LETTER

Recipients' Anonymity in Multihop Ad-hoc Networks

Helena RIFÀ-POUS[†], ***Member***, **Emmanouil A. PANAOUSIS**[††], ***and*** **Christos POLITIS**[††],

SUMMARY Multihop ad-hoc networks have a dynamic topology. Retrieving a route towards a remote peer requires the execution of a recipient lookup, which can publicly reveal sensitive information about him. Within this context, we propose an efficient, practical and scalable solution to guarantee the anonymity of recipients' nodes in ad-hoc networks.

key words: *anonymity, cryptography, scalability, ad-hoc network*

1. Introduction

The main characteristic of ad-hoc networks is that they consist of heterogeneous devices that temporarily join the network, in a spontaneous fashion, without the need of a network infrastructure such as access points. User terminals act as clients as well as routers of the network allowing any other device to connect to a remote terminal through a multihop path. In such decentralized and open networks, one of the main challenges is to protect the anonymity of the users and their locations. The particular architecture of multihop ad-hoc networks has a number of functional characteristics that are unique and specific to this type of network, and which prevent the direct adoption of anonymity schemes designed for classical wired networks. In particular, it is necessary to highlight that:

- The auto-management of the network is carried out through an open medium, susceptible to external attacks; active as well as passive. If the network is wireless (as they use to be), messages can be captured and heard by any user including malicious ones.
- Since any device can be participated in the forwarding of data across a multipath route, any compromised node can significantly harm the ad-hoc communication. Furthermore, the devices usually come without many advanced security components such as intrusion detection systems and trust managements mechanisms. These could prevent the network against internal attacks launched by compromised nodes.
- The network topology is dynamic and chaotic. In general, terminals are mobile nodes with a transitory and irregular state in the network.
- Network resources are limited. The terminals have low processing, memory and battery capacity.

The elements to provide anonymity in a network are subject anonymity, undetectability and unobservability of communications [1]. This paper focuses on subject anonymity, which means that an attacker cannot sufficiently identify the end-peers of a transaction within a group of subjects. Subject anonymity can be divided into two separate problems: source anonymity and recipients anonymity.

Source anonymity has typically being solved using pseudonyms, random strings, or one time identifiers. In multihop ad-hoc networks there is no much problem in hiding the source identity of a communication which can be established using bogus identifiers. Once the transmission is on and ready, the source can send its real identifier to the recipient over a ciphered channel.

Regarding recipients anonymity, hiding the identity of a destination is a challenging task. When a source wants to establish a communication with another node, it has to initiate a lookup for that particular recipient. In this lookup, all the nodes in the network have to collaborate to encounter the destination and construct a path towards it. Performing a target search while keeping its privacy preserved it is not a straightforward goal. Today's solutions are either computationally expensive and robust, or practical and vulnerable. In this letter, we briefly review the issues of present solutions and propose a novel and practical mechanism that provides recipient's anonymity for multihop ad-hoc networks. As shown in section 4, this protocol is efficient and can be scaled to large networks.

2. Related work

One-way trapdoor functions are normally used to establish the anonymous identification of a destination. A trapdoor is a function that can detrimentally compute in one direction and is difficult to compute in the other direction without any ”trapdoor” information. In the context of ad-hoc networks, when a source node looks for a recipient, it sends the identification information about the destination node, hidden in a trapdoor function so that only the legitimate receiver is allowed to recover it. The easiest way to implement a trapdoor function is by using public key cryptography. See for example the protocols SDAR [2], ARMR [3] and AnonDSR [4]. In these works the identity of the receiver is ciphered with its public key so that only the receiver will be able to successfully open the message. However, this solution has an overall high cost for the network since each node that gets the message has to compute a public key cryptographic operation to discover whether it is the intended recipient

[†] The author is with the Internet Interdisciplinary Institute, Universitat Oberta de Catalunya, 08018-Barcelona, Spain.

[††] The authors are with the Wireless Multimedia & Networking research group, Kingston University London, KT1 2EE, UK.

or not. Furthermore, considering that the route discovery is normally done via flooding mechanisms the final system load becomes significantly unsustainable.

In a similar way, protocols like ANODR [5] and ASR [6] make use of symmetric cryptography to hide the destination's identity. In this case, source and destination share a TESLA key [7]. Source node ciphers destination node's identity together with a random number using the symmetric key shared with the destination. The only node which is capable of opening the message and validate its identity it is the real recipient. Although these mechanisms are more efficient than those based on public key cryptography, the overhead across the network is still high. Moreover, they face the problem that source and destination must interchange a key before the ad-hoc network's establishment.

The ANAP protocol [8] proposes to identify the destination using the hash value of user's pseudonymous. However, the problem is how the source of a transmission can get the pseudonymous of the destination node. By assuming that such pseudonymous is public, attackers can precompute tables with the pairs pseudonymous-hashes. In this way, when a packet is captured in the network a destination node can be immediately discovered. On the other hand when the pseudonymous is secret, then using hashes does not provide enough strong security.

3. Proposed Methodology

In this section we discuss our work on how to preserve the anonymity of the recipient in multihop ad-hoc networks. We assume that the users of the network held a public key certificate. The certificate is the anchor to build the anonymous identifiers. The proposed protocol uses a two-level scheme to build a multihop path in which intermediate nodes can not identify the target of a communication, and a source can uniquely send its data to the intended destination peer.

To this end, a *high-level trapdoor* is applied first. This trapdoor is simple to evaluate and it points to a target group of nodes. Thereafter the nodes that belong to the target group need to evaluate a second and more costly trapdoor, called the *low-level trapdoor* to determine if they indeed are the intended destination of the communication.

In particular, the high-level trapdoor is implemented getting the truncated hash of users' public key to identify the users. Using a truncated hash provokes collisions, i.e., multiple users share the same truncated hash and so the identifier. Hence, the user identifier no longer points to a single user but a group of them. The benefit is that eavesdroppers and intermediate users that participate in a route discovery, have no means to find out who the packet is submitted to, although they discover that some users meet the criteria.

To uniquely identify the target of the transmission (the destination node), the scheme uses a public key ciphertext as the low-level trapdoor. The users that meet the terms of the high-level trapdoor, must try to decipher a byte sequence. However, only the intended recipient will be able to do this action successfully. Despite other protocols that use public key algorithms, our solution does not require every intermediate node to compute the costly decipherment operation. Thus the overhead of the protocol is not significantly high as we will see in section 4.

In the following, we describe in detail the process of looking up a destination and establishing a route between two ad-hoc nodes, according our protocol:

1. The source user, U_A, generates a random sequence r of 128 bytes and computes $salt = hasht_{sl}(r)$, where $hasht_{sl}$ is a hash function which output is truncated to the leftmost sl bits.
2. U_A prepares the high-level trapdoor function that points to the destination user U_B. To this end, it gets the destinations's public key pbk_B from a public directory, and computes $H = hasht_{hl}(salt \| pbk_B)$, with $\|$ the concatenation function and hl the length of the truncated hash output. Then, U_A builds a pseudonymous identifier pid_a_B for U_B in the following way: $pid\_a_B = salt \| H$.
3. U_A prepares the low-level trapdoor function ciphering the random sequence r: $pid\_b_B = E_{pbk_B}(r)$, where $E(.)$ is a public key cryptography algorithm.
4. Finally, U_A generates a lookup route request with the identifiers pid_a_B and pid_b_B.
5. When a user U_x receives a route request, it checks whether it is the target destination of the lookup by computing $H' = hash_{hl}(salt \| pbk_x)$. If $H' = H$, U_x has successfully opened the high-level trapdoor function and thus it is a candidate target.
6. Candidate targets attempt to open the low-level trapdoor function to verify whether they are the actual destination of a transmission or not. To this end, they compute: $a = D_{prk_x}(pid\_b_B)$, where $D(.)$ is a public key decipherment function and prk_x is their private key. Then, they check whether $salt = hash_{sl}(a)$ and if this matches, the user is the intended recipient.

4. Analysis

In this section we analyse the performance of the protocol in a particular network. We assume an open, wireless and multihop ad-hoc network which operates in a metropolitan environment where churn is one of the main characteristics. The number of potential nodes of the network can be significantly large (hundreds of thousands) since it comprises citizens, business people, tourists and any other user that joins and leaves the ad-hoc network periodically or just once. However, the number of the active users per second is within the range of $50 - 150$.

We configure the network with the length of the salt sl set to 10 bytes, and the length of the hash hl to 1 byte. The salt is used to avoid the in advance generation of tables that map users' keys to some particular identifiers, and to guarantee that a user is always identified with a different pid_a. Brute force attacks that compute the pid_a for all the users of the network cannot be performed in real time in large networks, hampering any profiling statistics of the

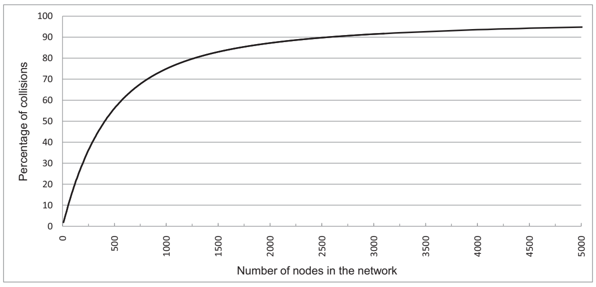

Fig. 1 Percentage of collisions in a network

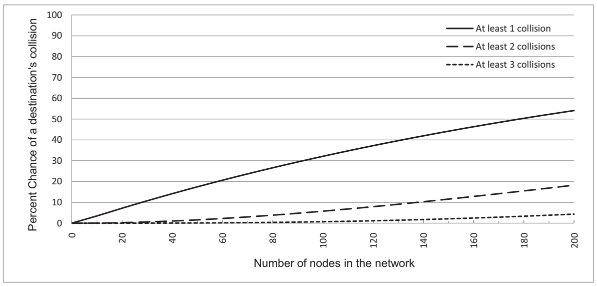

Fig. 2 Percent chance of collisions in a recipient's lookup

users. Nevertheless, even if the attacks succeed, a malicious user cannot know who the real destination of a message is, but only who belongs to the group of candidate recipients. The smaller the size of the truncated hash H, the bigger the group of destination candidates for a message and the higher the anonymity level. Contrary, bigger sizes of H increase the efficiency of the system since only a few users shall continue to the more costly low-level trapdoor phase of the protocol. Thus, there is a trade-off between the anonymity level and the efficiency of the system, which is determined by the size of the truncated hash.

We examined the scenario using Monte Carlo simulations. On the one hand, Fig. 1 shows the percentage of pid_a's collisions in networks of different sizes. The number of collisions in networks with more than 2,600 registered users is higher than 90%. Therefore, in public networks with lots of potential users, an attacker will not succeed to extract network information using traffic analysis.

On the other hand, Fig. 2 analyses the overhead of the system. It depicts the probability that when a route discovery protocol towards a particular recipient is executed in the network, the searched pid_a is shared among several users. The plot shows that in networks with 100 connected users, the probability of a collision is 32%, and the probability of 2 or more collisions is less than 6%. Thus, from the total 100 connected participants, only a few nodes which are not the intended receiver of the communication may have to exectue a resource-consuming [9], [10] public key decryption operation. The overhead of the system for a recipient's lookup is null with a probability of 74.65%, one decryption with 25.35%, two decryptions with 5.06%, and three or more decryptions with $< 1\%$. So, the protocol is efficient and does not suppose unaffordable costs for the ad-hoc network.

5. Conclusions

In this letter we have proposed a novel and simple method to provide recipients' anonymity in multihop ad-hoc networks and we have analysed its applicability to a particular scenario. Results show that the algorithm is efficient, robust, and scalable to large networks. Compared with anonymous methods that are solely based on hiding users identities with a hash, our proposal is more robust. In particular, identities change in every lookup so attackers cannot profile nodes in the network in addition to the fact that dictionary attacks are not possible since they usually require more time than the lifetime of a certain identity. Compared with solutions that are based on ciphering the identifier of the recipient with its own public key, our proposal is much more efficient since not all of the active nodes have to decipher a string to check whether they are the actual recipients.

Acknowledgments

This work has been developed during the visit of the first author to the Kingston University London by the Spanish Ministry of Science and Education under grant JC2010-0081. It has also been supported by the projects TSI2007-65406-C03-03 E-AEGIS and CONSOLIDER CSD2007-00004 ARES.

References

[1] A. Pfitzmann and M. Hansen, "A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management, v0.34." http://dud.inf.tu-dresden.de/Anon_Terminology.shtml, August 2010.

[2] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba, "An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks," Comput Commun, vol.28, no.10, pp.1193–1203, 2005.

[3] Y. Dong, T.W. Chim, V.O.K. Li, S.M. Yiu, and C.K. Hui, "Armr: Anonymous routing protocol with multiple routes for communications in mobile ad hoc networks," Ad Hoc Networks, vol.7, no.8, pp.1536–1550, 11 2009.

[4] R. Song, L. Korba, and G. Yee, "Anondsr: efficient anonymous dynamic source routing for mobile ad-hoc networks," Workshop on Security of Ad hoc and Sensor Networks, p.42, ACM, 2005.

[5] J. Kong and X. Hong, "Anodr: anonymous on demand routing with untraceable routes for mobile ad-hoc networks," Int. Symp. on Mobile ad hoc networking & computing, pp.291–302, ACM, 2003.

[6] B. Zhu, Z. Wan, M.S. Kankanhalli, F. Bao, and R.H. Deng, "Anonymous secure routing in mobile ad-hoc networks," IEEE Int. Conference on Local Computer Networks, pp.102–108, Citeseer, 2004.

[7] A. Perrig, R. Canetti, J.D. Tygar, and D. Song, "The tesla broadcast authentication protocol," RSA CryptoBytes, vol.5, no.2, pp.2–13, 2002.

[8] T. Ciszkowski and Z. Kotulski, "Anap: Anonymous authentication protocol in mobile ad hoc networks," CoRR, vol.abs/cs/0609016, 2006.

[9] H. Rifà-Pous and J. Herrera-Joancomartí, "Cryptographic Energy Costs Are Assumable in Ad Hoc Networks," IEICE Trans. Inf.& Syst., vol.92, pp.1194–1196, 2009.

[10] H. Rifà-Pous and J. Herrera-Joancomartí, "Computational and energy costs of cryptographic algorithms on handheld devices," Future Internet, vol.3, no.1, pp.31–48, 2011.