Computer Communications 33 (2010) 1094–1107

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

ECGK: An efficient clustering scheme for group key management in MANETs

K. Drira [a], H. Seba [b], H. Kheddouci [a,*]

[a] Lab. LIESP, University Lyon1, Villeurbanne, France
[b] Lab. LIESP, University Lyon1, IUT, Dépt informatique, Bourg en Bresse, France

ARTICLE INFO

Article history:
Received 2 September 2008
Received in revised form 4 February 2010
Accepted 5 February 2010
Available online 19 February 2010

Keywords:
Group key management
Mobile ad hoc networks
Trust
Clustering

ABSTRACT

Mobile Ad hoc NETworks (or MANETs) are flexible networks that are expected to support emerging group applications such as spontaneous collaborative activities and rescue operations. In order to provide secrecy to these applications, a common encryption key has to be established between group members of the application. This task is critical in MANETs because these networks have no fixed infrastructure, frequent node and link failures and a dynamic topology. The proposed approaches to cope with these characteristics aim to avoid centralized solutions and organize the network into clusters. However, the clustering criteria used in the literature are not always adequate for key management and security. In, this paper, we propose, a group key management framework based on a trust oriented clustering scheme. We show that trust is a relevant clustering criterion for group key management in MANETs. Trust information enforce authentication and is disseminated by the mobility of nodes. Furthermore, it helps to evict malicious nodes from the multicast session even if they are authorized members of the group. Simulation results show that our solution is efficient and typically adapted to mobility of nodes.

1. Introduction

A mobile ad hoc network is a set of wireless nodes which cooperatively form a network independent of any fixed infrastructure or centralized administration. In particular, a node communicates directly with nodes within wireless range and indirectly with all other nodes using a dynamically computed, multi-hop route via the other nodes. Nodes are free to move randomly and organize themselves arbitrarily. So, the network topology may change rapidly and unpredictably. Because of this flexibility, MANETs are intended to play a fundamental role in several emerging group based applications such as spontaneous collaborative activities, emergency communications and sensing applications. However, the flexibility of these networks associated with the vulnerability of wireless connections require a growing need in the security of data. Using wireless connections makes ad hoc networks vulnerable to listening, creation, modification and non-allowed destruction of data, denial of service, repudiation and traffic diversion. So, assuring a certain level of security in these networks is a requirement for the large deployment of group communication in these environments. However, for MANETs, the issues and problems of security are complex for the following main reasons:

1. Absence of a fixed infrastructure: MANETs do not require any pre-deployed infrastructure or any trusted centralized authority. Thus, security mechanisms based on centralized control is unrealizable. This is not due only to the difficulty of maintaining such an entity but especially it will be a point of weakness vulnerable to attacks.
2. Topological changes: mobility and dynamism make the topology of the network changes continuously. Thus, security protocols must be adaptive to these changes.
3. Network partitions and merges.
4. Network delay and frequent node and link failures.

Basic security mechanisms are authentication, access control, integrity verification and confidentiality. In this paper, we focus on group communication confidentiality in the environment of MANETs. Group communication confidentiality prevent non-group members from reading data exchanged within a secure communication session of the group. This confidentiality requires establishing and maintaining a common key between group members. This key, called *group key* or *traffic encryption key* (*TEK*) serves to encrypt/decrypt message exchange within the group. According to the model of [15,16], the management of the group key must meet the following properties:

1. Key secrecy which guarantees that it is computationally infeasible for a passive adversary to discover any group key.
2. Key independence which guarantees that a passive adversary who knows any proper subset of group keys cannot discover any other group key.

[*] Corresponding author.
E-mail addresses: kdrira@bat710.univ-lyon1.fr (K. Drira), hseba@bat710.univ-lyon1.fr (H. Seba), hkheddou@bat710.univ-lyon1.fr (H. Kheddouci).

0140-3664/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2010.02.007

3. Weak forward secrecy[1] which guarantees that new keys must remain out of reach of former group members.
4. Weak backward secrecy which guarantees that previously used group keys must not be discovered by new group members.

The two last requirements, generally called forward and backward secrecy in group key management literature, imply that a rekeying process refreshes the group key after each group membership change, i.e. join or leave operation. Thus, rekeying may induce an important communication overhead especially in the case of frequent group membership change. Consequently, one of the appreciated properties in any rekeying mechanism is its *1-affects-n* scalability [27] which measures how well it scales to large and dynamic groups. To enhance *1-affects-n* scalability, some group key management solutions propose to organize the secure group according to a logical topology mainly a tree or a cluster-based structure. Using clusters with different local traffic encryption keys reduces the impact of the key updating process (*1-affects-n*), but needs decryption and re-encryption operations between clusters. Furthermore, these schemes may generate excessive computation and communication overhead to maintain their virtual topology and are not always appropriate for key management or security. In this paper, we do not introduce a novel keying tool. However, we focus on the clustering criteria of keying schemes. We propose a self-organizing clustering scheme that relies on trust information available at the level of each node to organize the group into clusters. Then, we describe a combination of existing cryptographic solutions that take advantage of the proposed trust-based clustering architecture. As trust is related to security, it is a more adapted clustering criterion for group key management in MANETs. It generates stable clusters and helps to avoid malicious group members. In fact, a MANET can exist only if its nodes cooperate between them in relying packets and rendering services to each other. However, malicious nodes may exist and attempt to disturb or corrupt the application. For applications such as secure multicast this is not acceptable and malicious nodes must be evicted from the group session. To detect malicious behaviour, nodes must rely on their own appreciation and means and eventually on recommendations from trusted nodes. With trust as a clustering criterion, malicious nodes are automatically excluded from the group session even if they are authorized nodes.

The rest of the paper is organized as follows. In Section 2, we classify and discuss group key management protocols proposed in the literature for MANETs and we motivate our work. In Section 3, we give an overview of the proposed architecture, and then, we present the underlying clustering and keying schemes. In Section 4, we discuss and evaluate the performances of the proposed solution and compare it with existing ones. Section 5 brings our remarks concluding the paper.

2. Related work and motivation

Group key management protocols for secure multicast in MANETs can be classified into two approaches: the flat approach and the topology-oriented approach. In the following, we present each approach and discuss its strengths and weaknesses.

2.1. Flat approach

In this approach, there is no prior organization of the group members and all group members share a common *TEK*. The management of this single key is centralized at a unique server or distributed among all group members. In [4], the authors propose *CRTDH* (Chinese Remainder Theorem and Diffie–Hellman based scheme for secure group communication). *CRTDH* is a contributory key management protocol. This means that the common TEK is computed as a function of information contributed by all the participants in the group. *CRTDH* needs two rounds of broadcasts to establish the *TEK*. It assures no central trusted authority and computation is equally distributed among all members. However, it is broadcast-based and consequently not scalable. Furthermore, it suffices that a member fails or a message being lost for blocking the whole protocol or causing members to compute different *TEKs*. In [6], the authors propose a two round key agreement protocol which we note *TRP* in the rest of the paper. In *TRP*, the initiator of the protocol becomes the group leader. The leader begins by broadcasting an *INIT* message to launch the key computation process. Then, the protocol needs two rounds:

Round 1: Each participant i responds to the *INIT* request by choosing a random secret r_i and sending its blinded version g^{ri} to the initiator.

Round 2: The group leader ℓ raises each joining member's blinded secret to its secret r_ℓ and broadcasts them along with the original contributions to the group, i.e. it sends $\{g^{r_i}, g^{r_i r_\ell}\}$ for all $i = 1, n$ and $i\ \ell$, where n is the number of participants.

Then, each member i checks if its contribution is included correctly, removes its secret r_i from $g^{r_i r_\ell}$ to get g^{r_ℓ} and computes the group key:

$$TEK = g^{r_\ell} * \prod_{i=1,n}^{i\neq\ell} g^{r_i r_\ell} = g^{r_\ell\left(1+\sum_{i=1,n}^{i\neq\ell} r_i\right)}.$$

TRP requires less message overhead than *CRTDH*. However *TRP* suffers from the problem of the central point of failures and is vulnerable to message delay and node failures. In [1], the authors propose an authenticated version of *TRP* by adding a third round to the initial protocol. This round is used for the authentication of the members. In [41], The authors propose *GKMPAN*, a probabilistic approach to group key management. *GKMPAN* assumes the existence of a key server for initial key distribution and sending authenticated group key updates to the nodes. In *GKMPAN* initialization, all nodes in the ad hoc network are given a certain number m of keys out of a big pool of ℓ keys. The set of key indexes that a node possesses is deterministic and can be publicly computed from its unique node *ID*. This allows any node to be able to compute which keys a particular node has and communicate securely with him. The speed of the key distribution as well as security depends on the number of locally stored keys m and on the size of the key pool ℓ. If m is very small compared to ℓ then the probability of having a shared key with any neighbour is small. Thus, any two nodes that do not have a shared key need to find one or more intermediate node(s) to be able to communicate securely. In [28], the authors propose a distributed key management protocol based on threshold cryptography [33]. The particularity of this is approach is the management of compromised nodes. However, it relays on broadcasts and consequently is not scalable.

The flat approach suffers from the *1-affects-n* problem, where a single group membership change (join or leave) results in a rekeying process that disturbs all group members. Moreover, most protocols in this approach need a central server. So, they are neither scalable nor fault-tolerant.

[1] Note that Kim et al.'s definition of (weak) forward secrecy [15] is in conflict with the commonly used meaning of the term forward secrecy. This term commonly designates perfect forward secrecy (PFS) [10].