higher the overhead induced in the network, and vice versa. However, our experimental results of the next session will show that, even choosing t such that $t+1$ is the overall number of network communities, our scheme is much better performing than the TPS scheme.

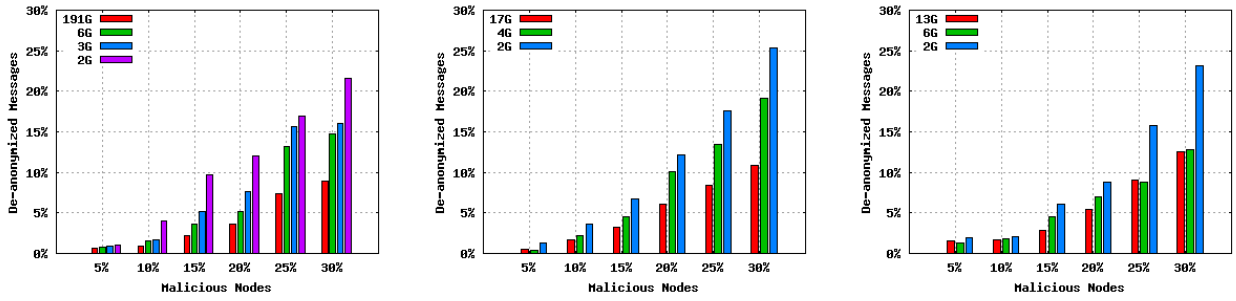
6. EXPERIMENTAL RESULTS

Here we study the performance of the FAN anonymity routing scheme, compared to that of the TPS scheme. We



(a) Dartmouth (b) Reality (c) Infocom06

Figure 2: Rate of de-anonymized messages routed with one-copy routing.



(a) Dartmouth (b) Reality (c) Infocom06

Figure 3: Rate of de-anonymized messages routed with Bubble.

detect social communities in each of the scenarios using the k-clique protocol [18, 11]. The number of different communities detected in each scenario are respectively: 58 of 20 members in average for Dartmouth, 4 of 17 members in average for Reality, and 4 of approx. 9 members in average for Infocom06. As in the results of Section 4 we generate 1000 messages through uniform traffic pattern and route them in the network with the FAN scheme combined with one-copy routing, Bubble and Delegation Forwarding.

First we investigate the performance of our scheme for different values of the parameter t (the number of fake destination communities) in terms of success rate. As expected, such parameter does not impact the success delivery rate nor the average delivery delay. Though, as we already discussed, it does impact the overhead induced in the network. However, we decide to set such parameter to the maximum value possible, and see, in comparison to the TPS scheme, what is the overload induced in the network. From the other side, for the TPS scheme we chose to set the parameter k (the group number) in such a way that the success delivery for every trace is maximized: 191 groups for Dartmouth, 4 groups for Reality, and 13 groups for Infocom06.

We assume that each message body is 140 bytes long (e.g. Tweets or SMS messages), whereas, an ID field is 32 bytes long. The metric we use for storage is Mega byte minute (Mb minute). This metric is intuitive, it is clearly cheaper to store one SMS for a second versus storing the same SMS for a minute. In case of high traffic, the difference can be huge—if a node relays one SMS a second, it stores an average of 60 messages per minute in the first case and 3600 messages per minute in the second case.

We assume that elliptic curve cryptography [15] is used for encryption, which is considered to be the most efficient encryption scheme. FAN packets contain the encryption of the message body ($752b$), the destination's IDb ($192b$) and the destination community's ID ($192b$). Thus, a FAN packet becomes $1136b$ long.

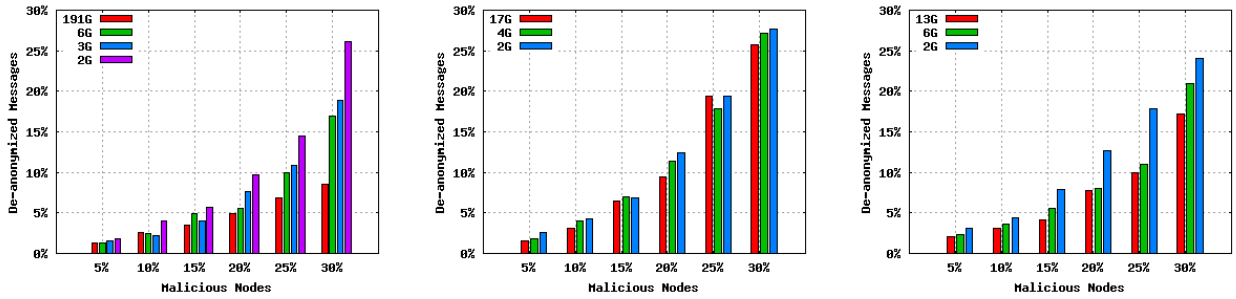
In the TPS case, a packet does not only contain the encryption of the message body ($752b$), and the destination ID ($192b$), but it also contains the encryption of the pivot node ID ($192b$) and the encryption of each of the k secret shares. According to [3, 4] each share's length should be as big as that of the secret (which is an RSA key of length 1024 bytes). Thus, the size of a TPS packet becomes $752b + 192b + 192b + 1024 \times k$ where k is the number of the groups in the network, which is, respectively for each of the scenarios, 196 kbytes (Dartmouth), 5k (Reality), and 14k (Infocom06).

Figure 6 depicts the memory occupation per delivered message induced by each scheme. As you can notice FAN always outperforms its TPS alter ego in terms of storage efficiency. Even though it induces more copy in the network, due to the fake destination communities, it still keeps the overhead low. Similarly, in Figure 7 we show the average delay per delivered message of both schemes. As you can notice, FAN, combined with any of the forwarding protocols, quickly pushes the message towards the destination.

Finally, Figure 8 shows the success delivery of both schemes when combined with each of the three forwarding protocols. Again, FAN outperforms its rival TPS in all three scenarios. In particular, in the Dartmouth and Infocom06 scenarios case (see Figures 8(a) and 8(a)), the out-performance of FAN is prominent!

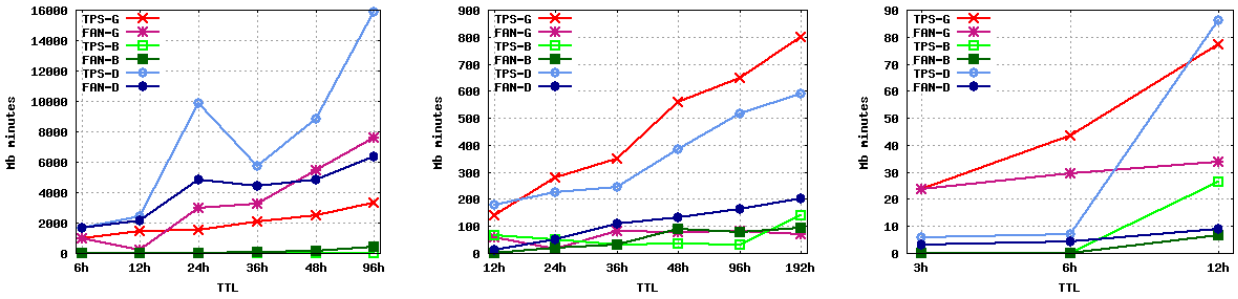
7. CONCLUSIONS

In this paper, we introduce the attack of the malicious community, a vicious attack to anonymity schemes in pocket



(a) Dartmouth (b) Reality (c) Infocom06

Figure 4: Rate of de-anonymized messages routed with Delegation Forwarding.



(a) Dartmouth (b) Reality (c) Infocom06

Figure 6: Memory occupation by the two schemes. TPS-G, TPS-B, TPS-D are respectively TPS single-copy routing, TPS-Bubble and TPS-Delegation. FAN-G, FAN-B, FAN-D are respectively FAN single-copy routing, FAN-Bubble and FAN-Delegation.

switched networks. We show by experimental means how, the state of the art anonymity scheme for DTNs, the TPS scheme, fails providing the claimed properties under this attack. Then, we propose FAN, the Friendship-based ANonymity scheme. FAN, not only provides sender and receiver anonymity in PSN's also in presence of malicious communities, but in addition, it outperforms its rival TPS when such attack is not deployed in terms of success, delivery delay, and storage.

8. REFERENCES

[1] M. V. Barbera and A. Mei. Personal marks and community certificates: Detecting clones in mobile wireless networks of smart-phones. Preprint: http://arxiv.org/abs/1105.3716, 2011.

[2] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba. Sdar: A secure distributed anonymous routing protocol for wireless and mobile ad hoc networks. In *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN04)*, 2004.

[3] R. M. Capocelli, A. D. Santis, L. Gargano, and U. Vaccaro. On the size of shares for secret sharing schemes. *Journal of Cryptology*, 6:157–168.

[4] M. Carpentieri, A. D. Santis, U. Vaccaro, A. De, and S. U. Vaccaro. Size of shares and probability of cheating in threshold schemes. In *In Proc. of Eurocrypto'93, Lecture Notes in Computer Science, LNCS 765*, pages 118–125. Springer-Verlag, 1993.

[5] L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24:84–90, February 1981.

[6] E. M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, 2007.

[7] R. Dingledine, N. Mathewson, and P. P. Syverson. Tor: the second-generation onion router. In *Proceedings of the 13th conference on USENIX Security Symposium (SSYM'04)*, 2004.

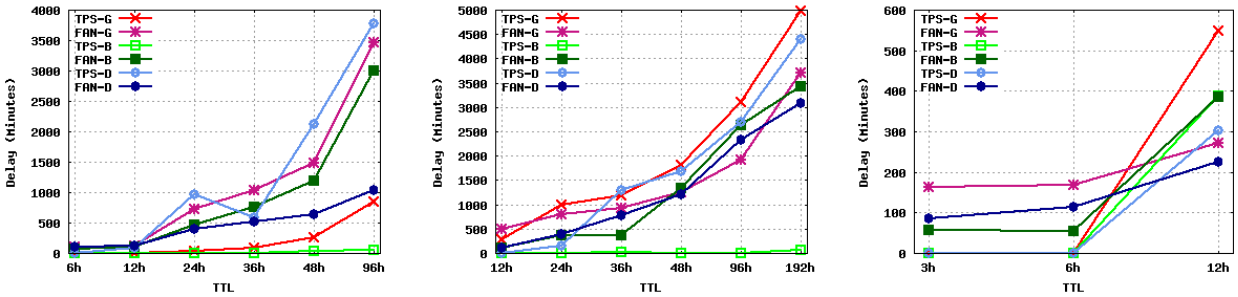
[8] N. Eagle and A. S. Pentland. CRAWDAD data set mit/reality (v. 2005-07-01). Downloaded from http://crawdad.cs.dartmouth.edu/mit/reality, July 2005.

[9] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot. Delegation forwarding. In *MobiHoc '08: Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, 2008.

[10] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot. Pocket switched networks and human mobility in conference environments. In *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, 2005.

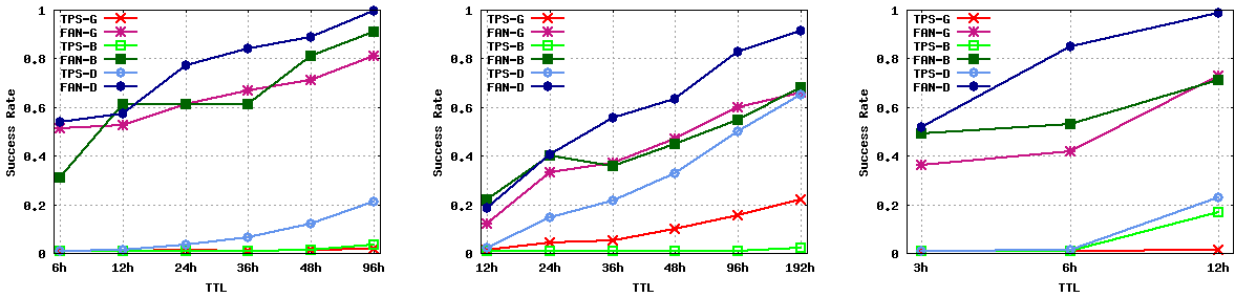
[11] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: social-based forwarding in delay tolerant networks. In *MobiHoc '08: Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, 2008.

[12] R. Jansen and R. Beverly. Toward Anonymity in Delay Tolerant Networks: Threshold Pivot Scheme. In *Proceedings of The Military Communications*



(a) Dartmouth (b) Reality (c) Infocom06

Figure 7: Average delay of the two schemes. TPS-G, TPS-B, TPS-D are respectively TPS single-copy routing, TPS-Bubble and TPS-Delegation. FAN-G, FAN-B, FAN-D are respectively FAN single-copy routing, FAN-Bubble and FAN-Delegation.



(a) Dartmouth (b) Reality (c) Infocom06

Figure 8: Average success rate of the two schemes. TPS-G, TPS-B, TPS-D are respectively TPS single-copy routing, TPS-Bubble and TPS-Delegation. FAN-G, FAN-B, FAN-D are respectively FAN single-copy routing, FAN-Bubble and FAN-Delegation.

Conference (MILCOM 2010).

[13] D. Kotz, T. Henderson, I. Abyzov, and J. Yeo. CRAWDAD data set dartmouth/campus (v. 2009-09-09). Downloaded from http://crawdad.cs.dartmouth.edu/dartmouth/campus, Sept. 2009.

[14] F. Li and J. Wu. LocalCom: a community-based epidemic forwarding scheme in disruption-tolerant networks. In *SECON'09: Proceedings of the 6th Annual IEEE communications society conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2009.

[15] A. Liu and P. Ning. Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks. In *Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN 2008), SPOTS Track*, 2008.

[16] X. Lu, P. Hui, D. Towsley, J. Pu, and Z. Xiong. Anti-localization anonymous routing for delay tolerant network. *Computer Networks*, 54(11):1899 – 1910, 2010.

[17] A. Mei, G. Morabito, P. Santi, and J. Stefa. Social-Aware Stateless Forwarding in Pocket Switched Networks. In *Proceedings of The 30th IEEE Conference on Computer Communications, mini-conference, (INFOCOM 2011)*, April 2011.

[18] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043), 2005.

[19] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. CRAWDAD trace cambridge/haggle/imote (v. 2009–05–29). Downloaded from http://crawdad.cs.dartmouth.edu/cambridge/haggle/imote, May 2009.

[20] S. Seys and B. Preneel. ARM: anonymous routing protocol for mobile ad hoc networks. *Int. J. Wire. Mob. Comput.*, 3:145–155, October 2009.

[21] A. Shamir. How to share a secret. *Communications of the ACM*, 11(22):612–613, 1979.

[22] R. Song, L. Korba, and G. Yee. Anondsr: efficient anonymous dynamic source routing for mobile ad-hoc networks. In *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks (SASN'05)*, 2005.

[23] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *ACM Sigcomm Workshops*, Aug. 2005.

[24] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, 2000.

[25] Y. Zhang, W. Liu, W. Lou, and Y. Fang. MASK: Anonymous on-demand routing in mobile ad hoc networks. *IEEE Transactions on Wireless Communications*, 21:2376–2385, 2006.