Design and Analysis of Lightweight Trust Mechanism for Accessing Data in MANETs

Adarsh Kumar[1], Krishna Gopal[2] and Alok Aggarwal[3]
[1,2] Jaypee Institute of Information Technology, Noida 201301 - India
[1][e-mail: adarsh.kumar@jiit.ac.in]
[2][e-mail: krishna.gopal@jiit.ac.in]
[3] JP Institute of Engineering and Technology, Noida 201301 - India
[1][e-mail: director@jpiet.com]

*Corresponding author: Adarsh Kumar

Received August 30, 2013; revised November 26, 2013; revised January 15, 2014; accepted February 10, 2014; published March 31, 2014

Abstract

Lightweight trust mechanism with lightweight cryptographic primitives has emerged as an important mechanism in resource constraint wireless sensor based mobile devices. In this work, outlier detection in lightweight Mobile Ad-hoc NETworks (MANETs) is extended to create the space of reliable trust cycle with anomaly detection mechanism and minimum energy losses [1]. Further, system is tested against outliers through detection ratios and anomaly scores before incorporating virtual programmable nodes to increase the efficiency. Security in proposed system is verified through ProVerif automated toolkit and mathematical analysis shows that it is strong against bad mouthing and on-off attacks. Performance of proposed technique is analyzed over different MANET routing protocols with variations in number of nodes and it is observed that system provide good amount of throughput with maximum of 20% increase in delay on increase of maximum of 100 nodes. System is reflecting good amount of scalability, optimization of resources and security. Lightweight modeling and policy analysis with lightweight cryptographic primitives shows that the intruders can be detection in few milliseconds without any conflicts in access rights.

Keywords: Anomaly, lightweight, MANETs, trust aggregation, estimation, propagation, prediction.

A preliminary version of this paper appeared in QSHINE 2013, Jan 11-12, Greater Noida, India. This version includes a trust management integrated approach for contructing secure MANETs and outlier detection using automated toolkits.

http://dx.doi.org/10.3837/tiis.2014.03.024

1. Introduction

Due to ever increasing use of sensor based mobile devices for various applications like: household appliances, military purpose, virtual navigation, tele-geoprocessing appliances, tele-medicine, virtual navigation, vehicular networks etc. more is the demand of decentralized mechanism for mobile communication. MANETs can be constructed using similar decentralized approach with short range wireless technologies like: Bluetooth, Zigbee, WiFi etc. Sensor based MANET's devices are resource constraint devices with limited bandwidth, computing, storage, energy etc. Thus, lightweight primitives are required to perform the necessary operations. One major challenge is the scarcity of resources in MANETs that increases the security threats and requirements to integrate lightweight cryptographic aspects. Lightweight cryptography is classified as cryptographic primitives and protocols. Primitives are the procedure to secure network through encryption/decryption, digital signature, hashing, message authentication codes etc. Various models are proposed to provide complete cryptographic solution for any system like: Bell-LaPadula Model, McCumber Model, Orange Book etc. [2]. McCumber model is preferred as compare to other models to provide security relationship between devices and communications [2]. In order to achieve complete security for MANETs, various lightweight cryptographic primitives are taken into consideration on three axes: X-axis {Transmission (TR_A), Storage (ST_O), and Processing (PR_O)}, Y-axis {Confidentiality (CO_N), Integrity (IN_T) and Availability (AV_A)}, Z-axis {Human Factor (HF_A), Policy & Practices (PP_R) and Technology (TE_C)}.

In this work, Trust management based fine grained access control mechanism is designed for end users in resource constraint networks using lightweight symmetric key management in {TR_A- IN_T- HF_A} and {TR_A- AV_A - HF_A} pairs. Access control mechanism establishes relationships among nodes. These relationships are maintained through network policies which establish trust among nodes. Lightweight trust management based mechanism is processed through subgroup formation, trust computation, trust propagation, trust aggregation and trust evaluation life cycle. Nodes start forming subgroups or Frisbees at local level. These local groups are linked in a hierarchy through subgroup controller to form global view. Once a hierarchy is formed then trust of node is calculated through positive vibrations in centrality calculation. Centrality is the weighting factor of links between nodes to establish trust. Trust is propagated through multiple routes and aggregated at destination for duplicate values. Unknown trust score is predicted from historical data in evaluation phase. Further, the proposed mechanism is tested against attacks through outlier detection techniques. A mathematical analysis of bad mouthing attack and on-off attack is done and verified through Proverif toolkit.

The remainder of this paper is structured as follows. Section 2 summarizes existing work on key management, anomaly detection mechanism and trust management in fine grained access control. Section 3 describes the notation, symbols and definitions used in this work. In section 4, lightweight trust cycle with it's four components: trust computation, trust propagation, trust aggregation and trust prediction are proposed. Section 5 describes the simulation of proposed scheme with analysis of anomalies and protection from well known attacks. This section also shows the performance analysis of network with proposed lightweight trust model. Lastly, section 6 present conclusions.

2. Related Work

In 1919, Arvid Damm proposed the automatic key generation mechanism. These automatic key generation mechanisms can be classified as: (i) Symmetric and asymmetric, (ii) Hybrid key, (iii) ID-based threshold key management, (iv) Re-keying based mechanisms, (iv) Group communication mechanisms etc [3]-[7]. In sensor based MANETs, Group key management is efficient approach for user rights. Group keys can be managed through different group key management protocols. First category of these protocols are based on Diffie-Hellman mechanism. For example, Group Diffie Hellman (GDH): GDH.1, GDH.2, GDH.3, A-GDH (Authenticated-GDH), SA-GDH [8]-[9] etc. Major concentration in these protocols is drawn towards reducing the number of communication steps and exponentiation calculations. However, these protocols lacks in providing proper authentication and non-repudiation. Second type of protocols that enhances the security level through session key, renewing procedure of session key and non-repuration through private identification marks are general group key management protocols. For example, Group Key Management Protocol (GKMP), Group Secure Association Key Management Protocol (GSAKMP), Group Data of Interpretation (GDOI), Dunigan and Cao (DC), Hao-Hua-Chu (HHC), Burmester Desmedt Group Key Agreement (BD GKA) etc. [10]-[19]. Similarly another set of protocols developed to provide identification based non-repudiation are classified as ID-based group key management (IGKM). For example, Bonch & Franklin, Yu & Tang, Deng, Mukherjee and Aggarwal and Zhang, Liu, Lou and Fang [20]-[23]. Sensor based ad-hoc networks consist of resource constraint device. Thus, these devices require lightweight key management algorithm to be integrated. In [24], three group key management protocols for lightweight devices are identified and compared: Teo & Tan, WLH and Tseng's Protocol. It is found that Teo & Tan protocol perform better than other protocols in terms of security, delay and throughput. Further, a scheme is proposed over Teo & Tan protocol with virtual nodes to improve efficiency of network with similar quality of service parameters. In [1], Frisbee Model is integrated with Markov chain to minimize the losses of resource constraint devices with virtual nodes. Local View Formation Algorithm (LVFA) was integrated with Global View Formation Algorithm (GVFA) to calculate the anomaly score which help to find outliers in network.

After developing the group keys for users, the permissions to access network information is control through access control mechanisms. Access control mechanisms ensure that the user and information interactions are authorized to enable data sharing. Level of access rights help to measure the significance of data sharing. Mechanism like fine-grained access control is developed to clarify the controls. Fine-grained access control mechanisms can be classified as: (A) Attribute based techniques: (i) Single secret sharing scheme and (ii) Multi secret sharing scheme. Multi secret sharing scheme can be classified as: (a) Weighted Muti-Secret Sharing, (b) Polynomial based techniques, (c) Chinese remainder based techniques, (d) Hierarchical techniques etc. (B) Identity based techniques: Fuzzy identity based mechanism. (C) Role based techniques: Ontology-based role interaction access control. Inconsistency and incompleteness are the general properties to analyze policy. Schaad and Moffett proposed role based access control policy to check the constraint violations due to administration overhead [25]. Formal methods plays an important role to check the mistakes in defining the policies that may arise due to expressiveness property of policies [26]. Fisler *et. al.* [27] developed a Margrave tool to check the userspecifies properties of a policy. Alloy [28]-[30] and Margrave help to check duty constraints, roles, absence or presence permission and behavioral response from policy members. For example, subgroup member, controller, virtual member and controller are policy members in this work. Constraints among roles and responsibilities of these policy members is analyzed using these toolkits. Specifying and enforcing constraint in role based access control policies is necessary to enhace the security of such systems [31]. Multiple policies in one system may exhibit common or mutually exclusive properties. Conflicts among these policies is required to be checked and avoid to implement necessary security requirements [32]. In this work, margrave vocabulary and policy are designed for trust based policy analysis to put constraint for avoiding conflcits.

Trust must be established to provide the fine grained access control in sensor network. Description field of **Table 1** shows the permission access control sets used in this work for members. Trust is a subjective parameter and can be defined in various ways [33]-[35]. Various parameters that can be taken into consideration for trust evaluation are: expectancy, attitude, belief, reliability, availability, confidence etc. [36]-[37]. A trust management system consists of trust computation, trust propagation, trust aggregation, trust prediction and trust applications [37]. Trust computation can be classified as: (a) Distributed trust computations and (b) Centralized trust computations. Pirzada *et. al.* developed a reliability based dynamic trust computational method for pure ad hoc networks [38]. This is a centralized authority based mechanism for trust management. A centralized authrotiy failure could lead to major system fault, therefore decentralized approach is required to compute trust. Probst et. al. proposed a statistical distributed approach for trust computation [39]. Distributed approaches put dynamic topology challenge to ad hoc networks. Reports from neighboring nodes help to update trust among sensor nodes in a dynamic model proposed by Liu *et. al.* [40]. Major challenge in this dynamic network is scalability. Xiong *et. al.* integrated and evaluated the network performance through peer to peer communication [41]. Velloso et. al. proposed experience based upon dynamic maturity model for trust computation. Majority of trust based mechsnims are prone to attacks due to its objectivity, thus some evaluation schemes should be integrated to increase the security. Sun et. al. has integrated evaluation schemas to identify attack in such networks [42]-[44]. Dynamic trust based propagation methods are required to increase the network security. Cheng *et. al.* and Trifunovic *et. al.* proposed such social network based distributed trust propagation method [45]-[46]. Due to its computational complexity these mechanisms are infeasible for sensor networks. Quericia *et. al.* proposed lightweight trust propagation methods for sensor networks [47]. After trust propagation, it's value is aggregated at destination. Lightweight trust aggregation methods are proposed by Huang *et. al.*, Bachrach *et. al.* and Padro *et. al.* independently [48]-[50]. For some nodes multiple trust or no trust could reach at destination. Some prediction mechanism are required that could be based on past experience. Wang. *et. al.* proposed a generlized model for trust aggregation [51]. Jonker *et. al.* added the past experience to increase the unknown or duplicate values and Ham *et. al.* built reputation on past as well as weighted path values [52]-[53]. Predictions can be evaluated against attacks through outlier detection techniques. Outliers are the deviations of data from its regular data to ensure availability of network in $\{TR_A - AV_A - HF_A\}$. Outliers can be classified on different categories: (i) Node & Network based, (ii) Local, Global & Semi-global based, (iii) Error, event or attack based, (iv) Bayesian network based, (v) Nearest neighbor based, (vi) Spectral decomposition based, (vii) Statistical based mechanisms, (viii) Supervised & Unsupervised based, (ix) Distance, density, machine learning or soft computing based etc. [54]-[58]. There is a need to use lightweight mechanism for finding an error in sensor based ad-hoc networks. Traag *et. al.* proposed a Markov chain based technique to distinguish between an event or error for mobile phones [59]. For MANETs, modifications over this technique is prepared and integrated with Teo & Tan's protocol for anomaly score calculation [1]. Rights to symmetric key for accessing important data can be constrained using access control mechanisms and policies after detection of anomaly in network.

3. Definitions and Notations

3.1 Definitions

Definition 1: (Trust [37]): Trust is an honest behavior or positive vibration sent to gain access to secret data. It is a subjective measure based on reliability, availability, confidence, quality of service, risk, accuracy, repudiation etc.

Definition 2: (Access set 'y' [60]): A collection of mobile sensor nodes y=$SM_{(j,k)}^{HL_i}$, where, j,k ϵ {1,2,3….n}, those are given rights to access $P_{SM_{(j,k)}^{HL_i}}$ on secret data. Participants of 'y' are known as an authorized users and the participant not in 'y' are called as an unauthorized users.

Definition 3: (CENTRALITY: $CENT^{E_i}$ $\epsilon\{CENT_{+}^{E_i}, CENT_{-}^{E_i}\}$): Centrality of an edge is defined as probability of any mobile sensor node $SM_{(j,k)}^{HL_i}$ to follow a particular path. A node can follow a different path in dynamic topology based networks. Markov path chain help to find probability of following a particular path based on hidden states. Positive ($CENT_{+}^{E_i}$) and negative ($CENT_{-}^{E_i}$) values of centrality are based on anomaly score. A path with detection of outliers is considered as negative. Otherwise, It will be positive.

3.2 Symbols &Notations

Table 1 shows the symbols and notations used in this work.

Table 1. Symbols and Notations

Symbol	Quantity	Description
R_S	Role Score	
P_S	Primary permission set	P_S ={$P_{SC_j^{HL_i}}$, $P_{SM_{(j,k)}^{HL_i}}$, $P_{VNSM_{(j,k)}^{HL_i}}$, $P_{VNSC_j^{HL_i}}$ }
$SG_{SC_1}^{HL_i}$	jth subgroup controller at ith hierarchical layer.	
$SM_{(j,k)}^{HL_i}$	kth subgroup member of jth subgroup controller at ith hierarchical layer.	
$VNSC_j^{HL_i}$	jth virtual node subgroup controller at ith hierarchical layer.	
$VNSM_{(j,k)}^{HL_i}$	kth virtual node subgroup member of jth subgroup controller at ith hierarchical layer.	
$P_{SC_j^{HL_i}}$	Permission set of jth subgroup controller at ith hierarchical layer.	$P_{SC_j^{HL_i}}$ ={READ, WRITE, ACCESS, USE, MODIFY}
$P_{SM_{(j,k)}^{HL_i}}$	Permission set of kth subgroup member of jth subgroup controller at ith hierarchical layer.	$P_{SM_{(j,k)}^{HL_i}}$ ={READ, ACCESS, USE}
$P_{VNSM_{(j,l)}^{HL_i}}$	Permission set of lth virtual node subgroup member of jth subgroup controller at ith hierarchical layer.	$P_{VNSM_{(j,l)}^{HL_i}}$ ={READ, ACCESS, USE}
$P_{VNSC_j^{HL_i}}$	Permission set of jth virtual node subgroup controller at ith hierarchical layer.	$P_{VNSC_j^{HL_i}}$ ={READ, WRITE, ACCESS, USE, MODIFY}
$S_{ADDRESS}$	Source Address	
$D_{ADDRESS}$	Destination Address	
$IDSM_{(j,k)}^{HL_i}$	Unique identification of kth sub-group member of jth subgroup controller at ith	

	hierarchical layer.
$V^{SM_{(j,k)}^{HL_i}}$	Vertex of kth subgroup member of jth subgroup controller at ith hierarchical layer.
$E_{SM_{(l,k)}^{HL_i}}^{SM_{(j,k)}^{HL_i}}$	Edge from $SM_{(j,k)}^{HL_i}$ to $SM_{(l,k)}^{HL_i}$
G_E^V	Graph constructed using V and E

4. Proposed Methodology

4.1 Frisbee Construction

In order to reduce losses, “Frisbee Model” is used to construct local zones as MANETs are having scarcity of resources [61]. Therefore, Frisbees are formed using trust establishment. If each node’s trajectory is observed and attendance of an event is marked then its trust value increases. Probability $P_{(i,\ j)}$ of any mobile node MN_x to move from $MN_z^{(x_i,y_i)}\ to\ MN_z^{(x_z,y_z)}$ using Markov chain through states $s_1^{(x_1,y_1)}$, $s_1^{(x_1,y_1)}$ …. $s_n^{(x_n,y_n)}$, where zϵ {1,2,3….n}, is calculated as:
P($s_1^{(x_1,y_1)}$, $s_1^{(x_2,y_2)}$ ….. $s_n^{(x_n,y_n)}$) = $s_1^{(x_1,y_1)}$, $s_1^{(x_2,y_2)}$ ….. $s_n^{(x_n,y_n)}$ = P($s_1^{(x_1^{'},y_1^{'})} = s_1^{(x_1,y_1)}$)$p_{x_1x_2}p_{x_2x_3}\cdots\cdot p_{x_{n-1}x_n}$ =$P_{S.}$ If routing and communication states are integrated then probability can be calculated as:
P_S = P(($S_{MOBC_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$), ……, ($S_{MOBC_n}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_n}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$)) = P ($s_1^{(x_1,y_1)}$ = ($S_{MOBC_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$)) $p_{x_1x_2}p_{x_2x_3}\cdot\cdot p_{x_{n-1}x_n}$.

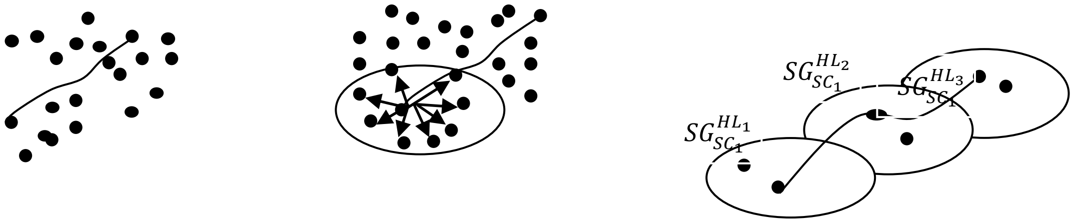


Figure 1a: Possible trajectory using Markov model

Figure 1b: 1-hop nearest neighbor sensing Frisbee formation

Figure 1c: Sequence of Frisbees formed during trajectory

Figure 1: Frisbee formation during LVFA

Probability of following a particular path that will proceed to regular event region ‘R’ is calculated as: $P_S^{AVG} = (1/(W-1)\sum_{v=1,v=w}^{W} P_s(MN,R,T_{WIN}^v)$, Where P_S^{AVG} is average value of P_S.According to Markov chain, every next state is dependent upon subsequent states. Thus probability of subsequent regular event can be calculated as:

$$P_S^{AVG} = (\frac{1}{W-1})(\sum_{v=1,v=w}^{W}(S_{MOBC_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))})\ .\ p_{x_1x_2}p_{x_2x_3}\cdot\cdot p_{x_{n-1}x_n},\ \mathrm{R},\ (T_{WIN}^{T_S}\ldots\ldots\ldots T_{WIN}^{T_S})_{\mathrm{v}})$$

Fig. 1. shows the Frisbee formation process at local level. Figure 1a and figure 1b show the trajectory path of single node which leads to single hop nearest neighbour sensing Frisbee formation. As shown in figure 1c, If Frisbee formation process is continued then the sequence of Frisbees are formed and each have its own subgroup controller.

4.2 Lightweight Trust Computation

Once the probability of a node to follow a particular path is calculated, it's trust value can be passed along with other nodes in a particular Frisbee using distributed or centralized computational methods. Distributed methods can be classified as: Neighbor sensing, Recommendation based and Hybrid methods. Centralized method is a trust agent based method [37]. **Fig. 1b** shows the single hop nearest neighbor sensing Frisbee formation. Algorithm 1 describes the behavior trust formation based on routing packets.

Algorithm 1: Trust Formation using routing behavioral characteristics.
Premises: $S^{i}_{NEIGH}$is a set of neighboring nodes of node i. Let $\mu \epsilon$ $\{\mu^{+}, \mu^{-}\}$ be the set of positive acknowledgement (μ^{+}) and negative acknowledgement or no acknowledgement (μ^{-}). Let μ^{+} consists of two set values {$D_{ADDRESS}$, $IDSM^{HL_i}_{(j,k)}$ }. Let 'W^{E} ' be the weight assigned to edge E. $CENT^{E_i}$ can have two values $\{CENT^{E_i}_{+}, CENT^{E_i}_{-}\}$ rand it epresents the centrality score of edge E_i, which is a subset of positive and negative centrality value.

1. $SM^{HL_i}_{(j,k)}$ senses $SM^{HL_i}_{(j+1,k)}$, $SM^{HL_i}_{(j+2,k)}$,……..…, $SM^{HL_i}_{(j+n,k)}$. Where n is total number of neighboring nodes (S^{i}_{NEIGH}).
2. After determining the probability P^{AVG}_{S} of $SM^{HL_i}_{(j,k)}$ in following the particular path, packets are forwarded to establish a route.
3. If ($\mu^{-} > \mu^{+}$) then anomaly score is calculated as:

   Anomaly Score = ($MN^{Attendee}_{Active}$ - ($AVG^{Attendee}_{(MN_{ACTIVE}+MN_{SLEEP})}$)) / STDEV

   $MN^{Attendee}_{Active}$ is an active presence of mobile nodes and $AVG^{Attendee}_{(MN_{ACTIVE}+MN_{SLEEP})}$represents total nodes including active and sleeping nodes.
4. If Anomaly Score < 4 then trust transformation can be processed as:

   a. Eigen_Trust_Transformation($V^{SM^{HL_i}_{(j,k)}}$, $E^{SM^{HL_i}_{(j,k)}}_{SM^{HL_i}_{(l,k)}}$)

      i. If G^{V}_{E} be the graph constructed from $V^{SM^{HL_i}_{(j,k)}}$ & $E^{SM^{HL_i}_{(j,k)}}_{SM^{HL_i}_{(l,k)}}$. Here, values of 'i' and 'k' are fixed and jϵ\{1,2,3….n\}.
      ii. Calculate CENTRALITY for each edge using probability of a node to following a particular path.
      iii. Construct a single edge directed graph and calculate:
      W^{E} = MAX (0;$\sum CENT^{E_i}$), iϵ\{1,2,3….n\}. This W^{E} is the trust value of edge $E^{SM^{HL_i}_{(j,k)}}_{SM^{HL_i}_{(l,k)}}$.

   b. Beta Transformation($V^{SM^{HL_i}_{(j,k)}}$, $E^{SM^{HL_i}_{(j,k)}}_{SM^{HL_i}_{(l,k)}}$)

      i. Step (i) and (ii) are same as in Eigen_Trust_Transformation.
      ii. Construct a single edge directed graph and calculate:

$$W^{E} = (\sum CENT_{+}^{E_i}) / (\sum CENT_{+}^{E_i} + \sum CENT_{-}^{E_i}),\ i\epsilon\{1,2,3\ldots.n\}.$$

c. Distance Method

i. Step i and ii are same as in Eigen_Trust_Transformation.

ii. Construct a single edge directed graph and calculate:

W^{E} = | Distance from $SM_{(j,k)}^{HL_i}$ to $SM_{(l,`k)}^{HL_i}$ |

d. Signal Strength Method

i. Step i and ii are same as in Eigen_Trust_Transformation.

ii. Construct a single edge directed graph and calculate:

W^{E} = | Signal Strength between $SM_{(j,k)}^{HL_i}$ to $SM_{(l,`k)}^{HL_i}$ |

Example: In order to understand the trust computation process, let take an example of graph ‘G’ with possibility of multiple vertices between edges in a local subgroup as shown in **Fig. 2**. **Fig. 3** shows the resultant graph of **Fig. 2**. Value of weights in **figure 3** varies according to centrality calculation method. If **Fig. 2** and **Fig. 3**’s graphs are taken as directed graphs then **Table 2** shows the directions and their values. Magnitude of negative values is considered for calculation as it is assumed in distance and signal calculation methods that there is no negative value. **Table 3** shows the maximum values of W_1, W_3 and W_6 , which provides better trust by taking negative centrality values into consideration. It can also be considered as a good method because remaining weight values are similar to values of other methods. Eigen trust transformation is second good method as compared to distance and signal strength. Thus distance and signal strength method will not always provide good trust transformations.

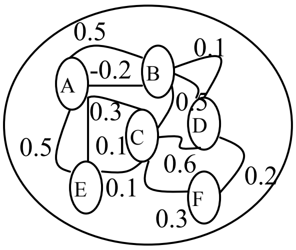


Fig. 2. Weighted Directed graph to calculate trust

Table 2. Centrality values for the graph.

Src. to Dest.	CENTRALITY Score
A→ B	{0.5,-0.2}
A→E	{0.5,0.1}
A→C	{0.3}
E→C	{0.1}
C→B	{0.5}
C→F	{0.3}
C→D	{0.6}
F→D	{0.2}
B→D	{0.1,0.5}

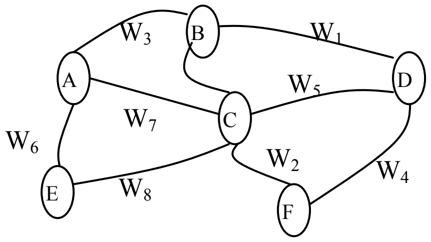


Fig. 3. Single Edge Transformed Weighted Directed graph to calculate trust

Table 3. Trust computation using different methods

	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8
Eigen_Trans	0.6	0.3	0.3	0.2	0.6	0.6	0.3	0.1
Beta_Trans	0.8	0.3	0.7	0.2	0.6	0.8	0.3	0.1
Distance	0.1	0.3	0.2	0.2	0.6	0.1	0.3	0.1
Signal_stren.	0.5	0.3	0.5	0.2	0.6	0.5	0.3	0.1

4.3 Lightweight Trust Propagation

Propagation of trust and anomaly values is performed using hierarchical trust formation. If anomaly score exceeds a threshold value then that node is considered as outlier. Further, its value can be transmitted to topmost subgroup controller through other subgroup controllers at different layers in order to form a global view. Similarly, trust value is also passed. Algorithm 2 describes the trust and anomaly score propagation among subgroups in a hierarchy.

Algorithm 2: Local trust collection and anomaly detection
Premises: Let HL_i be the hierarchy of subgroup with height 'h'.
Goal: To collect anomaly scores and trust values. Securely propagate these values to subgroup controller at next hierarchical layer.

Step 1: Subgroup controller collects anomaly and trust values.

a. $SG_{SC_j}^{HL_i}$ collects W^E from every edge $E_{SM_{(l,k)}^{HL_i}}^{SM_{(j,k)}^{HL_i}}$ using Burmester & Demesdt protocol (BD protocol)[62]. It also collects anomaly score from mobile nodes $V_{SM_{(l,k)}^{HL_i}}^{SM_{(j,k)}^{HL_i}}$.

b. According to Markov chain, trajectories to be followed by mobile node to participate in an event using formula:

$P_S = P(s_1^{(x_1',y_1')} = s_1^{(x_1,y_1)}) p_{x_1x_2} p_{x_2x_3} \cdots\cdots p_{x_{n-1}x_n}$
The best path is selected (i.e. when P_S approaches 1)

c. $SG_{SC_j}^{HL_i}$ generates a score packet H{$W_j^{E_i}$, Anomaly_score}, where, H is a PHOTON lightweight cryptographic hash function.

d. $SG_{SC_j}^{HL_i}$ forwards this packet to next layer's subgroup controller $SG_{SC_j}^{HL_{i+1}}$.

Step 2: Subgroup Controller passes the score packet to next layer subgroup controller through most trusted node.

a. $SG_{SC_j}^{HL_i}$ selects most trusted subgroup member $SM_{(j,k)}^{HL_i}$ that is close to next layer subgroup.

b. {score packet || H{$W_j^{E_i}$, Anomaly_score}} is send to $SG_{SC_j}^{HL_{i+1}}$ through most trusted $SM_{(j,k)}^{HL_i}$.

Step 3: Subgroup Controller at 'i+1' layer collects score packets from i^th layer.

a. $SG_{SC_j}^{HL_{i+1}}$ collects {score packet || H{$W_j^{E_i}$, Anomaly_score}} from $SM_{(j,k)}^{HL_i+1}$ using BD protocol. This $SM_{(j,k)}^{HL_i+1}$ is the most trusted subgroup member close to $SG_{SC_j}^{HL_{i+1}}$ and i^th layer subgroup.

b. Score at (i+1)^th hierarchical layer is collected as:
$SCORE^{HL_i} = \{ \{score\ packet \mid\mid H\{W_j^{E_i}, Anomaly\_score\}_i^1\}, \{score\ packet \mid\mid H\{W_j^{E_i}, Anomaly\_score\}_i^2\} \ldots. \{score\ packet \mid\mid H\{W_j^{E_i}, Anomaly\_score\}_i^n\ \}\}.$

c. After getting score packets these values are subsequently passed to top most $SG_{SC_j}^{HL_{i+n}}$.

4.4 Lightweight Trust Aggregation

Hierarchical group formation proposed in this work make it proficient enough to handle multiple trust values received from different locations. Although trust aggregation is not mandatory for checking multiple values but execution of trust accumulation in this can be performed through different ways: (i) Sequential Aggregation, (ii) Conditional sequential aggregation, (iii) Parallel Aggregation and (iv) Parallel loop aggregation [37][63]. Proposed hierarchical mechanism can be extended with trust aggregation schemes. As shown in Fig. 4, this extension is required for nodes that are away from $SG_{SC_j}^{HL_i}$ with more than 1-hop to avoid duplicates. Algorithm 3 describes the method of trust accumulation in trust aggregation.

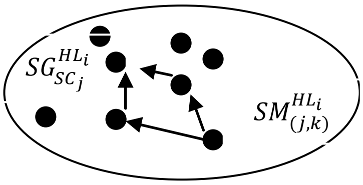


Fig. 4. Trust aggregation at local subgroup level.

Algorithm 3: Trust aggregation at local and global level.
Goal: Design a local trust aggregation (LTA) and Global trust aggregation (GTA) functions to avoid duplicate communication. Primary GTA (PGTA) is the trust score in main group.
Method 1: Sequential Trust Aggregation (STA)

Step 1: Every subgroup member $SM_{(j,k)}^{HL_i}$ passes its W^E value to nearby trusted nodes in order to transmit the value to $SG_{SC_j}^{HL_i}$.
Step 2: Every subsequent node will aggregate this value in LTA function. LTA received at $SG_{SC_j}^{HL_i}$ will contain following values:
LTA$_1$ = { W_1^E, W_2^E } , LTA$_2$ ={ W_2^E, W_3^E, W_4^E}, ……………..LTA$_n$= {W_n^E , W_{n+2}^E, W_{n+3}^E ….}
Step 3: In order to avoid duplicate packets, identification marks of nearby nodes to subgroup controller can be added and form the final LTA list at ithlayer in jth group as:

$$LTA_j^{HL_i}= \{IM_1||LTA_1, IM_2||LTA_2, IM_3||LTA_3 , …. , IM_n|| LTA_n\}$$

Step 4: $LTA_j^{HL_i}$, where j$\epsilon\{1,2,\dots n\}$ is passed to HL$_{i+1}$ layer to form a global trust aggregation.
$GTA_j^{HL_{i+1}}$={ $LTA_1^{HL_i}$, $LTA_2^{HL_i}$……………..$LTA_n^{HL_i}$}
Step 5: $GTA_j^{HL_i}$ values are passed to (i+1)th , (i+2)th layers and finally reaches to (i+n)th layer.

Method 2: Conditional Sequential Trust Aggregation (CSTA)
Goal: Condition of mirror values is checked at nodes closer to $SG_{SC_j}^{HL_i}$ in order to remove looping in sequential trust aggregation.
Step1 to step3 are same as in method 1.
Step4: Check and remove mirror values in $LTA_j^{HL_i}$.

```
for ID||trust in $LTA_j^{HL_i}$:
    for weigh in trust:
        for ID||trust in ($LTA_{j+1}^{HL_i}$,1, $LTA_{j+n}^{HL_i}$):
```

```
for weight_n in trust:
    if weight.equals(weight_n)
        trust.remove(weight)
        for  ID||trust in $LTA_j^{HL_i}$:
            for i in (0,1,n):
                for j in (0,1,n):
                    if ID||trust$_i$ > ID||trust$_{j:}$
                        y.append(ID||trust$_i$)
                    else
                        y.append(ID||trust$_i$)
```

Step 5 & 6 will be same as step 4 & 5 of method 1. Since duplicate values are checked at local level therefore there is no need to check at global level.

Method 3: Parallel Trust Aggregation (PTA)
Step 1: Every subgroup member $SM_{(j,k)}^{HL_i}$ passes its W^E value to nearby trusted nodes in order to transmit it's value to $SG_{SC_j}^{HL_i}$.
Step 2: Every subsequent node will aggregate this value to LTA. LTA received at $SG_{SC_j}^{HL_i}$ may contain duplicate values of trust:

$LTA_1 = \{ W_1^E, W_2^E \}$, $LTA_2 =\{ W_2^E, W_3^E, W_4^E \}$, ……………..$LTA_n= \{W_n^E , W_{n+1}^E, W_{n+2}^E, \ldots\}$
Step 3: Check and remove mirror values in $LTA_j^{HL_i}$.

```
for  ID||trust in $LTA_j^{HL_i}$:
    for weight in trust:
        for  ID||trust in ($LTA_{j+1}^{HL_i}$,1, $LTA_{j+n}^{HL_i}$):
            for weight_n in trust:
                    if weight.equals(weight_n)
                            trust.remove(weight)
```

Method 4: Parallel Loop Aggregation (PLA)
Step 1: Every subgroup member $SM_{(j,k)}^{HL_i}$ passes its W^E value to nearby trusted nodes in order to transmit it's value to $SG_{SC_j}^{HL_i}$. If some $SM_{(j,k)}^{HL_i}$ receive back it's aggregate value in the list then it will run following procedure to remove duplicates

```
for  ID||trust in $LTA_j^{HL_i}$:
    for weight in trust:
        for  ID||trust in ($LTA_{j+1}^{HL_i}$,1, $LTA_{j+n}^{HL_i}$):
            for weight_n in trust:
                if weight.equals(weight_n)
                  trust.remove(weight)
```
Step 2: Non duplicate values are aggregated in the list as:

$$LTA_j^{HL_i} = \{IM_1||LTA_1, IM_2||LTA_2, IM_3||LTA_3 , \ldots. , IM_n|| LTA_n\}$$

Step 3: Same as step 4 and step 5 of Method 1.

4.5 Lightweight Trust Prediction & Evaluation

Trust prediction methods are used to potentially calculate trust values of nodes based on present and past behaviors [37]. In algorithm 1, trust is predicted based on probability of following a path and anomaly score. Therefore, no extra mechanism is required to calculate trust of unknown nodes based on certain behavior. Anomaly analysis and protection from attacks is done in next section to evaluate the proposed trust system.

5. Simulation and Analysis

5.1 Anomaly Analysis

Simulation of this work is done using ns-3 simulator on Linux platform [64]. Variation of 50 to 200 nodes is done with different anomaly detection parameters: Anomaly detection ratio (ADR), Wrongly calculated anomaly ratio (WCAR), Average local anomaly detection ratio (ALADR) and Average local wrongly calculated anomaly ratio (ALWCAR)[1]. **Table 4** shows the analysis of various ratios.

Table 4. Different detection ratios to calculate success rate.

	N=50	**N=100**	**N=200**
ADR	0.860	0.770	0.700
WCAR	0.010	0.060	0.090
ALADR	0.910	0.800	0.740
ALWCAR	0.001	0.009	0.011

Observation 1: It is observed that with the increase in number of nodes, the ADR decreases and WCAR increases. It is observed that these changes are due to increase in trust level with increase in number of nodes therefore some virtual programmed nodes are added in each subgroup. These programmed nodes will try to gain maximum trust of other nodes with maximum probability of acting as outlier to disgruntle the network user access. Algorithm 4 represents the programmed concept to make virtual nodes.

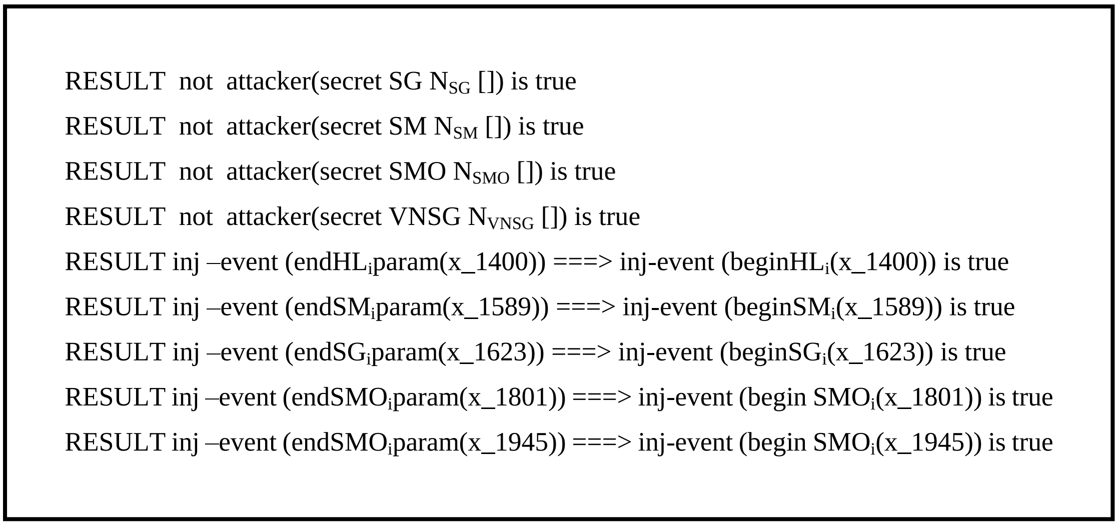
Algorithm 4: Programmed virtual node to add anomaly with trust satisfaction.
Goal: To observe the reasons of decrease in ADR with increase in number of nodes.
Premises: $VPSM_{(j,k)}^{HL_i}$ are the virtual programmed subgroup members to act as outliers. $VPW_j^{E_i}$ is the trust score of virtual programmed node.
Step 1: Make some $VPSM_{(j,k)}^{HL_i}$ nodes in every subgroup. These nodes will try to increase their $W_j^{E_i}$ value with minimum anomaly score.
Step 2: $VPSM_{(j,k)}^{HL_i}$ will be able to get access to network services with trust value $W_j^{E_i}$, if it gets new $W_j^{E_i}$ equals to $W_j^{E_i}$.
Step 3: $VPW_j^{E_i}$ will be increased by virtual programmable nodes with their self motivation.

Table 5 shows the results of anomaly detection when trust level of virtual programmable nodes is varied. If trust level is high then ADR decreases with increase in number of nodes but if trust is low then ADR increases with increase in number of nodes. Similar results are observed with WCAR also. Thus it can be observed that with the increase in number of nodes ADR is strongly dependent on trust level. High trust level signifies that the proposed work is well suited for large scale network and with minimum anomalies. Whereas low trust increases the chances of unauthorized access.

Table 5. Anomaly scores at different trust levels.

	Trust = HIGH			**Trust = LOW**		
	N=50	**N=100**	**N=200**	**N=50**	**N=100**	**N=200**
ADR	0.750	0.670	0.600	0.960	0.965	0.980
WCAR	0.007	0.059	0.088	0.009	0.008	0.010

Observation 2: It is also observed that ADR ratio at global level is having errors as compared to local level i.e. ALADR. Thus it can be said that these error happens because of (i) Communication barrier or (ii) Attacks. In order to remove the barrier in secure transmission, correction in local algorithm is made in previous work [1]. In this work, proposed algorithms are tested against different attacks: (i) Bad Mouthing Attack and (ii) On-off Attack [65]. **Fig. 5** verifies the protection of system from discussed attacks using ProvVerif automated verification tool.



RESULT not attacker(secret SG N_{SG} []) is true

RESULT not attacker(secret SM N_{SM} []) is true

RESULT not attacker(secret SMO N_{SMO} []) is true

RESULT not attacker(secret VNSG N_{VNSG} []) is true

RESULT inj –event (endHL$_i$param(x_1400)) ===> inj-event (beginHL$_i$(x_1400)) is true

RESULT inj –event (endSM$_i$param(x_1589)) ===> inj-event (beginSM$_i$(x_1589)) is true

RESULT inj –event (endSG$_i$param(x_1623)) ===> inj-event (beginSG$_i$(x_1623)) is true

RESULT inj –event (endSMO$_i$param(x_1801)) ===> inj-event (begin SMO$_i$(x_1801)) is true

RESULT inj –event (endSMO$_i$param(x_1945)) ===> inj-event (begin SMO$_i$(x_1945)) is true

Fig. 5. ProvVerif results showing passing of all tests

Attack 1: Bad Mouthing Attack.

Description: Trust evaluation is strongly dependent on response from others. This response can be judged from [negative, positive] or [high, low] values. Some node can show fraudulent behavior in order to gain advantage or provide benefit to favorable group of nodes. During fraudulent behavior, nodes can intentionally take benefits in terms of: (i) Trust computation and assigning high or low value to one or a group of nodes. If a malicious node wants to incorporate denial of service attack, provide malicious services, create a central point of attack etc. then a high trust value is assigned. But if malicious node wants to drive some honest nodes out of the subgroup, reduces the CENTRALITY value etc. then low trust value is assigned. (ii) Provide different trust response to different set of groups. A negative discrimination means providing good service to all except few. For example, providing good trust value to existing subgroup members but lesser value to new subgroup members coming from other subgroups with high confidence. A positive discrimination means providing good trust service to majority and average to some serving nodes. For example, providing high trust value to existing subgroup members except average trust value to boundary cases. It may be because subgroup controller is not having confidence over those nodes [66].

Background: Various techniques used to remove these attacka are: provide controlled anonymity, incorporating cluster filtering, channel aware detection algorithm [66]-[71].

Proposed System Protection: The proposed system is secures from Bad Mouthing Attack as:

I. Trust recommendation is based on CENTRALITY score, that is a probabilistic approach to calculate trust. Since it is not behavior or recommendation based, trust action is strongly dependent on probability of following a path and independent of recommendation. As a result, anonymity provide false trust which does not exist.

II. Positive centrality packets with P_S nearby 1 are forwarded to subsequent nodes connected with edge (E_i). Positive CENTRALITY score and Markov chain increases the trust over a node during propagation.

III. W^E is an additional parameters to believe and trust. Probability of following a path to attend an event and anomaly score can give intuition about trust on a node even if W^E score is low.

Proposed system protection can be analyzed by checking the system against fault acceptance probability (FAP).

FAP = Probability[High $\mathrm{CENT}_{+}^{\mathrm{E_i}}$] + Probability[following path as calculated by P_S] = Probability[High value of Anomaly Score or high value of $MN_{Active}^{Attendee}$ or high value of $AVG_{(MN_{ACTIVE}+MN_{SLEEP})}^{Attendee}$] + Probability [(($S_{MOBC_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$) , ...($S_{MOBC_n}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_n}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$)) ==1]

= Probability [Acceptable value of (MAX (0;$\sum CENT^{E_i}$), iϵ{1,2,3….n}) or (($\sum CENT_{+}^{E_i}$) / ($\sum CENT_{+}^{E_i}$+$\sum CENT_{-}^{E_i}$)), iϵ{1,2,3….n}) or (| Distance from $SM_{(j,k)}^{HL_i}$ to $SM_{(l,`k)}^{HL_i}$ |) or (| Signal Strength between $SM_{(j,k)}^{HL_i}$ and $SM_{(l,`k)}^{HL_i}$ |)] + Probability [(($S_{MOBC_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_1}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$), ...($S_{MOBC_n}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}||S_{MOBROU_n}^{((x_1^i,y_1^i)....(x_1^n,y_1^n))}$)) ==1].

Probability of fault acceptance of proposed system is depedent on behavior of nodes which includes distance among nodes, signal strength of nodes, movement of nodes, routing and communication capabilities of nodes, number of neighboring active and sleeping nodes and trust weight between target node and neighboring nodes. Hence it can be assumed that system is propected against the attack until threshold value of anomaly detection is under threshold and behavior factor of target nodes are taken into consideration.

Attack 2: On-Off Attack

Description: Due to dynamic nature of trust, Node may follow different paths to attend an event. At time t_1, it may show positive vibrations to follow a particular path but at time t_2 it can show negative vibrations to follow original however positive vibration to different path attends the same regular event and probability value decides the path in this work. Higher probability value and low anomaly score determines the chance of a node to follow a particular path. There may be deviation in following a particular path because of side channels like:

environment, voltage fluctuation etc. The bearable amount of deviation is considered as forgetting factor. In this work, forgetting factor is calculated as: $CENT_{+}^{E_i}/\,(CENT_{+}^{E_i}+CENT_{-}^{E_i})$.
Background: Adaptive forgetting scheme is proposed to remove on-off attack [65].

Table 6. Trust Aggregation Methods for on-off attack

	STA	CSTA	PTA	PLA
Loop Free	N	Y	Y	Y
Conditional Checking	N	Y	Y	Y
Overwriting Avoidance	N	N	Y	Y

N=NO, Y=YES

Proposed System Protection: The proposed system is secured from on-off attack because the proposed system provides the feasibility to decide the path with high value of trust aggregation. Four methods of trust aggregation are integrated from literature [37][63]. These methods provide the loop free, conditional checking and overwriting avoidance features to trust in trust aggregation phase. Most importantly, these values are passed through subgroup controller, which is assumed to be the high energy trust node. Even if some node at time t_1 shows different trust than at time t_2, subgroup controller can boost the trust by passing it's trust value of t_1. Table 6 shows the comparative analysis of trust aggregation methods to remove on-off attack. FAP against on-off attack = Probability [following a path calculated in P_S] = Probability [high value of path calculated in P_S] + Probability[deviation]. This deviation value is an acceptable change of path. FAP against on-off attack = Probability [maximum time following the same path] + Probability [deviation] = (1-Probability(maximum time following new path)) + Probability [deviation]. Now if 'N' communications are made by some target node then FAP against on-off attack can be calculated as: (1- (N/N+(N-1)/N+(N-2)/N+(N-3)/N+ ……(N-M+1)/N) + Probability[deviation]. Here M is minimum acceptable limit of existing paths. According to birthday paradox, complexity of following a different path is represented as: $e^{M/2}$. Hence probability of following same path is high if node is honest.

5.2 Lightweight Analysis

5.2.1 Lightweight Modeling and Analysis

Various formal method analysis based languages are available to perform software abstraction succinctly and efficiently. For example: B, Z, VDM, Alloy etc. [28]-[30]. Alloy is designed to have lightweight analysis rather than concentrating on proof and it provides powerful, small and simple design, automatic and animation analysis with fewer concepts than other languages. Alloy Analyzer is simulation and checking tool to analyze lightweight relationships for Alloy models. Table 7 shows the analysis of automatic subgroup controller, subgroup member and intruder alloy model. In this analysis, variation of number of subgroup controllers, subgroup members and intruders entities are analyzed to find the values that are acceptable for lightweight relationships. In preliminary analysis, relationships are analyzed for 1, 5 and 10 numbers of each entity using proposed trusted and basic strategies. Here, Basic strategy is implementation of identification, authentication, grouping and ownership transfer without proposed trust management cycle. Table 7 shows the time and number of steps required to find intruders in both strategies. Minimum of 14 steps in 23 msec. are required to find single intruder in trusted strategy as compared to 11 steps in 22 msec. for basic strategy in presence of 1 subgroup controller, 1 subgroup member and 1 intruder. Where, step is number of packet checker communications made to find intruder. With increase in any entity, the time and steps increases. This increase is 5 times more if subgroup controller or members are 5 more than intruders because in each of these scenarios number of authentic communications is more. Hence, more time and steps are required if network size increases with increase in any entity. Next, maximum bound of both strategies is find out by increasing the entities. It shows that lightweight relationships are not acceptable for 30 subgroup controller, 60 subgroup member and 40 intruders because of unacceptable increase in number of steps to find intruders.

Table 7. Automatic subgroup controller-intruder analysis (time in msec.).

Number of Subgroup Controller	Number of Subgroup Members	Intruder Assertions	Proposed Trusted Strategy		Basic Strategy	
			Time (Steps)	Result	Time (Steps)	Result
1	1/5/10	1	23/20/29 (14/326/1751)	Proved	22/23/27 (11/310/1605)	Proved
1	1/5/10	5	12/32/41 (22/550/2595)	Proved	14/30/39 (20/400/2513)	Proved
1	1/5/10	10	11/21/54 (32/830/3650)	Proved	10/19/46 (30/810/3616)	Proved
5	1/5/10	1	92/14/35 (446/326/1751)	Proved	80/13/33 (410/310/1605)	Proved
5	1/5/10	5	37/18/34 (550/550/2595)	Proved	35/16/30 (532/400/2513)	Proved
5	1/5/10	10	37/15/58 (680/830/3650)	Proved	33/14/49 (600/810/3616)	Proved
10	1/5/10	1	74/48/23 (1751/326/1751)	Proved	70/44/21 (1704/310/1605)	Proved
10	1/5/10	5	62/18/39 (3145/550/2595)	Proved	59/16/34 (2995/400/2513)	Proved
10	1/5/10	10	13/18/84 (50/830/3650)	Proved	11/16/77 (47/810/3616)	Proved
30	60	40	>2000 (100000)	Proved	>1500 (90000)	Proved
30-38	>42	Any	Time-out	Failed	>2200 (95000)	Proved
>= 39	Any	Any	Time-out	Failed	>2500 (100000)	Proved

5.2.2 Analysis of Lightweight Primitives

Table 8. Simple vs Lightweight Primitive Analysis for Proposed Schema

Sr. No.	Primitives	Layer	Variables	Clauses	Time
1	LED	Confusion	23145	16091	1673
		Diffusion	21671	14125	1304
2	PHOTON	Confusion	43276	45214	2203
		Diffusion	38765	24712	1751
3	AES	Confusion	81180	270849	3523
		Diffusion	61467	170374	2587

As shown in **Fig. 6**, confusion layer for lightweight primitives uses simple logical operations like: AND, OR, NOT etc. to minimize the hardware cost in terms of gate equivalents (GE).

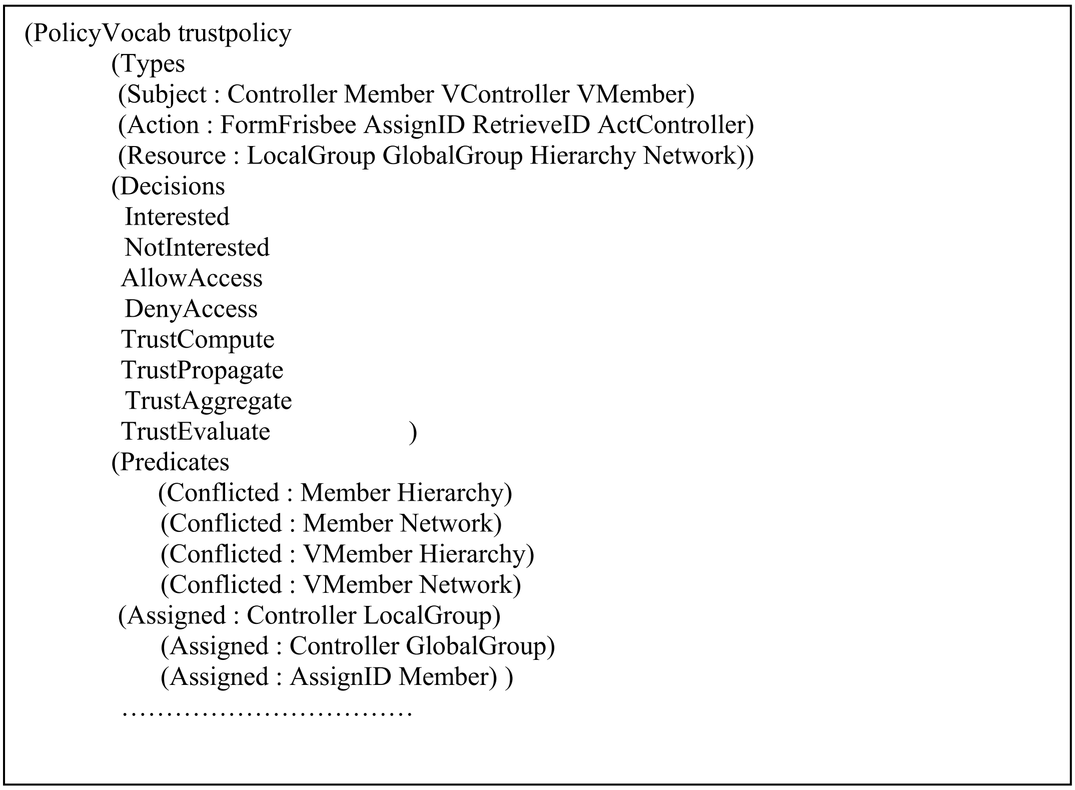
Here, A to P represents the data portion of rijandael matrix and rc1 to rc16 are the round constants. The data portion is processed through addconstant, substitute bytes and shift row phases. To achieve confidentiality and authentication using lightweight primitives, LED for encryption/decryption and PHOTON for hashing is integrated, modeled and analyzed with proposed trusted mechanism [72]. Both of these primitives are based on three operations: xoring the key, confusion and diffusion functions. **Table 8** shows the comparative analysis of substitution permutation network (SPN) based lightweight primitives (LED, PHOTON) with simple primitive (Advanced Encryption Standard (AES)) for proposed schema. In SPN networks, these three primitives use similar strategy. GE of lightweight primitive (LED and PHOTON) is less as compared to simple primitive (AES). Modeling and execution of these primitives shows that token generated in terms of variables and clauses for lightweight primitives are much lesser than simple primitive. Lightweight primitive consume less time to generate these tokens and complete operations with minimum use of GE.

```
enum Data {A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, rc1, rc2, rc3, rc4, rc5, rc6, rc7, rc8,r c9,
rc10, rc11, rc12, rc13, rc14, rc15, rc16 }

 sig AddConstant {data : set Data}
 abstract sig Model { addconstant: set AddConstant }
{#addconstant > 0}
abstract sig InputModel extends Model {}
one sig row1elements,row2elements,row3elements,row4elements extends AddConstant {}
one sig row1,row2,row3,row4 extends InputModel {}

fact addconstantsmatrix {
            row1elements.data = {xor.A.rc1} + {xor.B.rc2}+{xor.C.rc3}+{xor.D.rc4}
            row2elements.data={xor.E.rc5}+{xor.F.rc6}+{xor.G.rc7}+{xor.H.rc8}
            row3elements.data={xor.I.rc9}+{xor.J.rc10}+{xor.K.rc11}+{xor.L.rc12}
            row4elements.data={xor.M.rc13}+{xor.N.rc14}+{xor.O.rc15}+{xor.P.rc16}
            ……
}

pred substitute[from, from', to, to': set Subcell] {
     one item: from {
    (from' = from - item && to' = to + item)
     …….
            }
 }

fact shiftrows {
            row1elements.data = {xor.A.rc1} + {xor.B.rc2}+{xor.C.rc3}+{xor.D.rc4}
            row2elements.data= {xor.F.rc6}+{xor.G.rc7}+{xor.H.rc8}+{xor.E.rc5}
            row3elements.data= {xor.K.rc11}+{xor.L.rc12}+{xor.I.rc9}+{xor.J.rc10}
            row4elements.data= {xor.P.rc16}+{xor.M.rc13}+{xor.N.rc14}+{xor.O.rc15}
}
```

Fig. 6. Alloy specification for confusion layer in SPN based lightweight primitive

5.2.3 Lightweight Fine Grained Access Control Policy Analysis

Fig. 7 and **Fig. 8** show the policy and its vocabulary used for proposed schema. Proposed trust based mechanism is having: $SC_j^{HL_i}$, $SM_{(j,k)}^{HL_i}$, $VNSM_{(j,l)}^{HL_i}$ and $VNSC_j^{HL_i}$ with permission set{READ, WRITE, ACCESS, USE, MODIFY}, {READ, ACCESS, USE}, {READ, ACCESS, USE}and {READ, WRITE, ACCESS, USE, MODIFY} respectively. Fig. 8 shows that $SC_j^{HL_i}$ and $VNSC_j^{HL_i}$ are having access from bottom to top i.e from localgroups to network. Whereas $SM_{(j,k)}^{HL_i}$ and $VNSM_{(j,l)}^{HL_i}$ are having access to local groups only. These acess permission are avaiable to respective member if there is no conflict between actions and resources. In order to avoid any conflict, every member establishes relationship by processing through following phases in priority: TrustCompute, TrustPropagate, TrustAggregate, TrustEvaluate, Interested, NotInterested, DenyAccess and AllowAccess. Here, TrustCompute, TrustPropagate, TrustAggregate and TrustEvaluate are the proposed trust management phases. After passing through these phases, it has to show interest to access or deny participation. Member can compute trust and propagate its value to subgroup controller only. Subgroup controller can propagate, aggregate or evaluate trust score at global, hierachical or network level. Policy is checked through margrave language in racket toolkit. Results show that there is no conflict in any relationship among any member of proposed schema. It also confirms that subjects mentioned in vocabulary can perform necessary actions in resources and make decisions provided that it should not violates the conflicts and assigned tasks.



```
(PolicyVocab trustpolicy
            (Types
             (Subject : Controller Member VController VMember)
             (Action : FormFrisbee AssignID RetrieveID ActController)
             (Resource : LocalGroup GlobalGroup Hierarchy Network))
            (Decisions
              Interested
              NotInterested
             AllowAccess
              DenyAccess
             TrustCompute
             TrustPropagate
              TrustAggregate
             TrustEvaluate                  )
            (Predicates
                   (Conflicted : Member Hierarchy)
                    (Conflicted : Member Network)
                    (Conflicted : VMember Hierarchy)
                    (Conflicted : VMember Network)
             (Assigned : Controller LocalGroup)
                    (Assigned : Controller GlobalGroup)
                    (Assigned : AssignID Member) )
             ……………………………
```

Fig. 7. Access Control Margrave Vocabulary used in Policy for Proposed Schema

```
(Policy TrustPolicy1 uses trustpolicy
  (Target )
  (Rules
    (ControllerNoConflict = (Interested s a r) :- (!Conflicted s r) (ActController a) (LocalGroup r))
    (ControllerAssigned = (NotInterested s a r) :- (Assigned s r) (ActController a) (GlobalGroup r))
    (ControllerConflict = (DenyAccess s a r) :- (Conflicted s r) (ActController a) (Hierarchy r))
    (MemberNoConflict = (Interested s a r) :- (!Conflicted s r) (ActController a) (GlobalGroup r))
    (MemberAssigned  = (Interested s a r) :- (Assigned s r) (AssignID a) (LocalGroup r))
    (MemberConflict = (Interested s a r) :- (Conflicted s r) (RetrieveID a) (LocalGroup r))
    (MemberTrust = (TrustCompute s a r) :- (Assigned s r) (AssignID a) (LocalGroup r))
    (MemberTrustConflict =(TrustAggregate s a r) :- (Conflicted s r) (LocalGroup r))
    (MemberTrustEConflict =(TrustEvaluate s a r) :- (Conflicted s r) (LocalGroup r))
   )
(RComb O TrustCompute TrustPropagate TrustAggregate TrustEvaluate Interested NotInterested
DenyAccess AllowAccess)
(PComb FAC)
(Children)
)
```

Figure 8: Margrave policy for Access Control in Proposed Scheme.

5.3 Result Analysis

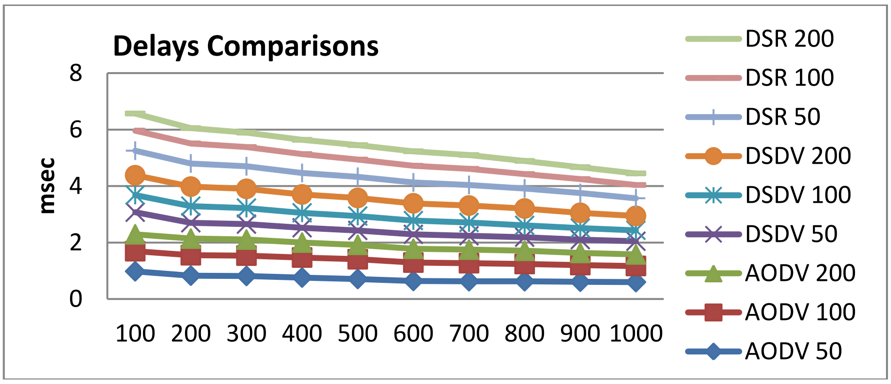


Fig. 9. End to end delay comparison of proposed scheme over MANET routing protocols with variation in number of nodes.

In simulation, initially zero trust is established among mobile nodes. Nodes use three different MANET's routing protocols to establish trusts: Ad-hoc On-demand Distance Vector (AODV), Destination Sequenced Distance Vector (DSDV) and Dynamic Source Routing (DSR). **Fig. 9** shows the effect of varying the number of nodes among these routing protocols. AODV with 50, 100 and 200 nodes give minimum delay and this delay decreases with increase in simulation time. It can also be observed that delay increases with increase in number of nodes however this growth will not be more than 20%. More passages are available for data communication due to increase in number of nodes. **Fig. 10** shows comparison of jitter, initial setup, propagation and processing delays. It can be observed that AODV posses minimum figures as compared to DSDV and DSR. This is because both proposed trust scheme and

AODV protocol are reactive in nature and build path prior to data transmission. A minimum traffic delay is developed because of establishing new routes. As shown in **Fig. 11**, throughput and power consumptions for three protocols are almost equal. But AODV provides minimum delay with same throughput and power consumption among three routing protocols thus AODV is considered to be the best protocol for proposed scheme.

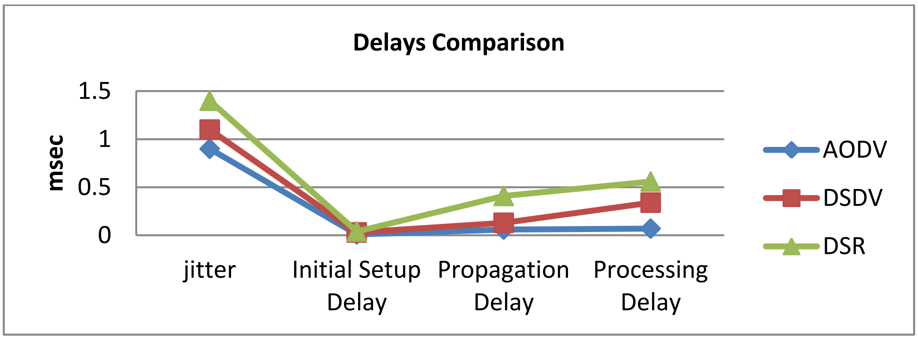


Fig. 10. Delay Comparison of proposed scheme over MANET routing protocols.

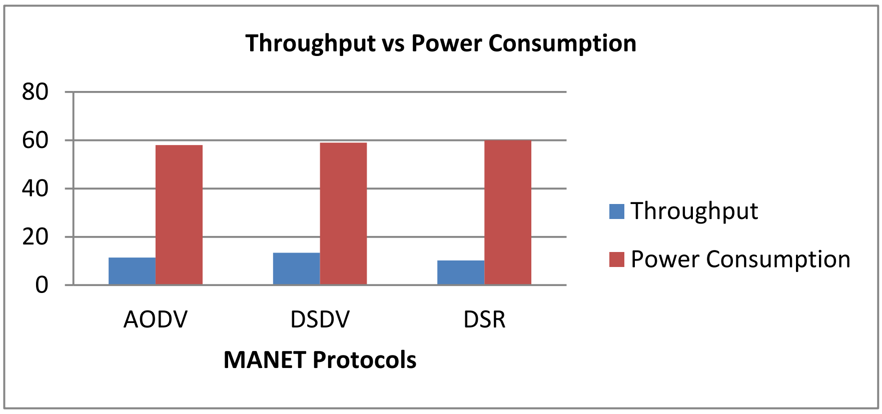


Fig. 11. Power vs Throughput comparison over MANET routing Protocols.

5. Conclusion

In this work, a new method is presented for lightweight trust computation, propagation, aggregation and prediction. The system computes trust at local subgroup level from it's members by calculating centrality score and transmit it to top hierarchies. Therefore, taking into consideration the entire system access control with single primary subgroup, Frisbee model is integrated to create such subgroups and hierarchies to avoid harmful losses for resource constraint networks. Access control policies designed for every member in network are modeled in Alloy and analyzed in Margrave. It is observed that lightweight strategy consume less time and show no right conflicts with minimum use of hardware resources. Furthermore, it is found that the proposed system is protected from various attacks with better quality of service incents subgroup members which can share access rights and self-defense of their own secure data for inauthentic data. At last, lightweight mechanism used in this work increases the complexity of system with time and number of rounds. Thus a re-initialization after regular intervals of time will boost the network services.

References

[1] A. Kumar, K. Gopal and A. Aggarwal, "Outlier Detection and Treatment for Lightweight Mobile Ad Hoc Networks", *Int. Conf. on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QSHINE 2013)*, Greater Noida, India, volume 115, pp 750-763, 2013. Article (CrossRef Link)

[2] J. McCumber, *Assessing and Managing Security Risk in IT Systems: A Structured Methodology*, 1st Edition, CRC Press, 2005.

[3] Henk C. A. van Tilborg, *Encyclopedia of Cryptography and Security*, 2nd edition, Springer-verlag, USA, 2011. Article (CrossRef Link)

[4] C. Adam and S. Farrell, "Internet X.509 public key infrastructure: Certificate management protocols." *Internet Request for Comments 2510*, 1999. Article (CrossRef Link)

[5] B. Ramsdell, "S/MIME Version 3 certificate handling", *Internet Request for Comments 2632*, 1999. Article (CrossRef Link)

[6] C. Boyd and A. Mathuria, "Key establishment protocols for secure mobile communications: A selective survey", *Elsevier Computer Communication*, vol. 23, issues 5-6, pp. 575-587, 1998. Article (CrossRef Link)

[7] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithm", *IEEE Transaction on Information Theory*, vol. 31, pp. 469-472, 1985. Article (CrossRef Link)

[8] Yair Amir, Yongdae Kim, Cristina Nita-Rotaru, and Gene Tsudik, "On the Performance of Group Key Agreement Protocols", *ACM Transactions on Information and System Security",* nol. 7, no. 3, Pages 457-488, (August 2004). Article (CrossRef Link)

[9] Paul Judge, Mostafa Ammar, "Security Issues and Solutions in Multicast Content Distribution: A Survey," *IEEE Network Magazine*, pp. 30-36, 2003. Article (CrossRef Link)

[10] H. Harney, C. Muckenhirn, "Group Key Management Protocol Architecture", *Internet Request for Comments 2094*, July 1997. Article (CrossRef Link)

[11] H. Harney, C. Muckenhirn, "Group Key Management Protocol Specification", *Internet Request for Comments 2093*, July 1997. Article (CrossRef Link)

[12] H. Harney, U. Meth, A. Colegrove, "Group Secure Association Key Management Protocol", *Internet Request for Comments* 4535, June 2006. Article (CrossRef Link)

[13] B. Weis, S. Rowles and T. Hardjono, " The Group Domain of Interpretation", *Internet Request for Comments 6407*, October 2011. Article (CrossRef Link)

[14] M. Baugher, B. Weis, T. Hardjono, H. Harney, "The Group Domain of Interpretation", *Internet Request for Comments 3547*, July 2003. Article (CrossRef Link)

[15] P. Hoffman, "Algorithm for Internet Key Exchange version 1 (IKEv1)", *Internet Request for Comments 4109*, May 2005. Article (CrossRef Link)

[16] C. Kaufman, "Internet Key Exchange (IKEv2) Protocol", *Internet Request for Comments 4306*, December 2005. http://www.ietf.org/rfc/rfc4306.txt

[17] T. H. Dunigan and C. Cao, "Group Key Management", *Technical Report ORNL/TM-13470*, 1998.

[18] M. Burmester and Y. Desmedt, "A Secure and scalable group key exchange system", *In Information Processing Letters*, 94(3), pp. 137-143, 2005. Article (CrossRef Link)

[19] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system", *In proceedings of Eurocrypt*, LNCS 950, pp. 275-286, Springer-Verlag, 1995. Article (CrossRef Link)

[20] D. Bonch and M. Franklin, "Identity-based encryption from weil pairing," *Advances in Cryptology-Crypto 2001*, LNCS 2139, pp. 213-229, Springer-Verlag, 2001. Article (CrossRef Link)

[21] J. V. D. Merwe, D. Dowoud and S. McDonald, "A Survey on Peer to Peer key management for Mobile Ad Hoc Networks", *ACM Computing Surveys*, vol. 39, No. 1, Article 1, April 2007. Article (CrossRef Link)

[22] H. Deng, A. Mukherjee, D. Aggarwal, " Threshold and identity based key management and authentication for wireless ad hoc networks," in *Proc. of the international conference on information technology: Coding and Computing (ITCC's 04),* pp. 1-9, 2004. Article (CrossRef Link)

[23] Y. Zhang, W. Liu, W. Lou and Y. Fang, " Securing mobile ad hoc networks with certificateless public keys," *IEEE Transaction on Dependable and Secure Computing*, vol. 3, pp. 386-399, 2006. Article (CrossRef Link)

[24] A. Kumar, A. Aggarwal, Charu, "Efficient Hierarchical Threshold Symmetric Group Key Management Protocol for Mobile Ad Hoc Networks," in *Proc. of International Conference on Contemporary Computing (IC3 2012), JIIT, Noida*, India, pp. 335-346, 2012. Article (CrossRef Link)

[25] A. Schaad and J. D. Moffett, "A lightweight approach to specification and analysis of role based access control extensions", Proceedings of the seventh ACM symposium on Access control models and technologies (SACMAT'02), New York, NY, USA, pp. 13-22, 2002. Article (CrossRef Link)

[26] J. W. Bryans, J. S. Fitzgerald, "Formal engineering of XACML access control policies in VDM++", *ICFEM 2007*, Butler, M. Hinchey, M. G., Larrondo-Petrie, M. M. (eds.), LNCS, Springer, Heidelberg, vol. 4789, pp. 37-56, 2007. Article (CrossRef Link)

[27] K. Fisler, S. Krishnamurthi, L. A. Meyerovich, and M. C. Tschantz, "Verification and change-impact analysis of access control policies," in P*roc. of 27th International Conference on Software Engineering*, pp. 196-205, 2005. Article (CrossRef Link)

[28] D. Jackson, *Software Abstractions: Logic, Languages, and Analysis*, MIT Press, ISBN: 978-0-262-10114-1, 2006.

[29] D. Jackson, "Micromodels of Software: Lightweight Modelling and Analysis with Alloy", *Software Design Group*, MIT Lab Manual, Feb. 2002.

[30] D. Jackson, "Alloy: a lightweight object modelling notation", *ACM Trans. Soft. Eng. Methodol.*, vol. 11, no. 2, pp. 256-290, 2002. Article (CrossRef Link)

[31] J. Crampton, "Specifying and enforcing constraints in role-based access control," in *Proc. of the 8th ACM Symposium on Access Control Models and Technologies (SACMAT 2003),* pp. 43-50, 2003. Article (CrossRef Link)

[32] R. Sandhu and P. Samarati, "Access control: Principles and practice", *IEEE Comm.*, pp. 2-10, Sept. 1994. Article (CrossRef Link)

[33] R. C. Mayer, J. H. Davis and F. D. Schoorman, "An integrative Model of Organizational Trust", *Academy of Management Executive*, vol. 20(3), pp. 709-773, 1995. Article (CrossRef Link)

[34] A. Josang, "The right type of trust for distributed systems," in *Proc. of the ACM New Security Paradigms Workshop*, pp. 119-131, 1996. Article (CrossRef Link)

[35] D. Denning, "A new paradigm for trusted systems", in *Proc. of ACM New Security Paradigm Workshop*, pp. 36-41, 1993. Article (CrossRef Link)

[36] D. H. Mcknight and N. L. Chervany, "The meaning of trust", University of Minnesota, *Technical repors*, http://misrc.umn.edu/wpaper/WorkingPapers/9604.pdf, 1996.

[37] K. Govindan, P. Mohapatra, "Trust Computations and Trust Dynamics in Mobile Adhoc Networks: A Survey", *IEEE Communications Surveys and Tutorials*, vol. 14(2), pp. 279-298, 2012. Article (CrossRef Link)

[38] A. A. Pirzada and C. McDonald, "Establishing Trust in Pure Ad-hoc Networks" , *Australasian Computer Science Conference*, The university of Otago, Dunedin, New Zealand, 2004. Article (CrossRef Link)

[39] M. J. Probst and S. K. Kasera, "Statistical trust establishment in wireless sensor networks", in *Proceedings of the 13th International Conference on Parallel and Distributed Systems*, pp. 1-8, 2007. Article (CrossRef Link)

[40] Z. Liu, A. W. Joy and R. A. Thompson, " A dynamic trust model for mobile ad hoc networks", in *Proc. of IEEE International Workshop on Future T rends of Distributed Computing Systems, FTDCS'04*, pp. 80-85, May 2004. Article (CrossRef Link)

[41] L. Xiong and L. Liu, "PeerTrust: Supporting reputation-based trust in peer-to-peer communities", *IEEE Transaction on Knowledgement and Data Engineering, Special Issue on Peer-to-Peer Based Data Management*, vol. 16, no. 7, pp. 843-857, July 2004. Article (CrossRef Link)

[42] P. B. Velloso, R. P. Laufer, D. O. Cunha, O. C. M. B. Duarte and G. Punjollel, "Trust management in mobile ad hoc networks using a scalable maturity-based model", *IEEE Trans. Netw. Service Manag*, vol. 7, No. 3, pp. 172-185, Sep. 2010. Article (CrossRef Link)

[43] Y. L. Sun, Z. Han, W. Yu and K. J. Ray Liu, "A trust evaluation framework in distributed networks: Vulnerability analysis and denfense against attacks", in *Proc. of IEEE International Conference on Computer Communications, INFOCOM'06*, pp. 1-13, April 2006. Article (CrossRef Link)

[44] Y. Sun, W. Yu, Z. Han and K. J. Ray Liu, "Information theoretic framework of trust modeling and evaluation for ad hoc networks", *IEEE Journal on Selected Areas of Communication*, Vol. 24, No. 2, pp. 305-317, Feb. 2006. Article (CrossRef Link)

[45] N. Cheng, K. Govindan and P. Mohapatra, "Rendezvous based trust propagation to enhance distributed network security", in *Proc. of INFOCOM-2011 Workshop SCNC*, 2011., pp. 1066-1070, April 2011. Article (CrossRef Link)

[46] S. Trifunovic, F. Legendre and C. Anastasiades, "Social trust in opportunistic networks", in *Proc. of INFOCOM IEEE Conference on Computer Communications Workshops*, pp. 1-6, 2010. Article (CrossRef Link)

[47] D. Quercia, S. Hailes and L. Capra, "Lightweight distributed trust propagation", in *Proc. of The Seventh IEEE International Conference on Data Mining*, pp. 282-291, 2007. Article (CrossRef Link)

[48] Ms", Autonomous Agents and . Pardo, "Aggregation of trust for iterated belief revision in probabilistic logics", *Scalable Uncertainity Management, Lecture notes in computer science, Springer-Verlag*, pp. 165-179, 2009. Article (CrossRef Link)

[49] Y. Bachrach, A. Parnes, A.D. Procaccia and J. S. Rosenschein, "Gossip-based aggregation of trust in decentralized reputation systems", *Autonomous Agents and Multi-Agent Systems*, vol. 19, No. 2, pp. 153-172, 2009. Article (CrossRef Link)

[50] J. Huang and D. Nicol, "A calculus of trust and its application to PKI and identity management", in *Proc. of The 8th ACM Symposium on Identity and Trust on the Internet, IDtrust'09*, pp. 23-37, 2009. Article (CrossRef Link)

[51] X. Wang, L. Liu and J. Su, "Rlm: A general model for trust representation and aggregation", *IEEE Transaction on Services Computing*, Vol. 5. No. 1, pp. 131-143, 2012. Article (CrossRef Link)

[52] C. M. Jonker and J. Treur, "Formal analysis of models for the dynamics of trust based on experiences", *in MAAMAW'99: Proceedings of the 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pp. 221-232, 1999. Article (CrossRef Link)

[53] F. M. Ham, E. Y. Imana, A. Ondi, R. Ford, W. Allen and M. Reedy, "Reputation prediction in mobile adhoc networks using RBF neural networks", *Engineering Applications of Neural Networks Communications in Computer and Information Science, EANN*, CCIS 43, pp. 485-494, 2009. Article (CrossRef Link)

[54] V. Chandola, A. Banerjee and V. Kumar, "Outlier Detection: A Survey", *ACM Computing Surveys*, pp. 1-72, 2009. Article (CrossRef Link)

[55] Y. Zhang, N. Meratnia and P. Havinga, "Outlier Detection Techniques for Wireless Sensor Networks: A Survey", *IEEE Communication Surveys & Tutorials*, Vol. 12, No. 2, pp. 159- 170, 2010. Article (CrossRef Link)

[56] P. Gogoi, B. Borah and D. K. Bhattacharyya, “Anomaly Detection Analysis of Intrusion Data using Supervised and Unsupervised Approach”, *Journal of Convergence Information Technology*, Vol. 5, No. 1, Feb. 2010.

[57] P. Gogoi, D. K. Bhattacharyya, B. Borah, J. K. Kalita, “ A Survey of Outlier Detection Methods in Network Anomaly Identification”, *The Computer Journal*, vol. 54, issue 4, pp. 570-588, April 2011. Article (CrossRef Link)

[58] D. M. Hawkin, “Identification of Outliers”, Chapman and Hall, London, 1980. Article (CrossRef Link)

[59] V. A. Traag, A. Browet, F. Calabrese and F. Morlot, “Social Event Detection in Massive Mobile Phone Data Using Probabilistic Location Interference”, *SocialCom/PASSAT*, pp. 625-628, October 9-11, 2011.

[60] A. Beimel, “Secure Scheme for secret Sharing and Key Distribution”, Ph. D. thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[61] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton and J. Zhao, “Habitat Monitoring Application Driver for Wireless Communication Technology”, *In Proceedings of the ACM SIGCOMM Workshop on Data Communication in Latin America and the Caribean*, San Jose, Costa Rica, volume 31, issue 2, pp. 20-41, 2001. Article (CrossRef Link)

[62] M. Burmester and Y. Desmedt, “A secure and efficient conference key distribution system”, *Advances in Cryptology-Eurocrypt’94*, pp. 275-286, 1994. Article (CrossRef Link)

[63] J. Huang and D. Nicol, “A calculus of trust and its application to PKI and identity management”, in *The 8th ACM Symposium on identity and Trust on the Internet*, *IDtrust’09*, pp. 23-37, 2009. Article (CrossRef Link)

[64] NS3 Simulator, http://www.nsnam.org

[65] Y. L. Sun, Z. Han, W. Yu and K. J. R. Liu, “A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks”, in *Proc. of 25th IEEE International Conference on Computer Communications (INFOCOM 2006)*, pp. 1-13, April 2006. Article (CrossRef Link)

[66] C. Dellarocas, “Mechanism for coping with unfair rating and discriminatory behavior in online reputation reporting systems”, *In proceedings of ICIS*,pp. 520-525, *2000. Article (CrossRef Link)*

[67] V. V. Vigilkumar, V. M. A. Rajam, “Detection of Colluding Selective Forwarding Nodes in Wireless Mesh Networks Based on Channel Aware Detection Algorithm”, *MES Journal of Technology and Management*, pp. 62-66, Vol II, Issue 1, ISSN: 0976-3724, 2011. Article (CrossRef Link)

[68] Y. L. Sun, Y. Liu, “Security of Online Reputation Systems: The evolution of attacks and defenses*”, IEEE Signal Process Mag*. Vol 29(2), pp. 87-97, 2012. Article (CrossRef Link)

[69] Y. Sun, H. Luo, S. K. Das, “A Trust Based Framework for fault tolerant data aggregation in wireless multimedia sensor networks”, *IEEE Trans. Dependable Sec. Comput.*, vol. 9(6), pp. 785-797, 2012. Article (CrossRef Link)

[70] S. D. Kamvar, M. T. Schlosser and H. Garcia Molina, “The eigentrust algorithm for reputation management in p2p networks”, *in Proceedings of the 12th international conference on world wide* web, pp. 640-651, 2003. Article (CrossRef Link)

[71] P. England, Q. Shi, B. Askwith and F. Bouhafs, *A Survey of Trust Management in Mobile Ad Hoc Networks*, ISBN: 978-1-902560-26-7, 2012.

[72] M. R. S. Abyaneh, “Security Analysis of Lightweight Schemes for RFID Systems”, Ph. D. THESIS, University of Bergen, Norway, (June 2012).

Adarsh Kumar is currently working as Assistant Professor in Computer Science Engineering and Information Technology department at Jaypee Institute of Information Technology, Noida, INDIA, since September 2005. Mr. Kumar received his B.Tech (Computer Science) and M.Tech (Software Engineering) from Punjab Technical University and Thapar University, Patiala in June 2003 and July 2005 respectively. He is pursuing PhD in Computer Science from Jaypee Insttute of Information Technology, Noida, INDIA.

Prof. Krishna Gopal is currently working as Dean (Academic and Research) at Jaypee Institute of Information Technology, Noida, INDIA since 2011. Prof. Gopal is having 45 years of teaching and R&D experience. . He received his Bachelor, Master and PhD in Electronics engineering from IIT, Madras, REC Kurukshetra in 1966, 1972, 1979 respectively. He published more than 100 papers in different journals, conferences, patents etc. He handled six sponsored projects in his career. He has done various administrative responsibilities like: Director, Dean in REC Kurukshetra. He is member of various professional bodies like: Life Member System Society of India, Indian Society for Technical Education, Past Member Institution of Engineers (India), Indian Association for Quality and Reliability, Senior member of IEEE etc.

Dr. Alok Aggarwal is currently working as Professor & Director at JPIET, Meerut, INDIA, since 2012.He is having work experience of sixteen years with a mix of software developer, research and teaching. He received his Bachelor, Master and PhD in Computer Science and Engineering from Kurukshetra University and IIT, Roorkee in 1995, 2001, 2010 respectively. He published four books and more than hundred research papers in different journals, conference proceedings etc.