Abstract—This paper focuses on the security problem in unreliable wireless ad hoc networks when involving network coding. A typical example of misbehaviors by bad nodes is presented to show the misbehavior propagation problem of network coding. To reduce the influence of bad nodes, we propose to combine network coding with trust scheme. In our trusted network coding, a trust model is established according to the behavior observation procedure. We take COPE (a typical network coding-based protocol) as a baseline to implement trusted network coding. Simulation results show that our trusted network coding performs much better than conventional network coding without trust scheme.

I. INTRODUCTION

Network coding theory has been widely used in many areas of communication since it emerged in 2000 [1]. R. W. Yeung et al. present the basic theory of linear network coding in [2]. The key idea of network coding is to allow the intermediate node in a network to encode the information from input links. Network coding has been proved to have the advantages of increasing network throughput, saving energy, enhancing robustness and load balancing [3][4][5].

Most of the former theoretical and experimental results of network coding are based on some idealized assumptions including no transmission error and no malicious node. Nevertheless, in practical wireless networks, these assumptions are usually unavailable especially in wireless ad hoc networks, which will limit the promised potential of network coding. As a result, scholars pay much attention to applying network coding theory to practical networks in recent years [6][7][8].

In wireless ad hoc networks a node can join and quit the network at anytime, which results in a dynamic network topology. Moreover, all nodes are in an equal level and there is no central device to manage the whole network. These conditions make it difficult to judge the potential malicious nodes. Dong et al. analyze these threats and challenges for network coding in [8] and present a wide range of attacks on different stages of network coding based protocol. In network coding system some nodes performs data encoding and decoding process. A coded packet contains the information of several packets, thus bad nodes’ attacks on the coded packets are more serious than in a conventional network without coding. We take an example to explain the error propagation and loss propagation problem brought by network coding in Section II of this paper. To fight against these challenges, a new security strategy specialized for network coding is needed urgently.

Previous work related to the security of network coding mainly focuses on theoretical analysis. In [9], a necessary and sufficient condition for the feasibility of constructing a secure linear network code is derived. R. W. Yeung et al. extends the analysis to a multi-source environment by studying the algebraic structure of network coding in [10][11]. Recently researchers proposed some approaches to fight against some specific attacks. Reference [12] presents a low-complexity cryptographic scheme that exploits the inherent security provided by random linear network coding. References [13] and [14] focus on the packet pollution attacks on network coding. Reference [15] proposes a resilient network coding scheme to tolerate Byzantine adversaries.

In this paper we aim to investigate how malicious nodes influence the performance of network coding and how to reduce the bad influence in a general approach. Inspired by the trust scheme introduced in [16], we propose to combine trust scheme with network coding. Trust is defined as the belief level that one node can put on another node for a specific action based on previous observations on the node’s behaviors [16]. With trust scheme a node can judge whether other nodes are reliable according to its previous observation. Unlike former research work, our solution doesn’t focus on a specific attack, but provides a framework to distinguish malicious nodes and make correspondent reactions according their misbehaviors. We make implementation of trusted network coding on a typical network coding scheme COPE [6]. The basic idea of our trusted COPE contains three core parts: trusted listening, trusted coding and trusted routing. The detailed implementation is presented in Section IV. To show the advantages of trusted COPE, we make some simulations and give the results in Section V.

There are three main contributions in this paper:

- It analyzes the error propagation and loss propagation problem brought by network coding in unreliable wireless ad hoc networks.
- It proposes to apply trust scheme to network coding based routing protocol to reduce the influence of bad nodes.
- Finally, this paper presents the implementation details of combining network coding and trust scheme in COPE.

978-1-4244-5637-6/10/$26.00 ©2010 IEEE
II. NETWORK CODING IN UNRELIABLE WIRELESS NETWORKS

In practical wireless ad hoc networks there exist some unreliable actions caused by many factors such as selfish behaviors, active malicious attacks, lack of energy and even unintentional misconfiguration. These unreliable actions include packet loss, data modification and transmission delay. We call the node with unreliable actions as a “bad node”. Bad nodes may cause fatal influence on network coding scheme. In [8], J. Dong et al. analyzed the threats and challenges for wireless network coding and presented simulations of Pollution, Over-coding, Under-decoding and Drop-Data attacks. We show how bad nodes influence network coding by the following examples.

As shown in Fig.1, the simple scenario contains 6 wireless nodes. The edges between nodes represent the data stream. We assume the transmission pattern is: A and B transmit packets to C and D respectively, E and F are relay nodes. Since the data transmission is in a broadcast manner, node C can overhear the data sent by B, and D can overhear the data sent by A. Denote the packet from A and B as a and b respectively. With network coding scheme, node E can XOR a and b to get a coded packet denoted by a + b. Node F receives a + b and forward it to C and D. Therefore, C can decode packet a with b and a + b, D can decode packet b in the same way.

Now we analyze the influence of bad nodes in an unreliable network situation. If node E tampers the packet from A or B, then the coded packets at node E will contains erroneous information which will cause both C and D decoding erroneous packets. However without network coding the misbehavior of node E will only influence one packet at a time. Therefore, network coding scheme spreads the influence of error, which is called “error propagation”. Similarly, if node E loses the coded packet a + b, then C can not decode a and D can not decode b. That is, one coded packet loss will influence two receivers. The more receivers, the wider influence a coded packet loss will cause. This phenomenon is referenced as “loss propagation”. Misbehavior propagation will decrease the performance of network coding sharply, especially decreasing the network throughput.

To solve the problem brought by bad nodes, we propose to apply trust scheme in the network coding based protocol. The following section will introduce the basic concept of trust model that we used in this paper. Our model is mainly based on the pioneer work of [16] and [17]. An information theoretical framework of trust modeling and evaluation is given in [16] and reference [17] presents a trust management framework for mobile ad hoc networks.

III. TRUST MODEL

A. Definition of Trust

Trust: Trust is the belief level that one node can put on another node for a specific action based on previous direct or indirect observations on behaviors of this node.

Trust property: The factors and actions that influence the performance of network coding based routing includes tampering data, losing packet, delay and energy, which can be called as “trust properties”.

Trust vector: We define trust vector for a node as: $V_T = \{T_1, T_2, ..., T_n\}$, where $T_i \in [0, 1]$, $i \leq n$. $T_i$ represents the trust value for property $i$ and takes a value ranging from 0 to 1. Higher trust value means more reliability for correspondent property.

Weight vector: Trust weight is used to mark the importance of trust properties, which is denoted as: $W_T = \{W_1, W_2, ..., W_n\}$, where $\sum_{i=1}^{n} W_i = 1$. $W_i$ is the weight for property $i$.

Weighted trust: Weighted trust is the overall trust of a node considering all properties, which is defined as

$$T_w = V_T \cdot W_T = \sum_{i=1}^{n} W_i \cdot T_i. \quad (1)$$

Path trust: For a routing path with $k$ nodes, the path trust is defined as

$$T_{path} = \prod_{i=1}^{k} T_{w,i}. \quad (2)$$

B. Trust Observation

We use a simple trust observation scheme introduced in [17]. Direct observation trust value is:

$$T_{obs} = \frac{k + 1}{N + 2} \quad (3)$$

where $N$ is the total number of observations, $k$ is the number of misbehaviors among $N$ observations.

For those properties which can not be counted by action times, we can judge the misbehavior by comparing the property value with a default threshold. The threshold can be set according to practical network environment. For some special attacks of malicious nodes, the observation process relies on the attack detection technique.

C. Trust Evaluation

Through observation, every node has the trust information for each of its neighbor nodes. All nodes broadcast the message containing their observation trust information to its neighbors periodically. After one period all nodes update their trust table by merging its own direct observation and the
indirect observation of its neighbors. The synthetical trust value $T_{syn}$ is calculated as follows:

$$T_{syn} = \sum_{i=1}^{n} w_i \cdot T_{obs,i}.$$  

(4)

$w_i$ is the weight for each observation trust. In general, a node’s own direct observation should have higher priority over indirect observations. Some researchers proposed to use “observation reputation” as the weight for indirect observations [18]. In this paper we focus on how trust scheme works for network coding, thus we don’t apply the complex trust evaluation method. Instead we set equal weights for all neighbors’ observation weight. Then

$$T_{syn} = w_{dir} T_{dir} + \frac{1 - w_{dir}}{n} \sum_{i=1}^{n} T_{obs,i}$$  

(5)

where $T_{dir}$ is the direct observation trust value, $w_{dir}(w_{dir} \in [0, 1])$ is the direct observation weight.

D. Trust Management

In our trust scheme, each node maintains a trust table. As shown on Fig.2, the entry of trust table contains the node id and its correspondent trust vector. For each property, direct observation trust and the synthetical trust are stored. Direct observation trust is periodically broadcasted to neighbors and synthetical trust is also updated periodically according to the neighbors’ feedback message. Note that the calculation of path trust uses synthetical trust value.

IV. TRUSTED NETWORK CODING

A. Baseline Protocol

There are many network coding based routing protocols such as COPE [6] and MORE. COPE is the first practical implementation of inter-flow network coding and MORE is the first intra-flow network coding scheme. For different network coding schemes we need to make correspondent improvements according to their coding and decoding strategy. In this paper we implement trust scheme on COPE in wireless ad hoc network. We choose COPE as a baseline for the following reasons:

- COPE is a typical implementation of network coding in practical networks.
- COPE addresses the common case of unicast and can also be used for multicast.
- COPE has robustness in practical environment with dynamic and bursty flows.

In wireless networks, transmission is in a broadcast manner. Therefore a node may overhear the packets from its neighbors even when they are not the intended recipients. In COPE each node keeps a buffer to store the overheard packets and share the packets information with its neighbors. In this case a node knows which packets its neighbors need and what they already have.

To introduce the key idea of COPE we present a typical scene (Fig.4) used in [6], which is called a “coding cell”. The coding cell is just a small transmission pattern formed by two or more crossing routes in wireless networks. Each node maintains a buffer to store the overheard and received packet in it. The next hops of packet $P_1, P_2, P_3$ are node $A$, $B$ and $C$ respectively.

The coding rule in COPE is:

To transmit $n$ packets, $p_1, ..., p_n$, to $n$ recipients, $r_1, ..., r_n$, a node can XOR the $n$ packets together only if each intended recipient $r_i$ has all $n - 1$ packets $p_j$ for $j \neq i$.

According to this rule, the center node $D$ chooses $P_1, P_2, P_3$ to make XOR operation and broadcast the coded packet $P_c$. Then $A$ can decode $P_1$ with $P_c, P_2$ and $P_3$. $B$ and $C$ decode the packets they need in the same way.

B. Trusted COPE (T-COPE)

The key idea of our trusted COPE is to prevent the bad nodes from the coding and decoding procedure through trust scheme. To apply trust scheme in COPE, the following improvements are made:

1) Trust Listening: In COPE, opportunistic listening is used to exchange packet state among neighbors. In trusted COPE, we add extra data of trust information to the hello packet in order to update the trust table of each node. In this way a node knows whether its neighbors are reliable and make correspondent actions to the overheard packets. For example, if a node always modifies the packet, its trust value for data reliability is low. Then, its neighbors check the buffer and directly delete the packets overheard from it, which avoid the error propagation.

2) Trusted Coding: In COPE, center node (e.g. node $D$ in Fig.3) in a coding cell encodes the packets without...
checking them. However, due to the existence of bad nodes, not all packets contain the proper information. To avoid the error propagation problem, the center node needs to check its own trust table to get the trust vector for the packets’ last hop. If the “data modification” trust value in the vector is under a threshold, it directly drops the packet.

3) **Trusted Routing:** In this paper COPE is based on AODV protocol. To prevent misbehavior propagation, we apply trust scheme in the routing. Each node knows the trust information of its neighbors through direct and indirect observations. When a source needs to find the route to a destination, it broadcast RREQ packets. Those intermediate nodes who receive the RREQ add entries in the routing table for the path back to source and forward the RREQ. When the destination receives the RREQ, it sends RREP along the path established. With trust scheme, an intermediate node adds trust information of its last hop in the RREP packet. When the source receives the RREP it obtains the trust vector of all intermediate nodes in different paths and calculate the weighted trust with its local weight vector. Then the source choose the path with the highest path trust as the transmission route.

C. Packet Format

This subsection presents the packet format detail of implementing trusted COPE. We add some extra information in the header of hello packet and coded packet.

(a) **Hello Packet**

To enable the function of trust listening, we add extra information in the packet header of a hello packet. As shown in Fig.4, REPORT_TIME records the neighbor state report time of a hello packet. If a node receives several hello packets from the same neighbor, it chooses the latest one according to the report time and drops others. There are two extra data blocks:

- Neighbor State: neighbor state information, which contains all the neighbors’ ID and its trust vector. A node updates its trust table through this information.
- Buffer State: This block contains all the packet ID in the buffer of a node. With this information a node knows its neighbors’ packet state and makes coding decisions according to it.

(b) **Data Packet**

There are two kinds of data packets: normal packet and coded packet. Coded packet need to include the coding information in the packet header. As shown in Fig.5, two extra blocks are added:

- Coding Information: this block contains the information of coded packets which helps to decode the coded packets and the further transmission. CODED_NUM is the number of packets that has been encoded in a packet.
- Buffer State: this block has the same function as the buffer state of hello packet header (Fig.4). Adding buffer state information in data packet makes it much faster to update the neighbor state information.

V. SIMULATIONS

A. Simulation Scenario

In this section we present simple simulations to show the advantages of T-COPE. All simulations are made on NS2 platform. The basic network parameters are shown in TABLE I. All nodes are set in promiscuous mode, thus they also receive packets not addressed to them. We just consider two typical misbehaviors of bad nodes: dropping packet and tampering data. Each bad node drops or tampers the data packets passing through it in a probability of 0.8. All the bad nodes are distributed randomly in the scenario and we don’t know which data flow they will influence, therefore we take the total end-to-end throughput of all flows and the successful delivery ratio as the main evaluation metrics.

<table>
<thead>
<tr>
<th>Network scale (m²)</th>
<th>1200*1200</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node number</td>
<td>100</td>
</tr>
<tr>
<td>Node mobility (m/s)</td>
<td>0-20 (at random)</td>
</tr>
<tr>
<td>Traffic pattern</td>
<td>CBR flow</td>
</tr>
<tr>
<td>Number of flows</td>
<td>60</td>
</tr>
<tr>
<td>Initial trust value</td>
<td>0.5</td>
</tr>
</tbody>
</table>

B. Simulation Analysis

In simulation 1, there are 30 bad nodes distributed randomly. Fig.6 plots the simulation results of total throughput as a function of time. We can see from the figure that the two throughput curves are almost coincident at the beginning.
of the simulation, which is because the trust observation process takes some time. As the simulation goes on, the total throughput of T-COPE is higher than COPE. T-COPE increases the throughput by 12% on average.

In simulation 2, we aim to compare the average packet delivery ratio of COPE and T-COPE, as a function of bad nodes number which varies from 0 to 50. Higher delivery ratio means a higher throughput. As shown on Fig.7, three observations are made. First, COPE’s delivery ratio decreases sharply with the increment of bad nodes, which shows the vulnerability of COPE. Second, T-COPE increases the delivery ratio evidently compared with COPE. Third, the gap between the two curves are increasing with the bad nodes’ number, which shows that T-COPE have higher advantages over COPE with more bad nodes.

All simulation results above show the advantages of our trusted COPE in terms of network throughput. In future research, we will extend our simulation to a more complex environment with more misbehavior types.

VI. CONCLUSIONS

In this paper we present an analysis of network coding in unreliable wireless network and propose to use trust scheme in network coding to reduce the bad influence of bad nodes. We take COPE as a baseline to explain how to implement trusted network coding. Our trusted COPE contains three parts: trust listening, trusted coding and trusted routing. Simulation results show that trusted network coding significantly outperforms normal network coding without trust scheme in terms of network throughput. Our future research will focus on the performance analysis of trusted network coding for different network environments with more types of attacks and multiple service patterns.

ACKNOWLEDGMENT

This work is supported by National 863 Projects of China (2009AA01Z205), Fund of National Laboratory (P080010), Natural Science Foundation of China (60872010, 60972016), Program for New Century Excellent Talents in University (NCET070339) and Funds for Distinguished Young Scientists of Hubei, China (2009CDA150).

REFERENCES