Computer Communications 36 (2013) 1159–1171

Contents lists available at SciVerse ScienceDirect

Computer Communications

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

Trust-based security for the OLSR routing protocol

Asma Adnane [a,*], Christophe Bidan [b], Rafael Timóteo de Sousa Júnior [c]

[a] School of Computing and Mathematics, University of Derby, United Kingdom
[b] SUPELEC, SSIR Team, Avenue de la Boulaie, Cesson-Sévigné 35510, France
[c] Electrical Engineering Department, University of Braslia, Braslia, DF 70910-900, Brazil

ARTICLE INFO

Article history:
Received 23 March 2011
Received in revised form 1 November 2012
Accepted 3 April 2013
Available online 20 April 2013

Keywords:
Trust
Ad hoc networks
Security
Routing protocol
OLSR

ABSTRACT

The trust is always present implicitly in the protocols based on cooperation, in particular, between the entities involved in routing operations in Ad hoc networks. Indeed, as the wireless range of such nodes is limited, the nodes mutually cooperate with their neighbors in order to extend the remote nodes and the entire network. In our work, we are interested by trust as security solution for OLSR protocol. This approach fits particularly with characteristics of ad hoc networks. Moreover, the explicit trust management allows entities to reason with and about trust, and to take decisions regarding other entities.

In this paper, we detail the techniques and the contributions in trust-based security in OLSR. We present trust-based analysis of the OLSR protocol using trust specification language, and we show how trust-based reasoning can allow each node to evaluate the behavior of the other nodes. After the detection of misbehaving nodes, we propose solutions of prevention and countermeasures to resolve the situations of inconsistency, and counter the malicious nodes. We demonstrate the effectiveness of our solution taking different simulated attacks scenarios. Our approach brings few modifications and is still compatible with the bare OLSR.

1. Introduction

Today, mobile Ad-hoc networks (MANETs) are a major element of the business environment, allowing wireless devices such as cell phones, laptops, and PDAs to provide mobility to users and enable them to keep in constant contact with others. Technically, MANETs are self-organized wireless mobile networks that do not rely on any centralized administration or fixed network infrastructure. The cooperation between the mobile devices allows to provide the network services. More precisely, each device participates in routing service: a communication between distant devices can be established only if intermediate devices cooperate by forwarding the messages they receive. Thus, each device of a MANET has to maintain a local routing table that determines the next hop toward all other devices. The routing table is managed using an *ad hoc routing protocol* (for example: OLSR, AODV).

Many ad hoc routing protocols have been developed for ad hoc networks [1]. Roughly speaking, they can be classified according to the type of route discovery: reactive and proactive. In reactive protocols, e.g. AODV (Ad hoc On-demand Distance Vector), the routing request is sent on-demand: if a device wants to communicate with another, then it broadcasts a route request and expects a response from the destination. Conversely, proactive protocols update their routing information continuously in order to have a permanent overview of the network topology (e.g. OLSR [2]).

The security of MANET is a major challenge, and the self organization characteristics of MANET imply that traditional security solutions are often inadequate. In other words, any device participating to the routing service can easily attack the MANET either by disrupting any communication with which it is involved, or by compromising the routing tables of other devices. It is important to point out that these two attacks affect the network at two different levels: the first one is the message routing, whereas the second is the ad hoc routing protocol.

As regards the security of the message routing, the classical approach consists in using reputation systems to detect misbehavioral devices (e.g. devices that do not forward the messages). Concerning the security of the ad hoc routing protocol, most research assumes that as long as the messages containing the topological information are secured, the routing tables cannot be compromised. Our point of view is that such an approach is not sufficient since in any ad hoc routing protocol, a device can easily compromise the routing tables by sending incorrect topological information in secured messages. Thus, solutions that guarantee the correctness of the routing tables have to be proposed.

Assuming that any protocol is based on implicit trust relations (as demonstrated in [3] and Section 4), we assert that such trust relations can be used by each device to assess the expected correct

* Corresponding author.
E-mail addresses: a.adnane@derby.ac.uk (A. Adnane), christophe.bidan@supelec.fr (C. Bidan), desousa@unb.br (R.T. de Sousa Júnior).

0140-3664/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.comcom.2013.04.003

behavior of the other devices, and also to reason about the correctness of its routing table. In this article, we illustrate this through the OLSR (Optimized Link State Routing protocol [2]) protocol. We summarize our contributions to the analysis of the implicit trust within OLSR, and to the trust-based reasoning and countermeasures for securing OLSR nodes.

The paper is organized as follows. In Section 2, related works on security in ad hoc networks are summarized. In Section 3, we introduce the concept of trust management and trust specification language. An overview of OLSR is presented in Section 4. In Section 5, we introduce the analysis of implicit trust in OLSR, then we present trust reasoning developed to secure OLSR in Section 6. Countermeasures concerning the attacks against the basic operations in OLSR, and a method of distribution of information about trust relation to prove the attack and prevent distant nodes in the network are detailed in Section 7. Finally, we conclude this paper by presenting simulation results and our future works.

2. Related works

As we pointed out before, the routing service in MANET can be attacked either by disrupting the message routing or by compromising the routing tables. In the former case, the main concern is to protect against misbehaving devices, and especially selfish devices (i.e. devices that do not properly forward messages). The traditional solution consists in forcing the devices to collaborate. One of the early works on collaboration is presented by Marti et al. [4]. The authors introduce the ***watchdog*** and ***pathrater*** mechanisms. Basically, the *watchdog* mechanism is used by each node to monitor the behavior of its neighbors. Using the information of the *watchdog*, the device can locally compute a rating for each of its neighbor, and when this rating is below a given threshold, it uses the *pathrater* mechanism to compute another path avoiding misbehaving devices. Thus, selfish devices are detected and not used anymore. Note that Marti et al. do not allow each device to notify other devices when a malicious device is detected.

Today, a major part of the research works on collaboration in MANET has been inspired by this previous work, especially by using the *watchdog* mechanism to build a reputation system [5,6,8–10]. For example, in [6], the authors propose a collaboration system called CONFIDANT (Cooperation Of Nodes: Fairness In Dynamic Ad-hoc NeTworks) based on a reputation system. More specifically, each device uses a *watchdog* mechanism to monitor and report the message routing behavior of other devices. Given the observed and reported behavior, each device computes a reputation score for each device to detect misbehaving ones. The detection of misbehaving devices leads to their isolation. Thus, the devices are constrained to cooperate so as not to be isolated. Later, the same authors extended their cooperation system in order to deal with devices that deliberately send false reputation scores [11].

Similarly, Michiardi et al. propose a cooperation enforcement mechanism, called CORE (COllaborative REputation) [7]. Basically, CORE uses a *watchdog* mechanism to allow each device to monitor its neighbors. Based on its own observation as well as the scores provided by other devices involved in the current operation, a device can compute a reputation score for each of its neighbor, this score represents the degree of cooperation. Then, when a selfish device is detected, it is gradually denied network services. Thus, a device cooperates, otherwise it can no longer use the MANET. Notice that CORE allows to rehabilitate selfish devices if they behave correctly again.

In contrast, *Buttyan and Hubaux* have proposed the collaboration mechanism, called Nuglets [12] adopting a completely different approach. They introduce a virtual currency called *nuglet*. Each node has to pay to use network services (forwarding its data), and must be paid for offering services to other nodes. Thus, selfish nodes will finish their nuglets and can no longer send packets. The drawback of this method is that the nuglets are managed by a centralized entity.

In brief, collaboration systems are based either on reputation systems monitoring the neighbors' behavior to detect misbehaving devices (e.g. selfish nodes), or on a virtual currency to enforce the nodes to collaborate. However, in both cases, the solutions implicitly assume that the routing tables are correct.

However, other works propose reputation systems that can also be applied for securing routing protocol by monitoring node behavior, in order to verify that nodes respect the routing protocol specification. For example, CORE [7] is suggested as a generic mechanism and can be integrated with any network service. Precisely, Meka et al. [10] propose trust-based reputation model for AODV. Reputation is calculated according to the degree of participation in the routing protocol and the information it provides about the network topology. Reputation system was also used as a security method to perform trust-based multi-path routing [13,14]. Thus, they do not allow to detect a malicious node which would be a normal behavior in terms of message routing, but a misbehavior with respect to the ad hoc routing protocol.

To deal with compromised routing tables, the major part of the research works is based on cryptography to secure the messages containing the topology information used to calculate the routing tables. The underlying assumption is that authenticated devices are known to behave correctly: when some topology information is authenticated, it is correct/trusted. In other words, the main concern of these research works is to keep *unknown* devices (i.e. intruders) out of the MANET, thus stopping such devices from modifying/falsifying topology information sent by authenticated devices.

Many research works have proposed security solutions for reactive ad hoc routing protocol based on cryptography [15–17]. For example, Zapata and Asokan [16] have proposed a secured version of AODV. The authors present two mechanisms to secure the AODV messages: digital signatures to authenticate the non-mutable fields of the messages, and hash chains to secure the hop count information (the only mutable information in the messages). Ariadne [17] is another secured protocol based on DSR and TESLA: the authors assume that a shared secret key is distributed for a group of trusted nodes using TESLA and that the nodes are synchronized.

Similarly, some research works have been undertaken for the security of proactive ad hoc routing protocols based on cryptography [19,20]. For example, Adjih et al. [20] have proposed a secured version of OLSR called SOLSR. Their approach is based on the signature and time-stamp of each OLSR control message. A signature is generated for each control message and sent with the message to prevent malicious nodes to modify or falsify topology information. In addition, a time-stamp is associated with each signature to estimate the freshness of the message. However, This solution does not ensure the correctness of the information provided by authenticated nodes, and assumes that any authenticated node is a trusted node without any verification. The solution of Hafslunf et al. consists in signing the OLSR packets. In our view, this latter approach is not adapted since a corrupted node can easily modify the content of a *TC* message before generating the signature of the new packet which will contain it.

Omar et al. [18] propose a fully distributed public key certificate management system based on trust graphs and threshold cryptography. It allows nodes to issue public key certificates, and to authenticate the other nodes via certificates chains without trusted authority. The proposed solution uses the threshold cryptography to resist against false public keys certification. The initialization