Estimation of Trust Metrics for MANET Using QoS Parameter and Source Routing Algorithms

D. Umuhoza[1], J.I. Agbinya[2], C.W. Omlin[3]
[1]Meraka Institute, PO Box 395, Pretoria 0001,
South Africa; (dumuhoza@csir.co.za);
[2]Information and Communication Technology Group, University of Technology, Sydney, NSW 2007,
Australia; (agbinya@eng.uts.edu.au);
[3]Department of Computer Science, University of the Western Cape, Private Bag X17, Bellville 7535,
South Africa; (comlin@uwc.ac.za)

Abstract

Estimation of trust in ad-hoc networks is an inevitable basis for hybrid networks to inter-operate. The contributions in this paper provide a framework for estimating the trust between nodes in an ad hoc network based on quality of service parameters. Probabilities of transit time variation, deleted, multiplied and inserted packets, processing delays are used to estimate and update trust. Functions which facilitate this are provided and evaluated. It has been shown that only two end nodes need to be involved and thereby achieve reduced overhead. The framework proposed is applicable and useful to estimate trust in covert unobservable and anonymous communications.

Key words: *Trust, metrics, ad hoc networks, unobservable and anonymous communications*

1. Introduction

Mobile Ad-hoc networks are self organized networks without reliance on any fixed network infrastructure. They are characterized by changing topology caused by mobility of nodes within the network or by nodes leaving and joining the network. The openness setting of ad-hoc networks attract users to act selfishly in order to save power or to take advantage of lack of central administration and perform malicious actions on the network.

Several routing protocols have been developed to suit Ad-hoc networks and [1] and [2] are the most popular.
Depending on the purpose of using a mobile ad-hoc network, more secure measures might be necessary in routing of data packets. Improving security in mobile ad-hoc networks has therefore become a hot topic in research in recent years and many solutions have been proposed [3 - 8].

Mobile ad-hoc networks can have a wide range of properties depending on the number of nodes in a network, distance between nodes, devices used and the movement of the nodes. The work presented in this paper at this stage is limited to specific cases where parameters of the environment are predictable. For example the use of mobile devices in a conference room, in an office or other places where movement and obstacles between devices can be predicted.

In such environment, it is possible to define the essential wireless link parameters necessary for creating the metrics of trust of a communication path which is composed of a set of contiguous wireless links.
In this paper we propose a simple metric of trust that will allow users of the network to monitor behavior of their communication path. Especially, packet oriented attacks are considered while designing the metrics of trust.

The metric of trust we propose is based on protocols that share the property of source routing for the reason that in source routing, data packets follow the same route. Hence packets are most likely to get to the final destination in the same order in which they are sent even in cases where link breakage occurs, because if a link fails, data packets are sent via a different route from the point of failure or from the source of the packets. Furthermore, these metrics of trust provide for privacy of identity and location as are fully implemented and yet users are able to monitor their communication traffic without violating the privacy rules.

The rest of this paper is organized as follows: Section 2 presents analytical summary of existing trust models incorporated in MANET routing protocols. The new metrics of trust are presented in section 3. Different probabilities for detecting anomalies are computed in section4. Section 5 presents results of experiments and section 6 concludes the paper.

2. Related work

Solutions to security in routing in ad-hoc networks have been proposed but yet, none of them is really a complete solution. The results in [3] and [8] enhance

security of routes by incorporating trust in the route discovery process but they have no considerations for covert or unobservable communications between two end nodes. Route discovery process should be resistant to attacks that modify or fabricate routing information with the aim of denial of service attacks. Once routes are discovered in compliance to those protocols, they are considered to be secure for any communication. To implement security at the stage of route discovery is vital. It is as well as necessary to consider the changing behavior of a node that might comply with route discovery process but does the opposite for packet forwarding.

CONFIDANT [9] and TAODV [10] consider the changing behavior of nodes in the network at anytime. They adjust the trust of nodes towards each other at every interaction. However, these two protocols have not considered nor distinguished between malicious node behavior and problems caused by traffic congestion or benign link failures which are the most likely causes of routing failures in mobile ad-hoc networks. Even if a failing node might be regarded as useless as malicious node, but at least when a failing node recovers from error it should not be regarded as harmful to communication of other nodes in the network. We recognize that distinguishing between a malicious node and a failing node because of error is still an open problem.

Group security provides an avenue for new attacks and security risks. In cases where users of mobile devices are roaming in insecure environment, nodes can be captured while already being in use. For example in SAR [8] and in SDAR [11] if at least one user of the same trust level group is compromised, all users of the same group are exposed to security attacks.

As nodes watch their neighbors forwarding packets and report to other nodes in the network, nodes giving false reports in CONFIDANT can force other nodes to be excluded from the network.

In CORE [12], a node can have rapidly increasing positive credits if there is a node reporting false positive information about that node. That positive credit that a node gets will not reduce in the same ratio as it has increased since only positive reports are considered. Nodes may collaborate to give false positive reports for each other and they may be able to remain in the network even if they are behaving maliciously.

3. A new metric of trust

The new metric of trust presented in this paper differs from the previous trust models in the sense that it considers an environment where only two end nodes collect evidences and update their opinion on the trustworthiness of the communication path. This metric does not consider unchangeable trust among members of a certain group in a network since each node can be individually compromised. Our method therefore permits unobservable communications and also reduces the overheard associated with determining trust at the intermediate nodes.

On a given ongoing communication, end users may have details of how normal traffic flows from one intermediate node to another, by attaching appropriate additional information to original traffic or in a manner of setting up separate packets. However, in our case we suppose that there are anonymity techniques that can wipe details of how traffic flow through intermediate nodes so that activities at intermediate nodes cannot be linked with their identity [4], [13]. Sender and receiver are then able to collect patterns of traffic at both ends of the communication only. Key anomalies are detected using traffic patterns collected. Trust is computed based on probabilities for anomalies to happen due to benign link faults or security attacks.

4. Anomaly Detection

The quantity of packets and timing of packets have been identified in this work as measured elements that contribute to the change of traffic behavior. The change of those elements causes anomalies.

Anomaly in network traffic is defined as any behaviour of traffic different from what is expected or not satisfactory to the users of a particular communication path. Therefore we assume that the source and destination have what is an acceptable or "satisfactory" communication through the paths. For example, in source routing, the receiver expects to receive packets in the same order as they were sent. Another example is that, in a network where users insist on anonymous communication, a user avoids non-peer users to know whom he/she is communicating with at a particular time.

To see how the behavior of network traffic changes, sender and receiver share the information collected on network traffic, put them together, and analyze them. We focus on both regular and frequent patterns because they can be meaningful.

We reason that an attacker might introduce regular patterns in traffic with the aim of mapping out a particular communication to link a sender with a receiver. We also argue that when anomalies occur on a path frequently, there might be a high chance for those anomalies to occur due to misbehavior of certain intermediate nodes on the path.

After a sender and a receiver start to exchange data packets, they build tables to keep traffic patterns. A table is built by the sender and another one built by the receiver. The two tables have the same structure. Each table is composed of two fields: Packet identification number and time stamp of action. Each time a packet is sent, the sender records the packet ID and the time.

Each time a packet is received a receiver records the packet ID and the time.

Every t seconds, the receiver sends the sender a table. Upon receipt of the table from the receiver, the sender merges it with its own table into an anomaly detection table. The anomaly detection table contains packet identification, sending timestamp and receiving timestamp for each packet. The sender gets the table refreshed every t seconds. Using this information the sender can calculate the various values that will be mentioned in the following subsections and keep them in respective databases.

We are aware that exchanging of tables containing traffic patterns between sender and receiver will cause traffic increase and therefore consume more bandwidth. However we do not intend to address the issue of bandwidth consumption in this paper. However every security comes with a cost. The investigation on whether that cost is worth it or not remains an open and subjective problem. We now discuss the traffic pattern parameters that are recorded by the source and destination for use in computing trust and updating trust.

4.1 Trust Computation Using the Probability of Transit Time Variation

Two equations are used to calculate trip time variation (ΔT_t). Trip time T_t is calculated using the following equation.

$$T_t = T_r - T_s \quad (1)$$

where T_s is the time a packet is sent and the time it is received is T_r and the estimated reference time (T_R) is given by the equation below.

$$T_R = \frac{(RQr - RQs) + (R\,\mathrm{Pr} - RPs)}{2} \quad (2)$$

where *RQr= Time the route request message is received*
RQs= Time route request message is sent
RPr= Time route reply message is received
RPs= Time route reply message is sent

Having the value calculated in equation (2), we can calculate the variation of trip time of each packet.
The variation of trip time is calculated during route discovery process by the following expression.

$$\Delta T_t = T_R - T_t \quad (3)$$

The trip time of one packet alone is not meaningful but observing trip time variations over a period of time will allow the computation of probability of a packet to be delayed. Comparing trip time variation of many packets helps in noticing and examining regular delays that are most likely to be caused by attacks.

With the aid of equation (1) and (2), we define the probability that a packet is not delayed in the path (network) by an external influence with equation (4) as follows:

$$P_{tt} = \frac{T_t}{T_R + b\sigma} \quad (4)$$

b is a constant which we assume lies in the range $2 \le b \le 5$ and σ is standard deviation in trip time of the packet. The choice of b is empirical and may be evaluated by experimentation.

If the trip time of a packet is bigger than the denominator of equation (4) we assume that an unusual event has occurred delaying the packet from arriving on time and hence the path should be less trusted and changed. Therefore, based on the equation (4), we define the trust update probability to be:

$$\Delta P_{tt} \frac{\Delta T_t}{T_R + b\sigma} \quad (5)$$

Therefore our trust update equation becomes equation (6) bellow:

$$\eta_u = P_{tt} + \Delta P_{tt} = \frac{T_t}{T_R + b\sigma} + \frac{\Delta T_t}{T_R + b\sigma} = \frac{T_t + \Delta T_t}{T_R + b\sigma} \quad (6)$$

Equation (6) is very significant as it provides the point at which we should start to distrust a path. Therefore, for the path to remain trusted, the trust update probability must lie within the range:

$$0 \le \Delta p_{tt} \le \frac{b\sigma}{T_R + b\sigma}$$

We define the trust changes with time (the rate of change of trust) as:

$$\eta(t) = \eta t + \frac{d\eta}{dt} t = (\eta + \Delta\eta)t \quad (7)$$

where

$$\Delta\eta = \Delta p_{tt} = \frac{\Delta T_t}{T_R + b\sigma}$$

4.2 Trust Computation Using Probability of Delays at Intermediate Nodes

Suppose delay is caused by effects such as reprocessing, inserting of a new time stamp or re-packaging. The probability of delay can be computed as follows:

$$p_d = \frac{\tau_d}{\tau_m + z\sigma} \quad (8)$$

Where τ_d is the total time of delays, τ_m is the expected average delay time for a path and z as before is a

constant not more than 5. Trust is computed by equation (9).

$$\eta = (1 - p_d(k)) \quad (9)$$

When delays are encountered on a path for sometimes, probability of delays to happen is increased and trust for that path decreases. Following the pattern of analysis in the previous sections, it can be shown that the trust update function is:

$$\eta_u = p_d + \Delta p_d = \frac{\tau_d + \Delta\tau_d}{\tau_m + z\sigma} \quad (10)$$

and

$$0 \le \Delta\tau_d \le \frac{z\sigma}{\tau_m + z\sigma}$$

Delays are expected in paths in normal circumstances, whereas sporadic delays could be evidence of attacks. If the regular delays are highly structured, they are more meaningful and suspected to be caused by attack. An attacker is more likely to introduce delays and use them as a regular pattern that she can follow to link the sender and the receiver in a communication path supposed to be anonymous.

4.3 Trust Computation Using the Probability of Lost, Inserted and Multiplied Packets

The difference in number of sent packets and received packets can be noticed easily. That difference might be caused by loss of packets, inserted packets or multiplied packets. The probability of packets being lost, inserted and multiplied can be computed by the following equation:

$$p_n = \frac{\pi_{dn}}{\pi_{ns}} \quad (12)$$

Where π_{dn} is obtained by
NumberofReceivedpackets - NumbeofSentePackets

and π_{ns} is the number of sent packets. Trust can be computed as the probability of the difference in number of sent and received packets not occurring at time *K*.

$$\eta = (1 - p_n(k)) \quad (13)$$

Probability of packets lost is computed as

$$p_{loss} = \frac{\pi_{nl}}{\pi_{ns}} \quad (14)$$

Where π_{nl} is the number of lost packets between time *k-1* and time *k*. At time *k* trust is computed as it was done previously based on probability of packets not being lost.

$$\eta = (1 - p_{loss}(k)) \quad (15)$$

In general, we can apply a similar consideration to packet variations. Thus at time *k* probability of packets inserted π_{isn} and the probability of packet multiplied (π_{mul}) are computed using number of inserted packet (π_{nisn}) and number of multiplied packets (π_{nmul}) respectively in the same way it is done in equation (15). Trust is updated based on probability of packets inserted and on probability of packets multiplied respectively as it is done in sections 4.1.

4.4 Trust Computation Using the Probability of Normal Traffic

We define the probability of normal traffic transmission as the bit error rate of the path as:

P_t = *bit error rate route at normal traffic*

This means that, although bit error rate varies over time, congestion, abnormality and bit error rate vary abnormality over the time a link or a path when subject to attack.. Here too, we use $\varepsilon_{t(k)}$ as bit error rate of a link or a path at time instance k and $\varepsilon_{t(k+1)}$ at instance (k+1) respectively. If the bit error rate of a path increases, trust should decrease. Bit error rates are usually very small and increases in them is a cause for concern. The trust value based on bit error rate is:

$$\eta = (1 - p_\varepsilon(k)) \quad (16)$$

All computed trusts are combined in order to calculate overall trust of a path. Many probabilities for anomalies are related. For example, when probability of transit time variation increases, probability of delay also increases. When probability of congestion increases, probability of delay also increases. When probability of regular delay increases, probability of delay also increases. When probability of the difference in send and received number of packets increase, also probabilities of inserted, lost or multiplied packets increase. Because of these inter-relationship in increase or decrease of probabilities of anomaly to occur, we sum all trust derived from these probabilities so we have the overall trust value that does not decrease slowly and take a long time to reach the threshold while anomalies are continuously being detected on the path.

We defined threshold value of overall trust as the average of all possible maximum values of the trust expressions.

5. Results and Analysis

We assessed the effectiveness of the metric for detecting the anomalies in the network traffic and for detecting attacks that change the behavior of traffic patterns. We measured the aptitude of our trust metric to react to the change in the network traffic. We also

measured the level of failure of the metric and we call it false positives.

In all the experiments we explain in this paper we used four nodes labeled N1, N2, N3 and N4. For simplicity we always used N1 as source node and N4 as destination node and other two nodes were always used as intermediate nodes. N1 and N4 are two h5550 iPAQs with integrated 802.11b radio. N2 is an h3870 iPAQ equipped with Pretec PocketPC 802.11b compactWLAN. N3 is a PC equipped with Netgear Wireless PC card and a 32-bit CardBus. Capacity storage of the iPAQs were enhanced with MMC cards; two 64 MB and one 1 GB. The PC runs Linux; the Debian sarge with 2.4 Kernel and all the iPAQs run Linux FamiliarV0.8.2 with 2.4 kernels. The routing algorithm used was DSR.

All experiments were done indoors in our lab on a small area at a maximum of 6m between any two devices. Change of position of nodes was done by a random walk by individuals carrying the iPAQs at a speed of about 2m/s. In the following section we present some of the results.

Effectiveness in Detecting Regular Packets Drops: At N3, one packet is dropped after every 500 packets are forwarded to N4.

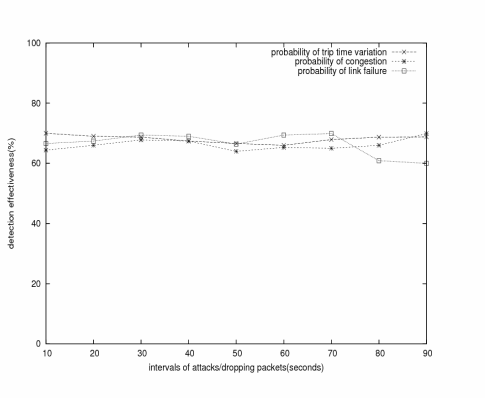
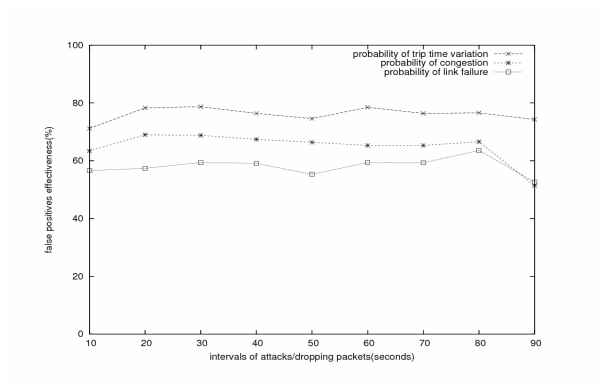


Figure 1: Comparison of Performance of Metrics for dropping packets

Figure 1 represents the comparison of the performance of the metrics on dropping packets with the expectation on the path. The vertical axis describes the effectiveness of the metric that is used to model attacks in the link. The attacks are instigated at regular intervals.

False positives are caused in the system as shown in Figure 2 in instances when there are dropped packets but not really caused by an attack. The system tends to think that at such times attacks have happened.



FiFiggure 2: False Positives for Dropped Packets

Effectiveness in Delay Detection: the performance of the metric for detecting delay in the path was demonstrated through experiments and results are shown in Figure 3. The Figure shows delays due to attacks as compared with delays when there are no attacks.

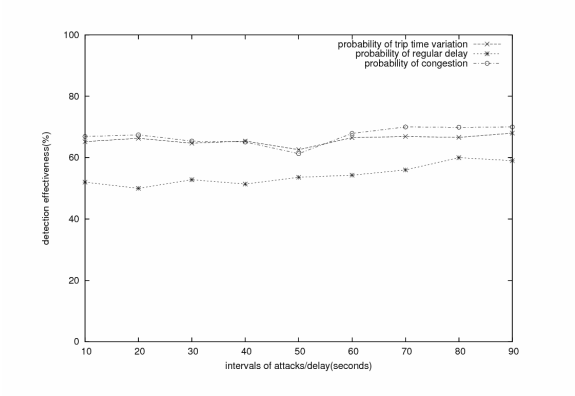


Figure 3: Effectiveness in Detecting Delays Due to Attacks

The graphs show that the metrics defined are effective in detecting delay attacks about 65% of the time. The lowest of the three graphs shows the system in this case has not performed better than about 60% of the time in detecting delay attacks.

False positives for delay attacks are insignificant for this metric and lie mostly below 20% of the time. Infact they are on average not more than 15% as shown in Figure 3.

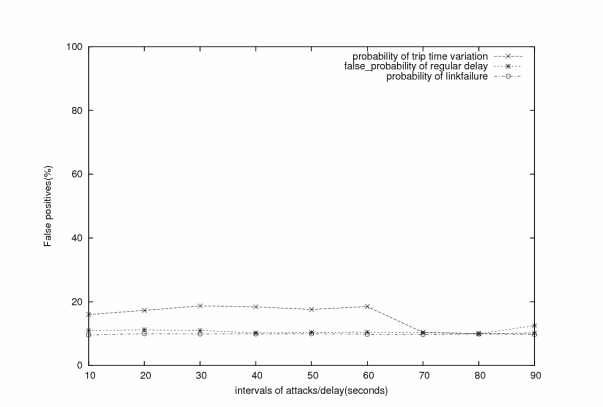


Figure 4: False Positives in Delay Attacks in Paths

6. Conclusion

This paper presented our work on developing a metric of trust for mobile ad hoc networks. The paper discussed how the characteristics of these networks contribute to the routing problem. Analytical review of existing trust models incorporated in routing protocols was presented and a novel solution was introduced.

Detailed descriptions of a new metric of trust and probability model to be used to measure trust were given. It is our intention to use traffic analysis techniques to collect statistics of communication pattern under benign as well as suspicious conditions. The metric is intended to distinguish between security attacks and benign link faults. It will be particularly useful in unobservable networks where nodes activities are not supposed to reveal any valuable information to outside observers.

Some results on the performance on the proposed metric with respect to delay and drop of packets attacks were presented.

In the next step of our work, we will implement our metric and conduct a performance analysis on detection of other attacks like multiplied packets, inserted packets and others. Security problems in mobile ad hoc networks are not yet fully addressed. More research into novel mechanisms for secure communication in such networks is necessary.

References

[1] Charles E. Perkins and Elizabeth M. Royer, "Ad hoc On-Demand Distance Vector Routing," *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, February 1999, pp. 90-100.

[2] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing*, T. Imielinski and H. Korth, Ed., Kluwer, 1996.

[3] K. Sanzgiri, B. Dahill, B. Neil Levine, C. Shields & E. M. Belding-Royer, "A Secure Routing Protocol for Ad Hoc Networks," In Proceedings of 2002 IEEE International Conference on Network Protocols (ICNP). November 2002.

[4] P. Papadimitrats and Z. J. Haas, "Secure Routing Mobile Ad hoc Networks," In Proceedings of the SCS Communication Network and Distributed Systems Modeling and Simulation Conference(CNDS 2002). January 2002.

[5] A. Khalili, J. Katz, and W.A. Arbaugh, "Toward Secure Key Distribution in Truly Ad-Hoc Networks," *2003 Symp. Applications and the Internet Workshops (SAINT 03 Workshops),* IEEE CS Press, 2003, pp. 342-346.

[6] B. Awerbuch et al., "An On-Demand Secure Routing Protocol Resilent to Byzantine Failures," *Proc. ACM Workshop Wireless Security,* ACM Press, 2002, pp. 21-30.

[7] A. Patwardhan, J. Parker, A. Joshi, A. Karygiannis and M. Iorga, "Secure Routing and Intrusion Detection in Ad Hoc Networks," Third IEEE International Conference on Pervasive Computing and Communications, Kauaii Island, Hawaii, March 8-12, 2005.

[8] S. Yi, P. Naldurg, R. Kravets, "A Security-Aware Routing Protocol for Wireless Ad Hoc Networks," ACM Symposium on Mobile Ad Hoc Networking & Computing (Mobihoc '01). October, 2001.

[9] S. Buchegger and J. Le Boudec, *"Performance Analysis of the CONFIDANT protocol: Cooperation Of Nodes Fairness In Dynamic Ad-hoc NeTworks,* " Proceedings of 3rd IEEE/ACM Symposium on Mobile Ad hoc Networking and Computing (MobilHOC), 2002, pp. 226 – 236

[10] A.A. Pirzada and C. McDonald, "*Secure Routing with the AODV Protocol*, " Proceedings 2005 Asia-Pacific Conference on Communications, October 03-05 2005, pp. 57 – 61

[11] A. Boukerche et al. "*SDAR: a secure distributed anonymous routing protocol for wireless and mobile ad hoc networks,* " Proceedings 29th Annual IEEE International Conference on Local Computer Networks, 2004. 16 – 18 Nov. 2004, pp. 618 -624

[12] P. Michiardi and R, Molva, "*CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks*, " Sixth IFIP conference on security communications, and multimedia (CMS 2002), Portoroz, Slovenia, Septe,ber 26 – 27, 2002, pp. 107 -121

[13] K. El-Khatib, L. Korba, R. Song and G. Yee, *"Secure Dynamic Distributed Routing Algorithm for Ad Hoc Wireless Networks"* icppw , p. 359, 2003.