Zigzag: Partial Mutual Revocation Based Trust Management in Tactical Ad Hoc Networks

Xin Chen*, Harshal Patankar*, Sencun Zhu*, Mudhakar Srivatsa† and Jeff Opper‡
*Department of Computer Science and Engineering, Penn State University, University Park, PA, USA
† IBM TJ Watson, NYC, USA ‡ BBN, Boston, USA
Email: {xchen, hdp116, sxz16}@psu.edu, msrivats@us.ibm.com, jopper@bbn.com

***Abstract*—One of the key challenges in operational trust management is to continually monitor the behavior of a node and update its trust score accordingly – evidently, both *speed* and *accuracy* is of great importance here. To achieve these goals, several papers have explored the concept of mutual revocation (sometimes termed *suicide*) wherein the trust value of both the accuser and the accused node are temporarily set to zero without involving a quorum. In this paper we explore a partial mutual revocation approach wherein we design a class of trust update functions to temporarily punish both the accuser and accused node (without involving a quorum) – however, the trust update function does not essentially set their trust values to zero; instead it partially lowers the trust values of both the accuser and the accused. In addition, we allow a trusted authority or a quorum may (periodically) review such partial mutual revocations and update the trust values of the accuser and the accused nodes accordingly (e.g., reward the accuser and punish the accused if the accusation was deemed true). We present a detailed design of the trust update functions for partial mutual revocation. Through both analysis and simulations, we evaluate the effectiveness of partial revocation under different attack strategies and report its performance in terms of revocation immediacy, revocation accuracy and abuse resistance.**

***Index Terms*—Partial Trust Revocation, Mutual Revocation, Trust Management, Ah Hoc Networks**

I. Introduction

Trust management is a central issue in ad hoc networks that are often deployed in adversarial settings and disaster management scenarios [1], [2], [3]. In such settings, compromised nodes can divert and monitor traffic, influence quorum-based decisions or spread harmful information. Also, since mobile ad hoc network (MANET) routing involves a cooperative process where route information is relayed between nodes, any secure routing mechanism must evaluate the trustworthiness of other nodes. Therefore, to limit the damage caused by compromised nodes and to provide a secure routing mechanism, agile trust management schemes that allow rapid impeachment of malicious nodes are vital for the security of the network.

One of the key challenges in operational trust management is to continually monitor the behavior of a node and update its trust score accordingly – evidently, both *speed* and *accuracy* is of great importance here. Past work has explored quorum based approaches wherein a group of nodes deliberate on trust revocation decisions and update the trust values of the accuser and the accused nodes accordingly – unfortunately, in the context of ad hoc networks such an approach lacks speed and agility. To address this problem, several papers have explored the concept of mutual revocation (also termed *suicide* [4], [5], [6], [7], [8]) wherein the trust value of both the accuser and the accused node are temporarily set to zero without involving a quorum; a trusted authority or a quorum (periodically) reviews such mutual revocations (suicides) and updates the trust values of the accuser and the accused nodes accordingly – immediate revocation offers higher agility and speed which generally favors ad hoc network operation, albeit at the cost of lowering the accuracy (i.e., both good and bad nodes may get revoked).

In this paper we present *Zigzag* – a partial mutual revocation approach wherein we design a trust update function to temporarily punish both the accuser and accused node (without involving a quorum) – however, the trust update function does not essentially set their trust values to zero; instead it partially lowers the trust values of both the accuser and the accused. Similar to the complete mutual revocation approach a trusted authority or a quorum may (periodically) review such partial mutual revocations and update the trust values of the accuser and the accused nodes accordingly. In doing so we intuitively interpret trust score as a currency: a currency that is expended by both the accuser and the accused when they participate in (partial) mutual revocation; a currency that is earned when a quorum or a trusted authority agrees with the accuser's decision to (partially) revoke the accused; a currency that allows a node to exert higher influence in network operations (e.g., leader election, routing and forwarding decisions, etc.).

We remark that one of the key benefits of mutual revocation is its inherent ability to work in sparse dynamic networks where global (network-wide) evidences are hard to collect. Essentially, partial revocation offers the node a certain degree of freedom to tradeoff the extent of sacrifice with the global good of the network. In particular, we construct trust update mechanisms for partial mutual revocation that not only enables this tradeoff, but also is robust to strategic (false) accusations made by bad nodes and erroneous accusations made by good nodes (e.g., due to benign errors in network monitoring). We present both analytical solutions that model evolving trust using recurrence equations over time and experimental evaluation to quantify the efficacy of the proposed approach using three important metrics: revocation immediacy, accuracy and abuse resistance. Revocation immediacy is the time taken for a node to be revoked from network once it has been identified as malicious. Accuracy is mainly concerned with minimizing the effects caused due to misidentification of nodes. And finally,

abuse resistance deals with avoiding malicious nodes from taking advantage of the proposed trust revocation scheme for their own benefit. Our scheme encourages honest nodes to accuse malicious nodes by incentivizing them but at the same time discourages malicious nodes to do the same by penalizing them for making false accusations.

The rest of the paper is organized as follows. Section II introduces the related work on revocation and trust management. Section III gives the security and network models. Our design of Zigzag and trust update policies are presented in Section IV and its security analysis is in Section V. Section VI provides simulation results and finally Section VII concludes the paper.

II. Related work

The process of arriving at a revocation decision is the primary focus of the majority of revocation schemes presented to-date in the ad hoc networking literature [9], [10], [11], [5], [6], [8], [12]. Assuming that a node has amassed sufficient evidence, various approaches have been introduced that require differing amounts of participation from other nodes in the network. That is, revocation decision making may be the result of a *collaborative* or *unilateral* decision process.

In collaborative schemes, nodes accuse other nodes of misbehaving by casting negative votes against them. If a predetermined threshold of negative votes are cast, then the offending node is considered revoked. The concept of unilateral decision making as a method of revocation was first introduced by Rivest in dealing with key compromise [13] in Public Key Infrastructures (PKIs). A user, upon detecting that his key has been exposed, declares his key invalid by issuing a signed message using the compromised key (indicating that this key is no longer to be trusted). This notion of suicide has recently been extended for use in ad hoc networks [4], [5], [6], [8]. A node commits suicide by broadcasting a signed instruction to revoke both its own key and the key of the misbehaving node. Suicide as a method of revocation in ad hoc networks has a number of attractive features when compared with collaborative decision making. With suicide, nodes can act immediately to a perceived threat. Additionally, suicide as a method of revocation, is resistant to abuse due to the high cost associated with revoking another node.

It was first pointed out by Raya *et al.* [6] that in order for the suicide scheme to work properly, the node should value the network utility more that his own utility. Raya *et. al.* [6] and Reidt *et. al.* [7] have developed methods to incentivize good nodes to participate in mutual revocation schemes. However, as duly acknowledged by both Raya *et. al.* [6] and Reidt *et. al.* [7], complete mutual revocation takes a heavy toll on the node. For example, consider a small network that contains only 10 nodes out of which two good nodes get involved in a mutual revocation. Raya *et al.* loose both the good nodes; Reidt *et al.* revive one of these nodes, but the other node is permanently revoked. Therefore, an accidental revocation profits the adversary. Every honest node revoked helps the adversary strengthen its influence on the network; also, in a non-cooperative environment it is more likely that the malicious nodes will collude to bring down an honest node than honest nodes cooperating to bring down a bad node.

III. Network and Security Models

Trust Model: We assume every node in the tactical ad hoc network has a public/private key pair, and the public key is known by every other node (or through public key certificate signed by a well-known CA). We further assume that each node in the network may have one or more identifiers along with its corresponding private keys. The trust level associated with an identifier is a fuzzy trust value which ranges between 0 (untrusted) and 1 (trusted).

Every node has an embedded Intrusion Detection System (IDS), which monitors its neighbors for any kind of malicious activity. For the sake of simplicity in this paper we focus on simple packet dropping attacks. Irrespective of the nature of such malicious activity, we assume that the IDS may be imperfect (typically represented by false positive and false negative rates). Hence, given an input from an imperfect IDS, a node may decide to launch an accusation against the purported bad node by broadcasting a digitally signed accusation message into the entire network.

In our model, a trusted authority (TA), as a network manager or administrator, can join the network in need. For example, in a tactical network, a TA could be the commander. Ideally most of the accusations that might take place in a network would be the result of malicious activity (e.g., actual packet dropping witnessed by nodes). But the rest of the accusations could be a result of the intentional false accusations made by the malicious nodes and unintentional false accusations made by the good nodes (due to IDS imperfection). This can indeed be true as malicious nodes with the goal of disrupting the operation of the entire system may attempt to accuse as many honest nodes as possible. Therefore, to control any random or unjustified accusations, the TA reviews past accusations and helps identify whether an accusation was justified or not. The TA does so by making probabilistically correct decisions by, for example, posthumously interrogating witnesses or collecting evidences from other nodes in the network. Also, to incentivize nodes to make correct accusations, the TA rewards a node for a justified accusation by providing it additional trust and thus rewarding the node for its actions. After every judgement round, TA will permanently revoke any identifier with trust below a predefined threshold.

Adversary Model: In our model, we assume that the main goal of an adversary is to bring down the throughput of the entire network by dropping as many packets as possible. The simplest way to achieve this goal would be to drop all the packets that reach the malicious nodes. But, by doing so they also risk of being detected. So in order to maximize their overall influence on the network during their own lifetime, the malicious nodes can carefully choose a packet dropping rate. This would not only enable them to drop packets in an effective manner but also help them remain undetected by honest nodes for an otherwise longer time. Also based on different strategies, the adversary may even choose to abuse

the scheme by making false accusations on honest nodes with the goal of reducing average trust of honest nodes. If the traffic in the network is trust driven, then this could lower the throughput.

IV. Zigzag: Partial Mutual Revocation

A. Overview

In contrast to complete mutual revocation, during a partial mutual trust revocation, both accuser and the accused lose partial trust. For example, if node A accuses node B on its forwarding behavior, then both of them may partially lose data forwarding capability. The benefits of this are at least two-folds. First, intrusion detection systems (IDS), especially those based on forwarding behavior monitoring, are prone to errors because of network and systems complexity. The network loses two benign nodes completely when a false accusation occurs, while by partial revocation the impact of such errors is limited. Second, even if a node is not completely trusted in data forwarding, it may still be safe to involve it in forwarding less critical messages. With appropriate replication through either multi-path routing or forward error correction, it would be possible to leverage the remaining resources of a suspicious node for enhancing network throughput.

B. Detailed Design of Zigzag

At a high level our protocol for partial mutual revocation can be summarized as follows: (i) Initially when the nodes are deployed in the environment they are all assumed to be benign. (ii) Whenever a node behaves maliciously, some of its neighboring nodes will accuse it for being malicious. This will lead to a drop in trust levels of both the nodes (accuser & accused). (iii) After a while the TA would come online and analyze all the accusations occurred during its absence. It would then pass its own judgment based on the information it gathers. Based on that judgment the accuser and accused node's trust levels would be modified again. If the judgment is taken in favor of the accuser, then it will be given a small bonus in form of trust and the trust level of the accused node will be left as it is. However, if the judgment is taken in favor of the accused node then the trust level of the accused node will be brought back to the original value and the trust level of the accuser node will not be recovered as a punishment of false accusation. Steps (ii) and (iii) are repeated as long as the network remains operational.

1) Trust Reduction: In our scheme once the accusation takes place, trust levels of both the nodes (viz. accuser and the accused) are reduced as follows:

$$T'_{Accuser} = T_{Accuser}(1 - \beta T_{Accused}) \quad (1)$$

$$T'_{Accused} = T_{Accused}(1 - \beta T_{Accuser}) \quad (2)$$

$\beta \in [0, 1]$ is a parameter to control the severity of accusation. Normally, an accuser could choose β based on the observed attack intensity—the more malicious the observed attack is, the larger β should be (e.g., using a linear or an exponential function). An accuser may also choose β based on other technical or nontechnical considerations. Ultimately it gives the accuser the flexibility to decide the level of sacrifice it is willing to make. We will show how β impacts on the system in our evaluation section.

A key intuition here is that trust should be reduced in the same amount for both nodes (i.e., $\delta T_{Accuser} = \delta T_{Accused} = \beta T_{Accused} T_{Accuser}$) to prevent malicious nodes from taking advantage of the system. As soon as a node finds out that a neighbor is acting in a malicious way, it makes an accusation. For a malicious node, it can accuse another node at will. The accusation involves partially reducing both the accuser and accused node's trust and broadcasting a signed message to the entire network indicating the identifiers of both the accuser and the accused node. After the accusation takes place, each node carries on with it tasks – may it be forwarding packets or even making further accusations.

2) Judgment Criteria: Many such accusations can take place until the TA comes online. When the TA does come online, it collects all the accusations that took place since its last visit[1]. In order to pass a judgment on a specific accusation, the TA takes probabilistic decision based on collected opinions from other nodes on the accused node.

		TA Judgment	
		Good	Bad
Reality	Good	p_t	p_f
	Bad	q_f	q_t

TABLE I
TA's Judgment Probabilities

We abstractly quantify the efficacy of TA's decision using its false positive and false negative probabilities as described in Table I. p_t denotes probability that TA classifies an honest node as an honest node. q_t denotes probability that TA classifies a malicious node as a malicious node. Also p_f and q_f are the false positive and false negative probabilities respectively. They are the probabilities with which TA misclassifies an honest node as a malicious one and malicious one as honest. Here $p_t + p_f = q_t + q_f = 1$.

If an accusation was deemed correct, then the trust of accuser is restored and it is given a small incentive in the form of additional trust. However, if the accusation was deemed incorrect, then the trust level of accused is brought back to the original value. The trust level of the accuser, however, is not brought back to the original value. This is done in order to penalize for making false accusations.

One can use several past approaches for trust assessment to provide us the functionality of a TA. In this paper we assume that the TA collects evidence from several nodes in the network and builds a classifier based on the *k*-means clustering algorithm [14]. The classifier outputs a 0 or 1 which indicates that the accused node is indeed guilty or not based on available evidences. The details of our approach are excluded from this paper. We note that the correctness of our partial mutual revocation protocol depends only on the false positive

[1]How to collect information in an MANET is a well-studied problem, so we do not study it here.

Event	Honest node profit	Malicious node profit
Honest node makes no accusation	0	0
Honest node accuses another honest node	$bp_f - \delta T p_t$	positive
Honest node accuses malicious node	1)$bq_t - \delta T q_f > 0$	2)$(\frac{M}{H-\delta T} - \frac{M}{H})q_f - (\frac{M}{H} - \frac{M-\delta T}{H+b})q_t < 0$
Malicious node makes no accusation	0	0
Malicious node accuses another malicious node	0	3) negative
Malicious node accuses honest node	$-\delta T p_f$	2)$(\frac{M+b}{H-\delta T} - \frac{M}{H})p_f - (\frac{M}{H} - \frac{M-\delta T}{H})p_t < 0$

TABLE II

PROFITS MADE BY AN HONEST AND MALICIOUS NODE FOR DIFFERENT KINDS OF ACCUSATIONS EVENTS. THE PROFITS STATED REPRESENT AN HONEST NODE'S LOCAL VIEW OF THE NETWORK AND A GLOBAL VIEW FOR MALICIOUS NODE. IN THE COLUMN OF "HONEST NODE PROFIT", THE INEQUALITIES SHOULD HOLD TRUE FOR THE SCHEME TO BE BENEFICIAL FOR HONEST NODES, WHEREAS IN THE COLUMN OF "MALICIOUS NODE PROFIT", THE INEQUALITIES SHOULD HOLD TRUE FOR OUR SCHEME TO BE DISADVANTAGEOUS FOR THE MALICIOUS NODES.

and false negative probabilities of the judgment system; indeed one could in general use any judgment mechanism as long as one can quantify all the parameters in Table I.

3) Trust Update: After the TA passes its judgment on an accusation, the trust levels of both the nodes (accuser and accused) are updated as follows. If the TA rules in favor of the accused, the trust level of the accused node needs to be brought back by adding the reduction value in the accusation. However, if the TA rules in favor of the accuser, then a bonus needs to be provided to the accuser node besides restoring the reduction value. The bonus encourages honest nodes and gives them a reason to make correct accusations and expose the nodes behaving maliciously. Now, bonus can be calculated considering many aspects including: accused node's trust level prior to the accusation $T_{Accused}$, reduction in trust level of the accuser due to the accusation $\delta T_{Accuser}$, trust level of the accuser prior to accusation $T_{Accuser}$, node's previous streak (either winning or losing), etc. Now among these various aspects, if bonus is based on $\delta T_{Accused}$, then the amount of bonus received would be based on the amount of trust lost by the accused node.

Intuition: If an honest node correctly accuses a malicious node which had a high trust level for some reason, then the honest node needs to get due credit for it. The bonus function to achieve this is:

$$b = \gamma \cdot \delta T_{Accused} \tag{3}$$

where $\gamma \in (0, 1)$ is a system set parameter. Therefore, the $T^{New}_{Accuser}$ can be written as:

$$T^{New}_{Accuser} = T_{Accuser} + \gamma \cdot \delta T_{Accused} \tag{4}$$

The reason for limiting the parameter to 1 is to ensure that accuser never receives bonus more than the amount of trust lost by the accused during the accusation. If on the other hand, $\gamma > 1$ were permitted then two malicious nodes could take undue advantage of this and build their own trust by simple accusing each other over and over.

4) Profit Evaluation: Table II lists the expected profit for honest and malicious nodes for each type of accusation events. These expected profits are based on the TA judgment probabilities listed in Table I. For an honest node the main motive is to be as useful as possible. This can be achieved by increasing its trust or by making correct accusations (thereby pinpointing malicious nodes in the network). On the other hand malicious nodes want to disrupt network operations by reducing the average trust value of honest nodes or increasing their own average trust value. Therefore, we count malicious nodes's profit as the ratio of total trust values of malicious nodes to those of honest nodes. Then, we examine all possible accusation events to ensure the following requirements:

1) An honest node can obtain profit by following rules to accuse a malicious node;
2) Malicious nodes cannot gain profit by accusing honest nodes or being accused by honest nodes;
3) Malicious nodes cannot gain profit by accusing with each other.

Let M be the total trust values of malicious nodes in the network and let H be the total trust values of honest nodes. δT is the amount of trust reduction of the accuser node or accused node in the accusation. An honest node can gain a bonus b if the accusation is justified whereas it will lose δT in trust level if not justified. Then for condition 1), we have $bq_t - \delta T q_f > 0$. For condition 2), in the case that an honest node accuses a malicious node, the ratio of total malicious nodes' trust values to total honest nodes' trust values is increased by $\frac{M}{H-\delta T} - \frac{M}{H}$ if TA favors the malicious node whereas the ratio is reduced by $\frac{M}{H} - \frac{M-\delta T}{H+b}$ if TA favors the honest node; in the other case that a malicious accuses an honest node, the ratio of total malicious nodes' trust values to total honest nodes' trust values is increased by $\frac{M+b}{H-\delta T} - \frac{M}{H}$ if TA favors the malicious node whereas the ratio is reduced by $\frac{M}{H} - \frac{M-\delta T}{H}$ if TA favors the honest node. Hence, we can conclude two inequalities for condition 2) as shown in Table II. For condition 3), recall from the trust update section that the choice of parameter γ ($\gamma < 1$) ensures that malicious nodes do not gain by accusing other malicious nodes. For condition 1) and 2), we derive the lower bounds for the TA's judgment probability p_t and q_t:

$$p_t > \frac{bH + \delta T M}{\delta T(H - \delta T)} p_f,\ q_t > \max(\frac{(H+b)\delta T M}{(H-\delta T)(bM + \delta T H)}, \frac{\delta T}{b}) q_f$$

The loosely derived lower bounds for both are as follows:

$$p_t > \frac{\gamma H + M}{H - 1} p_f,\ q_t > \max(\frac{M}{H-1}, \frac{1}{\gamma}) q_f$$

5) Trust-Aware Partial Data Access: Different with the situation in complete mutual revocation, a node being partially revoked in partial mutual revocation still retains some extent of network capability based on its trust level, and hence can contribute to the network. However, to differentiate nodes with

various trust levels, it is necessary to associate a node's trust level with its network capability. For example, trust is widely used to assist route selection [15], [16] and therefore nodes with low trust levels are more likely to be excluded from packet forwarding. Here, we propose trust-aware partial data access, which aims to tie a node's trust level to its capability in data access. A group key is used to protect the confidentiality of communication within the network. We require that a node with low trust can only have partial group key and thus understand partial data packets encrypted by the group key.

Problem Model: We assume every data packet has a secrecy level which is ranged from 0 (least secrecy) to $M-1$ (most secrecy). Every node j has a hierarchical rank R_j from 0 (least capability) to $M-1$ (most capability) in the MANET corresponding to its trust level T_j. Hierarchical ranks of nodes are evaluated by TA periodically by using a linear function: $R_j = \lfloor T_j \cdot M \rfloor$. The design goal is to ensure that a node can only decrypt any data packet whose secrecy level is at most the node's rank evaluated by TA recently.

Scheme: Whenever TA finishes updating all nodes' trust levels after accusation judgment, TA evaluates all nodes' ranks and distributes new group keys based on the new ranks. In each group rekeying, TA generates a key chain of size M. Let the keys in the key chain generated for the rekeying at time t be $K^{M-1}(t), K^{M-2}(t), \cdots, K^1(t), K^0(t)$, where $K^0(t) = H(K^1(t)) = H^2(K^2(t)) = \cdots = H^{M-1}(K^{M-1}(t))$ and H is a one-way hash function such as SHA-1. Due to the one-wayness of the hash function, a node that knows $K^i(t)$ can iteratively compute the keys $K^{i-1}(t), \cdots, K^0(t)$ and understand any packets encrypted by these keys, but cannot compute any of the keys $K^{i+1}(t), \cdots, K^{M-1}(t)$ and understand the associated packets. Then, TA sends to each node j key $K^{R_j}(t)$ for the next communication session. Each node derives its own sub keychain based on the one way function. During the session, a sender will encrypt a data packet whose secrecy level is j with group key $K^j(t)$. And, only the nodes whose ranks are at least j can decrypt and understand the packet. Besides, the scheme ensures that a node cannot properly encrypt any data packet whose secrecy level is higher than its own rank.

V. Security analysis

A. Basic Analytical Model

To make our analysis tractable, we assume that TA comes online every interval time T_1 to handle all the accusation events that happened during its absence. T_1 is a system parameter that the authority can adjust by considering both the efficiency and overhead. Also, each honest node takes a time period of T_2 to gather enough evidence, via its IDS, to launch an accusation against a malicious node. The choice of this parameter mainly depends on the performance of IDS by considering a tradeoff between accusation immediacy and accuracy. We define the timeline of TA's online round i as $[i \cdot T_1, (i+1) \cdot T_1)$. Hence, the maximum number of accusations that can be launched by a node in each TA's online round is given by: $\omega = T_1/T_2$. To avoid malicious nodes from repeatedly accusing honest nodes for abusing the whole system and limit the overhead of accusation traffic, any node that make more than ω accusations in one round will be directly revoked by the TA. For analysis purpose, we assume that the IDS of an honest node will accuse a malicious node with a probability of α_i when the malicious node's attack intensity is α_i: in the context of packet forwarding α_i is the probability with which a malicious node drops packets in round i. Hence, the average number of accusations made by each honest node in i is:

$$\lambda_i = \sum_{k=0}^{\omega} k \cdot \binom{\omega}{k} \cdot \alpha_i^k \cdot (1-\alpha_i)^{\omega-k} = \omega\alpha_i \tag{5}$$

Denote the *average* trust levels of honest nodes and malicious nodes as two-dimension arrays T_h and T_m, respectively. Specifically, $T_h(i, j)$ and $T_m(i, j)$ are the respective average trust levels of honest nodes and malicious nodes after the jth accusation round during round i. i starts from 0 and continues till the network is operational. j starts from 0 and ends at λ_i.

Given that two malicious nodes have no incentive to accuse each other and two honest nodes rarely accuse each other, the total number of accusations that involve honest nodes must equal those that involve malicious nodes. Hence, in each round for every accusation that involves an honest node, there are $\frac{n}{m}$ accusations that involves a malicious node (where n is the number of honest nodes and m is the number of malicious nodes and typically, $n > m$). Therefore, based on Formula (1) and (2), trust value of malicious nodes evolves as follows [2]: $T_m(i, j) = T_m(i, j-1) \cdot (1 - \beta T_h(i, j-1))^{\frac{n}{m}}$

It is a little more complex to approximate $T_h(i, j)$ because the average trust value of malicious nodes varies considerably after every m accusations. Hence, we divide an accusation round into $\frac{n}{m}$ phases, and at each phase m of n honest nodes accuse malicious nodes for a total of m times. Hence, the average trust of m honest nodes at each phase k (from 1 to $\frac{n}{m}$) is formulated as below:

$$T_{h\_k}(i, j) = T_h(i, j-1)(1-\beta T_m(i, j-1) \cdot (1-\beta T_h(i, j-1))^{k-1})$$

To approximate the average trust of n honest nodes, we add up the average trust of m honest nodes at each phase with a weight of $\frac{m}{n}$, i.e., $T_h(i, j) = \sum_{k=1}^{\frac{n}{m}} \frac{m}{n} \cdot T_{h\_k}(i, j)$. After deduction, we have:

$$T_h(i, j) = T_h(i, j-1) \cdot \left(1 - \frac{mT_m(i, j-1)(1-(1-\beta T_h(i, j-1))^{\frac{n}{m}})}{n\beta T_h(i, j-1)}\right)$$

To recursively solve the above two formulas, we first need to determine $T_h(i, 0)$ and $T_m(i, 0)$. Clearly, $T_h(0, 0)$ and $T_m(0, 0)$ are the initial trust levels assigned to good nodes and bad nodes, respectively. For analysis purpose, let us assume they are both 0.8. More general, $T_h(i, 0)$ is determined by judgment decisions based on λ_{i-1} accusations at the $(i-1)$th

[2]We acknowledge that the function approximates the expectation of trust levels by directly involving the expected value for the simplicity.

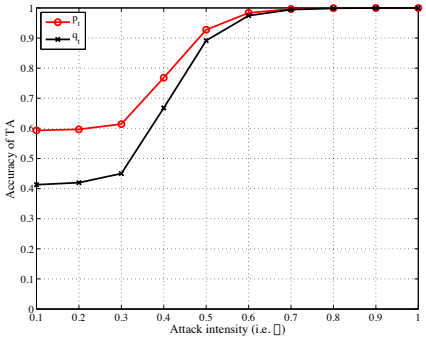

Fig. 1. TA accuracy vs. attack intensities

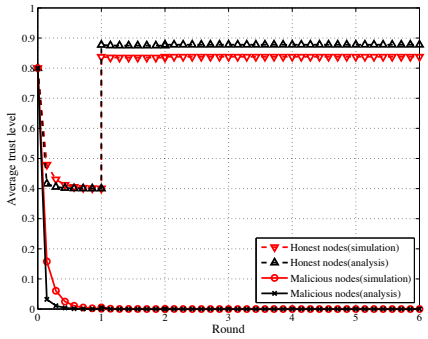

Fig. 2. Avg. Trust for $\alpha = 0.7$

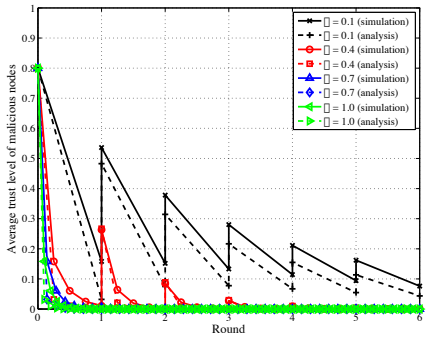

Fig. 3. Avg. Trust of malicious nodes vs. α

round. We can formulate $T_h(i,\,0)$ by considering all of λ_{i-1} accusations one by one in a probabilistic way. It is much simpler to get the average trust level of malicious nodes $T_m(i,\,0)$ as TA passes its judgment only once for each malicious node and all accusation events containing the same accused node follow this judgment decision. If accusations against malicious nodes are justified, the malicious nodes would keep the trust level as in the end of TA's previous online round; otherwise, the average trust level of malicious nodes would be restored to that in the beginning of TA's previous online round. To summarize, we have the following formulas:

$$\begin{aligned} T_h(i,\,0) &= T_h(i-1,\,0) \\ &\quad + \sum_{j=1}^{\lambda_{i-1}} (q_t \cdot \gamma \cdot (T_m(i-1,\,j-1) - T_m(i-1,\,j)) \\ &\quad -(1-q_t) \cdot (T_h(i-1,\,j-1) - T_h(i-1,\,j))) \quad (6) \\ T_m(i,\,0) &= q_t \cdot T_m(i-1,\,\lambda_{i-1}) + (1-q_t) \cdot T_m(i-1,\,0) \quad (7) \end{aligned}$$

Now that we have captured the evolution of trust for malicious and honest nodes through recurrence equations over time, we quantify the expected reward for a malicious node as follows. For the sake of simplicity we assume that in each round i, the expected reward for a malicious node is proportional to the product of their average trust level and their attack intensity α. The reward for a malicious node at each round i is evaluated as $\frac{\sum_{j=0}^{\lambda_i-1} T_m(i,\,j)}{\lambda_i} \cdot \theta^i \cdot \alpha_i$, where $0 < \theta < 1$ is a discount factor[3] that weighs immediate reward for a malicious node more than its future rewards. Hence, the total profit of a malicious node over the network lifetime is:

$$R = \sum_{i=0}^{\infty} \frac{\sum_{0}^{\lambda_i-1} T_m(i,\,j)}{\lambda_i} \cdot \theta^i \cdot \alpha_i \quad (8)$$

The malicious nodes could strategically choose a vector $\alpha = (\alpha_1,\,\alpha_2,\,\ldots)$ to maximize their expected long-term reward.

B. Evaluation

This subsection contains two revocation cases: honest nodes accusing malicious nodes and malicious nodes accusing honest nodes. For the first case, our evaluation is based on the basic analytical formulas derived in Section V-A. For the second, we wrote C simulators based on our trust update formulas (Formulas 1-4) without considering node mobility.

1) Honest Nodes Accusing Malicious Nodes: In this part of evaluation, we assume only honest nodes accuse malicious nodes. Figure 1 shows how the TA's accuracy (q_t and p_t) changes with the attack intensity α based on the aforementioned k-means clustering algorithm. As the attack intensity α increases, the probabilities that TA correctly identifies a malicious and an honest node dramatically increase, and they both nearly reach 1 when α is over 0.7.

The analytical results shown next were carried out with a fixed malicious-to-honest node ratio $(\frac{m}{n}) = 0.5$ and the varying TA's accuracy which changes with the attack intensity based on the curves shown in Figure 1. The number of accusation rounds (i.e. ω) in a TA's online interval is set to 10. β is set to 1. Except for Figure 4, γ is 0.1 by default. To validate our analytical model, we also did a number of simulations with the same parameter settings as the above and the validation results are shown in Figure 2 and 3.

Figure 2 shows how the average trust of honest and malicious nodes changes over different rounds. Here the attack intensity for a malicious node is set to 0.7. First, we can see the simulation results match with the analytical results very well. Second, as the rounds progress, the average trust associated to malicious nodes decreases quickly whereas the average trust associated to honest nodes does not. This is due to the fact that during each round many accusations against malicious nodes take place and in the end of each round all those accusations are judged by the TA. When the TA passes its judgment, malicious nodes hardly recover to their trust of the previous round. On the other hand, the honest nodes are awarded a bonus for making correct accusations. Since the deducted trust of accused nodes also depends on the trust level of accusers, with higher trust levels, honest nodes can bring down the average trust level of malicious nodes more in later rounds.

Figure 3 shows how the average trust of malicious nodes changes over different rounds with varying attack intensities. Each curve in the figure, either by simulation or by analysis, corresponds to a fixed attack intensity. First, we can

[3]Discount factor is commonly used approximation for infinite horizon problems [17], [18].

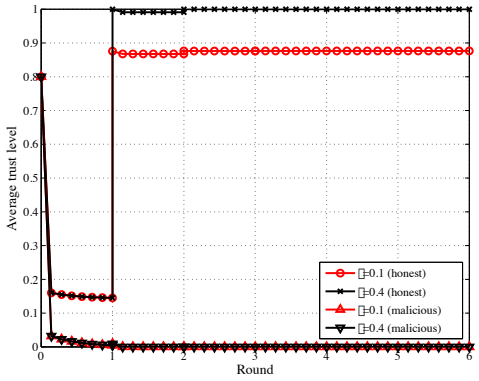

Fig. 4. Avg. Trust vs. γ

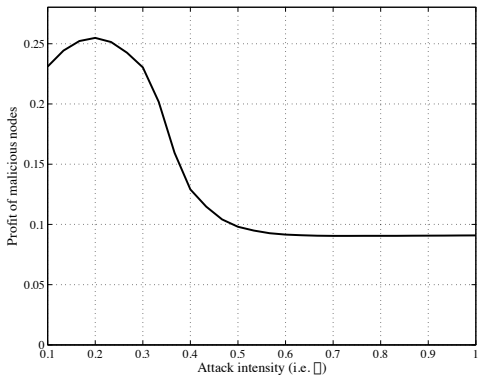

Fig. 5. Profit of mal. nodes vs. α

see that the analysis and simulation results are very close, thus validating our analytical models. Second, as the attack intensity α increases, the average trust of malicious nodes decreases more quickly over each round. This happens because honest nodes have a higher probability to accuse bad nodes at each accusation round as α increases. Higher accusation probability reflects higher accusation frequency in the figure. Another reason is that as α increases, the TA's accuracy q_t also increases (as shown in Figure 1). Thus, the trust level of malicious nodes is less likely to be restored by TA. We can also see that the change of trust level over rounds is a zigzag shape, so we name our revocation protocol *Zigzag*.

Figure 4 shows how the average trusts of malicious nodes and honest nodes change under different values of γ. It is very clear that average trust of honest nodes increases as the bonus parameter γ increases. Consequently, the average trust of malicious nodes drops much faster.

Figure 5 shows the total profit earned by a malicious node over its entire lifetime with various attack intensities. Here, the attack intensity α is fixed to a certain value during the entire lifetime, which means $\alpha_1, \alpha_2, \ldots$ are equal. It reveals that in our scheme malicious nodes can achieve higher long-term profit with a relatively low attack intensity. Especially, the optimal α is around 0.2. This is because q_t and p_t are affected by the attack intensity. Here the functionality of bad nodes is assumed to be proportional to their trust levels.

Here an interesting question is: assuming the global knowledge of the system, would the malicious nodes gain a higher long-term profit by adopting a different attack intensity α_i at a different round? That is, the malicious nodes may drop packets at different rates at different rounds so that they may drop the maximal number of packets. We use an efficient heuristic search algorithm to figure out this optimal attack intensity vector under the searching granularity of 0.1. Interestingly, our result suggests the optimal attack intensity vector α_{opt} in this setting is the same as the previous optimal α, that is, all $\alpha_i = 0.2$.

2) Malicious Nodes Accusing Honest Nodes: As noted previously, the main motive of malicious nodes is to disrupt the network. Accusing honest nodes repeatedly and bringing their average trust level down could be one of the effective ways as this can lower the network throughput. We present two different types of attack strategies below.

Figure 6 represents the *One-to-One* attack scenario where each malicious node accuses a different honest node at each accusation round. No two malicious nodes accuse the same honest node. Their main motive is to bring down as many honest nodes as possible in a single round. As the TA updates trust levels at the end of each round, a malicious node keeps accusing the same honest node until it is evicted from the network. Here, we compare the average trusts of malicious nodes and honest nodes under three different settings of $\frac{m}{n}$. It can be seen from the figure that as $\frac{m}{n}$ increases, i.e., with more malicious nodes, one-to-one accusation is more effective by the end of TA's round. However, as the TA updates the trust levels, the trust of malicious nodes decreases whereas the trust of honest nodes increases. This happens because the TA judgment probability (p_t) is usually higher than 0.5 to support honest nodes. As the rounds progress, the average trust of honest nodes is not affected much but the average trust of malicious nodes reaches zero.

Figure 7 represents a *Many-to-One* attack scenario where malicious nodes collude and keep on accusing honest nodes sequentially. Specifically, at each accusation round, each malicious node will pick to accuse the active honest node whose trust level is the lowest. As a result, an honest node will receive multiple accusations until its trust level is below a threshold and considered inactive. The main objective is to break down as many honest nodes as possible. A predefined threshold (0.05 in our simulation) is set to determine whether an honest node is active according to its trust level. It can be seen from the figure that many honest nodes are brought down in the first couple of rounds. When the round comes to an end the TA updates all the trust levels, the average trust of malicious nodes falls below that of honest nodes. This trend continues in the first few rounds until their difference becomes rather large. Once that happens malicious nodes are not able to make any sizable impact on honest nodes' trust levels, let alone bring them down. As they continue this type of sequential accusation, their own trust level decreases and

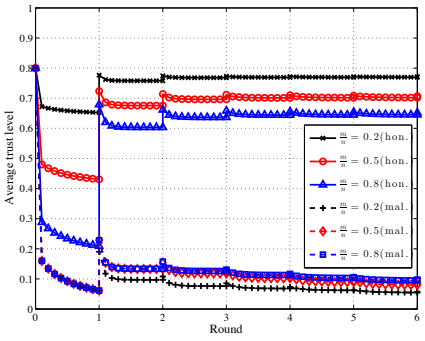

Fig. 6. One-to-One accusation ($p_t = 0.8$)

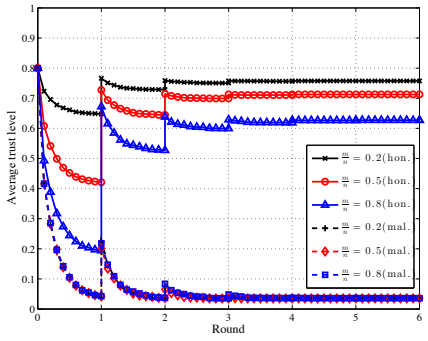

Fig. 7. Many-to-One accusation ($p_t = 0.8$)

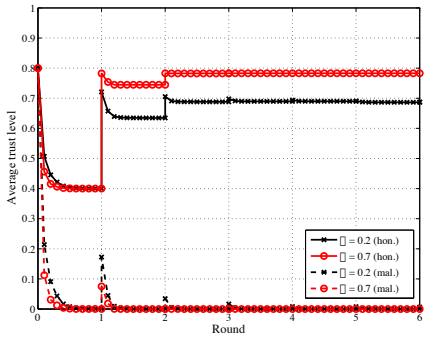

Fig. 8. Mutual accusations

	Parameters	Description	Default
Scenario	N	# of nodes in the network	100
	n	# of honest nodes in the network	80
	m	# of malicious nodes in the network	20
	α	attack intensity of malicious nodes	1
	T_2	interval of TA online round	100 units
Zigzag	β	coefficient of trust reduction	0.1
	γ	coefficient of trust reward	1
Complete revocation	b	coefficient of trust reward	1
IDS	p_{f_IDS}	false positive of IDS	0.05
	q_{f_IDS}	false negative of IDS	0.05
	T_1	interval of accusation round	2 units
TA	p_f	false positive of TA judgment	0.2
	q_f	false negative of TA judgment	0.2

TABLE III
PARAMETER SETTING

after few rounds it approaches zero.

Based on the comparison of 6 and Figure 7, it can be observed that from the adversary's point of view, the Many-to-One attack and the One-to-One attack do not have much difference in their accusation effectiveness. This observation has the following important indication. As these two accusation strategies represent two extreme cases for an adversary, any other accusation strategy, which we do not enumerate here, would be a hybrid version of these two cases, so its attack effectiveness would also be similar.

3) Mutual Accusation: Finally, we consider the case that malicious nodes and honest nodes accuse each other. Here, we mix the above scenarios to simulate a more realistic setting. Under this attack scenario, malicious nodes perform malicious activities dropping packets with the attack intensity α of 0.7, which lead to the potential accusations by honest nodes. Besides, at each accusation round, they will randomly choose an active honest node (i.e., whose trust level is above 0.05 in our setting) to accuse. It can been see from Figure 8 that compared with Figure 7 and Figure 6, the average trust of malicious nodes falls more quickly because their dropping behavior is sensed and hence they are accused by honest nodes.

VI. COMPARATIVE STUDY

We further evaluate Zigzag through simulations and show its *revocation immediacy* and *accuracy* in comparison with the complete revocation scheme [7]. Besides, we study how β impacts security.

Our simulation is based on the simulator GloMoSim 2.03. Table III describes the general parameter settings. We choose DSR as the routing protocol and 802.11 as the MAC layer protocol. The mobility model is random waypoint; node's velocity is between 0 and 10 m/s and its pausing time is 30 time units. In the simulation, periodically each node requests a simple response from a neighbor chosen at random. An honest node will always send back an ACK in response, whereas a malicious node only responds to the request at a probability pre-determined by its attack intensity. At each accusation round, an honest node will choose to accuse a node that is identified to be malicious by its IDS, if any. Because of the imperfection of its IDS, an accused node might or might not be an actual bad node. The TA will come online to judge all accusations and update the trust levels of all accuser and accused nodes every T_2 time. By default, the judgment accuracy is fixed. Both Zigzag and complete revocation are implemented and evaluated under this setting. The results are averaged over 20 independent runs.

Revocation Immediacy is the time taken (or number of accusations needed) for a node to be revoked from network once it is identified as malicious. *Accuracy* is mainly concerned with minimizing the effect caused by faulty IDS (leading to false accusations). Figure 9 provides a comparison on these two metrics between partial and complete revocation. Figure 9 shows that the average trust level of malicious nodes drop quickly. In the complete revocation scheme, it reaches

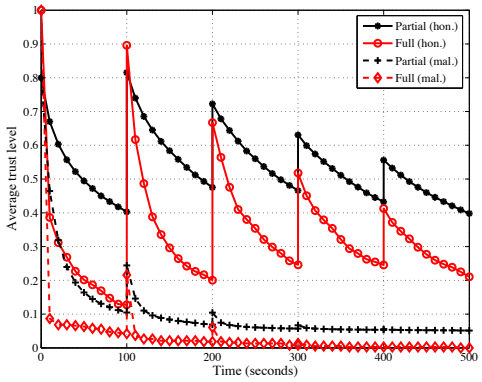

Fig. 9. Partial vs. complete revocation

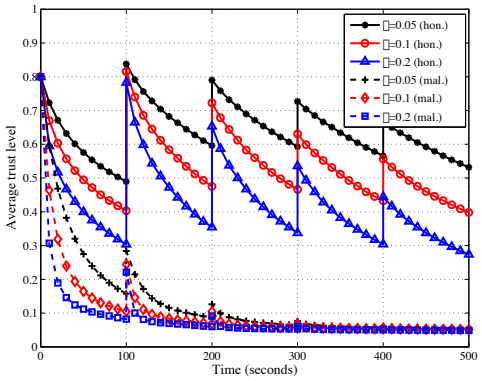

Fig. 10. Ave. trust level vs. β

almost 0 in 3 rounds, and in ZigZag it reaches to 0.05 in 3 rounds. Although this indicates the former has higher revocation immediacy, in our partial revocation scheme, a node is considered revoked after its trust level drops below 0.05. So in this sense, the revocation immediacy of Zigzag is close to that of the complete revocation scheme.

On the other hand, the figure also shows the big advantage of Zigzag over the complete revocation scheme in terms of accuracy. Due to false positives of IDSs in mobile nodes, false accusations have a much greater influence on the complete revocation scheme, as its average trust level of good nodes is far below that of Zigzag. This demonstrates Zigzag works much better in tolerating IDS inaccuracies.

The parameter β denotes the degree of trust reduction in Zigzag. From Figure 10, we can observe the tradeoff between accusation immediacy and accuracy. When β decreases, the revocation speed is lower but accuracy is better. To the opposite, when β increases, the revocation speed is faster but accuracy is worse. Hence, β can be flexibly selected based on the context of network, the severity of observed attacks, and other factors.

VII. Conclusions and Future Work

In this paper, we have introduced Zigzag, a new scheme for trust management in ad hoc networks. First, based on a node's fuzzy trust value, its network privileges are modulated under a model of partial revocation. Second, for better revocation immediacy and abuse resistance, we explored the idea of mutual trust revocation. The partial revocation approach presents its trade-offs between revocation immediacy and accuracy. Third, by providing trust in the form of incentives, it encourages honest nodes to make right accusations but at the same time also discourages malicious nodes by penalizing them for making false accusations. Our future work will study other possible attack strategies as well as more extensive simulations to compare Zigzag with other existing revocation schemes.

VIII. Acknowledgments

This work was supported in part by the NS-CTA grant from the Army Research Laboratory (ARL) and the U.S. NSF CAREER 0643906. The views and conclusions contained here are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either express or implied, of ARL or NSF.

References

[1] K. Aberer and Z. Despotovic, "Managing trust in a peer-2-peer information system," in *Proc. CIKM*, 2001.
[2] B. Ooi, C. Liau, and K. Tan, "Managing trust in peer-to-peer systems using reputation-based techniques," *Advances in Web-Age Information Management*, pp. 2–12, 2003.
[3] Y. Wang and J. Vassileva, "Trust and reputation model in peer-to-peer networks," in *Proc. P2P*, 2003.
[4] J. Clulow and T. Moore, "Suicide for the common good: A new strategy for credential revocation in self-organizing systems," *SIGOPS Oper. Syst. Rev.*, vol. 40, no. 3, pp. 18–21, 2006.
[5] T. Moore, M. Raya, J. Clulow, P. Papadimitratos, R. Anderson, and J.-P. Hubaux, "Fast exclusion of errant devices from vehicular networks," in *Proc. SECON*, 2008.
[6] M. Raya, M. H. Manshaei, M. Félegyhazi, and J.-P. Hubaux, "Revocation games In ephemeral networks," in *Proc. CCS*, 2008.
[7] S. Reidt, M. Srivatsa, and S. Balfe, "The fable of the bees: incentivizing robust revocation decision making in ad hoc networks," in *Proc. CCS*, 2009.
[8] R. A. T. Moore, J. Clulow and S. Nagaraja, "New strategies for revocation in ad-hoc networks," in *Proc. ESAS*, 2007.
[9] S. Zhu, S. Setia, S. Xu, and S. Jajodia, "Gkmpan: An efficient group rekeying scheme for secure multicast in ad-hoc networks," *Journal of Computer Security*, vol. 14, no. 4, pp. 301–325, 2006.
[10] D. Liu, P. Ning, and K. Sun, "Efficient self-healing group key distribution with revocation capability," in *Proc. CCS*, 2003.
[11] W. Liu, "Securing mobile ad hoc networks with certificateless public keys," *IEEE TDSC*, vol. 3, no. 4, pp. 386–399, 2006.
[12] S. Yi and R. Kravets, "MOCA: Mobile certificate authority for wireless ad hoc networks," in *Proc. PKI*, 2003.
[13] R. Rivest, "Can we eliminate certificate revocation lists?" in *Proc. FC*, 1998.
[14] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE TPAMI*, 2002.
[15] S. Marti, T. Giuli, K. Lai, M. Baker *et al.*, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. International Conference on Mobile Computing and Networking*, 2000.
[16] A. Pirzada, A. Datta, and C. McDonald, "Trust-based routing for ad-hoc wireless networks," in *Proc. ICON*, 2004.
[17] J. M. McCune, E. Shi, A. Perrig, and M. K. Reiter, "Detection of denial-of-message attacks on sensor network broadcasts," in *Proc. S&P*, 2005.
[18] D. J. White and C. E. White, *Markov decision processes*, 1st ed. Wiley, John & Sons, Incorporated, 1993.