Protection of MANETs from a range of attacks using an intrusion detection & prevention system

Adnan Nadeem and Michael Howarth

Abstract Mobile ad hoc networks (MANETs) are well known to be vulnerable to various attacks due to their lack of centralized control, and their dynamic topology and energy-constrained operation. Much research in securing MANETs has focused on proposals which detect and prevent a specific kind of attack such as sleep deprivation, black hole, grey hole, rushing or sybil attacks. In this paper we propose a generalized intrusion detection and prevention mechanism. We use a combination of anomaly-based and knowledge-based intrusion detection to secure MANETs from a wide variety of attacks. This approach also has the capability to detect new unforeseen attacks. Simulation results of a case study shows that our proposed mechanism can successfully detect attacks, including multiple simultaneous different attacks, and identify and isolate the intruders causing a variety of attacks, with an affordable network overhead. We also investigate the impact on the MANET performance of (a) the various attacks and (b) the type of intrusion response, and we demonstrate the need for an adaptive intrusion response.

Keywords *MANETs. Ad hoc network security. Intrusion detection & prevention. Secure routing.*

Adnan Nadeem and Michael Howarth

Centre for Communication Systems Research, University of Surrey, Guildford GU27XH, Surrey, United Kingdom
Adnan Nadeem

e-mail: a.nadeem@surrey.ac.uk
Michael Howarth
e-mail: m.howarth@surrey.ac.uk

This is an extended version of our paper previously published at IEEE ICUMT, 2009.

1. Introduction

MANETs are an infrastructure-less network of autonomous devices, where these devices also act as intermediate routers. MANET routing protocols can be classified as either proactive or reactive.. Reactive routing protocols such as AODV [1] and DSR [2] are now considered more effective and scalable compared to their proactive counterparts such as OLSR [3], because they have less routing overhead. AODV and DSR are designed under the assumption that all nodes trust each other and there are no malicious intruder nodes in the network. Therefore, the presence of any such node imposes security challenges. Malicious nodes can cause severe disruption through a wide variety of attacks including both routing and data forwarding attacks. Attacks are generally classified as either passive or active attacks. In passive attacks, the attacker does not disturb the operation of the network but attempts to discover valuable information. On the other hand active attacks cause various degrees of damage to the network depending on the type of attack; we investigate several such attacks in this paper.

Intrusion detection and prevention (IDP) [4] provides a way to protect nodes against active routing attacks. There are two intrusion detection (ID) techniques, known as knowledge-based intrusion detection (KBID) and anomaly-based intrusion detection (ABID). KBID maintains a knowledge base containing signatures or patterns of known attacks and looks for these patterns in an attempt to detect them; for example a rule based expert system to detect intrusion is proposed in [5]. KBID has a potentially low false detection rate but it can only detect attacks whose signatures are in the database, and it is difficult to gather signatures and keep them up to date. On the other hand, ABID can flag observed activities that deviate significantly from the established normal profile. ABID consist of two phases: training and testing. This technique not only provides early warnings for potential intrusions but also can detect attempts to exploit new and unforeseen vulnerabilities; however, it is more prone to generate false positives than KBID.

In our initial work [6] we proposed Adaptive Intrusion Detection and Prevention (AIDP), which used ABID to detect denial of service (DoS) attacks. In this extended version of our paper [25] we propose Generalised Intrusion Detection & Prevention (GIDP) mechanism. It uses a combination of anomaly-based and knowledge-based ID that takes advantage of both techniques to guard MANETs against a wide variety of attacks. It has the capability to detect new intrusive activities that degrade network performance. We further analyze various attacks and their impact on network performance and compare this with the impact on network performance of GIDP's intrusion response, which is to isolate the intruder from the network. Finally, we demonstrate the need of an adaptive intrusion response for IDP in MANETs.

The reminder of this paper is organized as follows. Section 2 describes the related work and challenges in intrusion detection and securing MANETs. Section 3 reviews typical MANET routing attacks. Section 4 presents our proposed mechanism, GIDP. Section 5 illustrates the implementation of GIDP through a case study, including simulation. Section 6 presents an investigation of the impact of various attacks and intrusion responses on network performance, again including simulation results. Finally, we summarize our results and future work in Section 7.

2. Related Work

2.1 Intrusion Detection

ID in MANETs is more challenging than in fixed networks because the former lack a concentration point where traffic can be analyzed, and because of their dynamic topology and limited computational ability of nodes. In spite of these challenges some research in the literature has focus on ID in MANETs. For example Zhang and Lee[7] argue that many ID techniques developed for fixed wired networks are not applicable in MANETs, and they propose an ID and response mechanism in which an Intrusion Detection System (IDS) agent performs local data collection and local detection. They then trigger a cooperative detection and global response when a node reports an intrusion. In [8] Hijazi and Nasser studied and analysed the feasibility of mobile agents in MANETs and they concluded that many mobile agents' features are the exact requirement for MANET IDS. In [9] Cretu et al. proposed an anomaly detection approach for MANETs in which it models device behavior that peers can use to determine trustworthiness of other nodes. Jiang and Wang [10] proposed an anomaly detection algorithm based on Markov chains for wireless ad hoc networks. This algorithm consists of two parts: the first constructs a Markov chain table with state transitions. Then second part is a classifier which checks whether the current transition is in the Markov chain by calculating trace values and setting the threshold to detect anomalies.

2.2 Securing MANETs

A significant research effort has already been made to secure MANETs, but most of the work has focused on detecting and preventing specific attacks. For example TOGBAD was proposed in [11] to identify nodes that attempt to create black hole attacks in MANETs. It detects the attack using a topology graph, looking at the number of neighbours a node claims to have and the actual number of neighbours according to the graph. It was developed for the OLSR proactive routing protocol where the topology information can be obtained, but would not be effective for reactive routing protocols, where acquiring topology information is not operationally feasible. Kurosawa and Jamalipour [12] also propose a black hole detection mechanism, this time for AODV, where three feature vectors are used to model normal states of the network and then a discrimination module is used for identifying the abnormal state that represents the black hole attack. Xiaopeng and Wei [13] propose a grey hole attack detection scheme for the DSR routing protocol. This requires each node to produce evidence on forwarding packets using an aggregated signature algorithm, and then a checkup algorithm detects whether packets are dropped or not; finally a source node uses a diagnostic algorithm to trace the malicious node. Another mechanism for grey hole detection for AODV is proposed in [14]. Ping and Zhang [15] considered a route request (RREQ) flooding attack in MANETs. They proposed a RREQ flood prevention mechanism based on neighbour's supervision. In another example Yu and Ray [16] defined two types of injecting traffic attack in MANETs as query and data packet flooding. They detect the attack if requests are made a certain number of times in t sec. These methods are based on static thresholds to detect malicious RREQ flooding, but in our opinion this does not cope well with the dynamic environment of MANETs. In [17] Perrig and Johnson analyzed how an attacker can launch a rushing attack (RU) in DSR and proposed a rushing attack prevention mechanism for MANETs.

Though most researchers have concentrated on protecting MANETs against specific types of attack, some have suggested a more general approach. For example ARAN [18] is a hop-to-hop authenticated routing mechanism that can protect MANETs against a number of attacks from external malicious nodes. A similar approach, Ariadne [19], has been proposed for end-to-end authentication based on shared key pairs. In [20] CRADS, a cross layer approach, is proposed that uses a support vector machine (SVM) to detect routing

attacks based on the proactive routing protocol OLSR. SEAD was proposed in [21] as a secure routing protocol that uses a one-way hash function to provide authentication for the proactive routing protocol DSDV.

We believe more effort is needed on mechanisms which can guard MANETs against a wide variety of attacks, and especially for reactive routing protocols since these are more widely used.

3 Routing Attacks

The on-demand MANET routing protocols, such as AODV and DSR, allow intruders to launch a wider variety of attacks. In order to illustrate these routing attacks we consider AODV as an example in this paper. Using AODV we now give examples of how different intrusive activities can cause various attacks in MANETs.

a) Sleep Deprivation through malicious RREQ flooding:

Sleep deprivation (SD) [22] is a denial of service attack in which an attacker interacts with the node in a manner that appears to be legitimate, but where the purpose of interaction is to keep the victim node out of its power-conserving sleep mode. An intruder can cause SD of a node by exploiting the vulnerability of the route discovery process of the protocol through malicious route request (RREQ) flooding in the following ways:

Malicious RREQ Flooding 1: an intruder broadcasts a RREQ with a destination IP address that is within the network address range but which does not exist. This will compel all nodes to forward this RREQ because no-one will have the route for this destination IP address.

Malicious RREQ Flooding 2: after broadcasting a RREQ an intruder does not wait for the *ring traversal time* and continues resending the RREQ for the same destination with higher TTL values.

b) Black & Grey Hole attack by false RREP and packet dropping:

In AODV, the destination sequence number (*dest_seq*) is used to describe the freshness of the route. A higher value of *dest_seq* means a fresher route. On receiving a RREQ an intruder can advertise itself as having the fresher route by sending a Route Reply (RREP) packet with a new *dest_seq* number larger than the current *dest_seq* number. In this way the intruder becomes part of the route to that destination. The intruder can then choose to drop all packets, causing a black hole (BH) [12] in the network. The severity of the attack depends on the number of routes in the network the intruder successfully becomes part of; we analyze this further in Section 5.

A Grey Hole attack (GH) [14] is a special case of the BH attack, in which intruder only drops packets selectively, e.g. from specific nodes.

c) Rushing attack through a forged RREQ:

In order to limit the routing protocol overhead an on-demand protocol only requires nodes to forward the first RREQ that arrives for each route discovery. An attacker can exploit this property by spreading RREQ packets quickly throughout the network so as to suppress any later legitimate RREQ packets. An intruder can forward the forged rushed RREQ, giving them a higher source sequence (*src_seq*) number and minimum delay. This will suppress the later legitimate RREQ and increase the probability that routes that include the intruder will be discovered rather than other valid routes, causing a rushing attack.

d) Sybil attack through forged control packet

Each node in a MANET requires a unique address to participate in routing, and nodes are identified through this address in the network. There is no central authority to verify these identities in MANETs. An attacker can exploit this property and send control packet, for example RREQ or RREP, using different identities; this is known as a sybil attack [23].

4 Our Proposed Mechanism

4.1 Assumptions

We disregard attacks aimed at the physical and data link layers. We have not considered attacks from colluding intruders in this paper. To illustrate the implementation of GIDP we assume a clustered MANET organization. We select the most capable nodes in terms of their processing abilities as cluster heads (CHs) and the others nodes become cluster nodes (CNs). At present we assume secure communication between CH and CNs. We use ABID to detect intrusion in the network; this requires traffic traces that contain only normal activities to build a training profile. However, in contrast with fixed networks, data resources such as [24] that reflect normal activities or events are not currently available for MANETs. Therefore we assume that the initial behaviour during the settling period of the network formed on-the-fly is free from anomalies.

4.2 GIDP Architecture

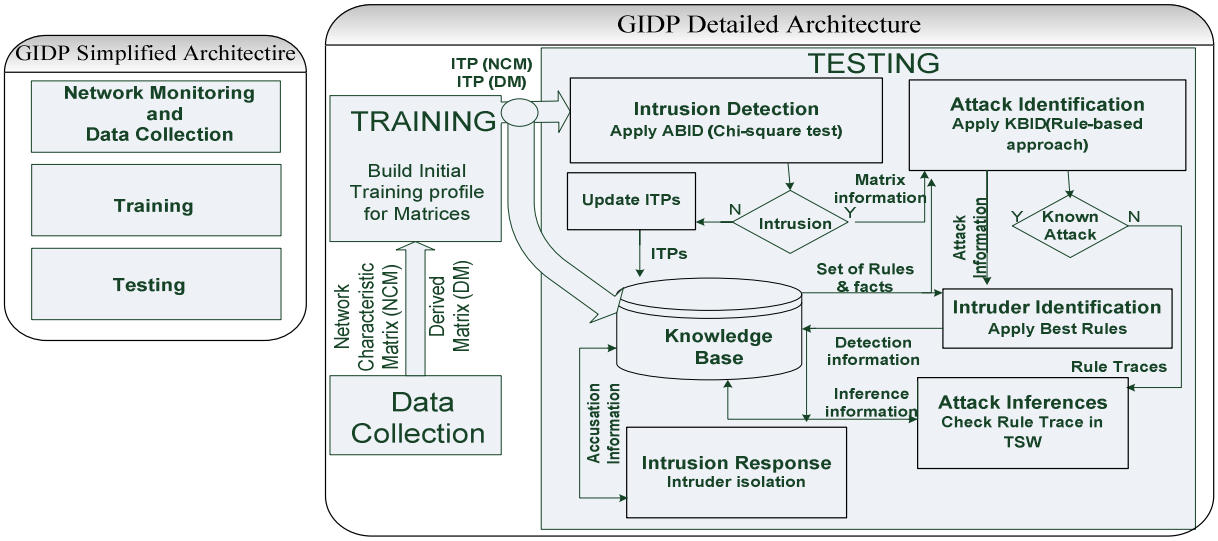


Fig.1 GIDP architecture: (a) simplified, left; (b) detailed, right.

We now describe our proposed mechanism GIDP. This is a hybrid IDP approach that uses a combination of anomaly-based and knowledge-based ID. The diagram (a) on the left of Fig.1 shows the simplified architecture. GIDP monitors the network and collects audit data specific for intrusion detection throughout the network's lifespan. Once the network is established, training is performed for *N* time intervals (TI) to obtain an initial training profile (ITP). The testing module is then called after the training module has run, and this continuously tests the network for intrusion detection and prevention after each further TI.

The detailed architecture of GIDP is represented by diagram (b) on the right of Fig.1. During data collection a cluster head gathers data in the form of two matrices: the network characteristic matrix (NCM) and a derived matrix (DM). The NCM contains data specific to the network routing protocol; for example in the case study in this paper, the NCM consists of seven parameters:

NCM= {RREQ (route request), RREP (route reply), RERR (route error), TTL (time to live) values, RREQ src_seq, RREQ dest_seq, RREP dest_seq}

The DM consists of parameters which reflects the network performance and can be derived from NCM parameters. Network throughput is also included as a parameter in this matrix. In the case study in this paper DM consists of four parameters:

DM= {RPO (routing protocol overhead), PDR (data packet delivery ratio), CPD (number of control packet dropped), Throughput}

The cluster head (CH) employs two phases: training and testing. Fig.2 shows the time-based operation of GIDP. When the network is established, the CH continuously gathers NCM and DM information and applies the GIDP training module for *N* time intervals (TI), resulting in initial training profiles (ITPs) of the NCM and DM. The ITPs reflects the normal behaviour of the nodes in the network and the expected network performance. In the testing phase the CH applies the testing module after each TI. The testing phase consists of several tasks as shown in Fig.1(b). Firstly it detects intrusion in the network. If there is no intrusion in the network then it updates the ITPs in order to adapt the variation in the network behaviour as time progresses. If there is intrusion, in the second task the CH identifies the attack or attacks using existing information in the knowledge base. In the case of known attacks the CH identifies intruding nodes using existing intruder identification rules specific to the known attack in the knowledge base. To optimise the probability of identifying intruders correctly with a low level of false positives, it maintains a test sliding window (TSW) as shown in Fig.2, in which *d* detections of a node are required in *p* time intervals (TI). If this detection threshold is passed then the CH will blacklist the node and isolate the node by informing all CNs.

If attack identification detects an attack that does not match the rules for known attacks then the CH applies the attack inferences. Attack inference stores the rule trace of the current TI as Detected Rule Trace and looks for its match in a TSW. If the match is found in a TSW then the CH confirms the new attack by constructing & adding a rule for the new attack in the set of rules stored in knowledge base.

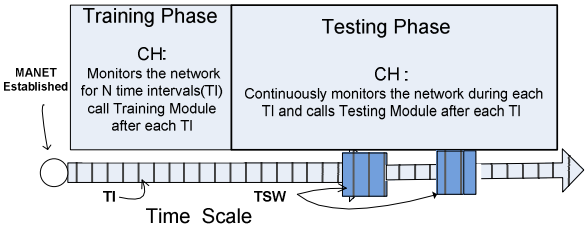


Fig.2 Time-based operation of GIDP.

4.3 Algorithm & Technical Details

We now explain the GIDP training & testing modules.

4.3.1 Training

The NCM consists of X_i parameters mentioned above, where i=1 to 7 and each X_i ={X_1, X2, X3,…,X_M} is a set of random variables from 1 to M, where M is the maximum number of random variables of parameter X_i. For example *NCM [X_i]* represent the number of RREQ received by all CNs in the *j*th time interval (TI), where M is the maximum number of RREQ received in a *jth* TI. The probability distribution of *NCM[X_i]* is calculated for the TI. The CH then calculates the DM parameters RPO (i.e. the ratio of the number of control packet to the number of data packets delivered), PDR (i.e. the ratio of the number of data packets received to data packet originated), CPD (i.e. the number of control packets dropped in establishing & maintaining routes in the network) & throughput for the *j*th TI. This whole process is then repeated for the *N* time intervals in the training phase. We then calculate the mean $\overline{X}_i$ of *P(NCM[X_i])* and the means of RPO, PDR and CPD for *N* intervals, and these are stored as an ITP (NCM) and ITP (DM) respectively, containing the expected values for that particular network observed for the total time of $N*TI$ seconds.

4.3.2 Testing

In the testing phase GIDP operates in three stages: a) intrusion detection, b) attack identification and inferences and c) identification and isolation of intruding nodes (Fig.1). We now explain the algorithms of stages a, b & c. For stage a) it employs ABID using chi-square goodness of fit test on NCM. In stages b) and c), KBID is applied on both matrices NCM & DM using a rule-based approach.

Testing Modules

This module only takes NCM parameters into account and applies the chi-square test to identify any intrusion in the network.

a) Intrusion Detection

. Do after each TI

. Collect NCM(X i) from all other CNs in TI, for ∀i

. Calculate the probability distribution P (NCM(X i))

.Calculate averages of P(NCM(X_i)) & store as observed values

. End do

.For ∀i perform hypothesis testing by first calculating

chi- computed (χ2[i] using eq.1) for Xi

H_o[i]: Observed distribution of NCM (X_i) fits the expected

H_a[i]: Observed distribution of NCM (X_i) does not fit expected

.If (chi-computed[i] (α.d.f[i]) > P-value[i] (α.d.f[i]))

Reject H_o[i]. endif.

.End for

.Combined Null Hypothesis Testing

Combine H_o: Observed distribution of NCM fits the expected

Combine H_a: Observed distribution of NCM does not fit expected

.If (combined H_o is rejected)

Perform Attack identification & inferences Fig.4

else: Update Expected values NCM($\overline{X}_i$) (i.e. ITP(NCM))

.Exit

Fig.3 Pseudocode of intrusion detection module.

$$\chi^2[i] = \forall_i \left(\sum_{k=1}^{M} \frac{(NCM(X_{ik}) - NCM(\overline{X_{ik}}))^2}{NCM(\overline{X_{ik}})}\right) \ldots\ldots\ldots (1)$$

This module continuously monitors the network. In each TI the CH first performs hypothesis testing for each parameter X_i of NCM at calculated chi-computed values obtain from eq.1, where X_i is the parameter of NCM and k(1 to M) is the number of random variable in each parameter X_i. The CH then performs combined hypothesis testing of NCM as shown in Fig.3. If the combined H_o is rejected then it assumes intrusion in the TI. Else we update the ITP (NCM) using an exponentially weighted moving average (EWMA):

$$\forall_i \left(NCM(\overline{X_i^{(q,k_i^M)}}) = \alpha * NCM(X_i^{(q,k_i^M)}) + (1-\alpha) * NCM(\overline{X_i^{(q,k_i^M)}})\right) .. (2)$$

where $NCM(\overline{X_i^{(q,k_i^M)}})$ and $NCM(X_i^{(q,k_i^M)})$ represent the expected and observed values for update period number (q) respectively. The value of q is incremented in the TI when no intrusion in the MANET is detected. k represents the random variable from 1 to M in each X_i and α=2/(q-1) is the weighting factor. As q increases the weighting for older data points decreases exponentially giving more importance to the current observation.

b) Attack identification and inferences

*.Read **set of rules** Fig5*

.Set up the Interpreter for rule-based approach

*.Interpreter applies forward-chaining on **set of rules** Fig.5*

.If (Any Goal Condition of known attacks are fulfilled)

*Apply rules for **Intruder Identification & Isolation** Fig.7*

.endif.

.If (Goal Condition==" POTENTIALUNKNOWNATTACK")

*Interpreter applies **Attack Inferences** Fig.6*

.endif.

. Exit.

Fig.4 Pseudocode of attack identification & inference module.

Set of Rules example

Rule.1 $\exists x$ *(chi-squaretest(NCM[x]))-> (CheckDerivedMatrix=TRUE)*

Rule.2 CheckDerivedMatrix Λ $\exists y$ *(Test(DM[y]))-> (PotentialAttack=TRUE)*

Rule.3 PotentialAttack ->(BestRule=TRUE)

Best Rules for some known attacks:

Rule.4 BestRules Λ *(chi-squaretest(NCM[RREQ]))* Λ *Test(DM[RPO]) -> "SLEEP DEPRIVATION"*

Rule.5 BestRules Λ *(chi-squaretest(NCM[RREPdest_seq]))* Λ *(Test(DM[PDR]) V Lowest(PDR)) -> "BLACKHOLE"*

Rule.6 BestRules Λ *(chi-squaretest(NCM[RREPdest_seq]))* Λ *(Test(DM[PDR]) -> "GREYHOLE"*

Rule.7 BestRules Λ *(chi-squaretest(NCM[RREQsrc_seq]))* Λ *(Test(DM[CPD]) -> "RUSHING"*

Rule.8 $\neg(\forall x$ *(chi-square-test(NCM[x])))* Λ $\neg(\forall y($ *Test(DM[y])))* --> *"POTENTIALFALSEALARM"*

Rule.9 (Rule.1 Λ *Rule.2* Λ $\neg$*BestRule) -> " POTENTIALUNKOWNATTACK"*

Fig.5 Set of Rule examples in knowledge base.

Attack Inferences

. If (Detected Rule Trace is empty)

Store Detected Rule Trace = Rule Trace

Else If (Rule Trace == Detected Rule Trace)

New attack Rule Trace= Rule Trace

Construct a rule for New attack Rule Trace

Append New attack Rule Trace in set of rule trace

Set Detected Rule Trace =Empty . endif

.endif

Fig.6 Pseudocode of Attack inferences.

In case of intrusion the CH calls the Attack Identification and Inferences module (Fig.4). This module obtains a set of rules from knowledge base, an example set being presented in Fig.5. We have constructed these rules from our previous work [6] (our AIDP simulation results), analyzing various attacks & their impact on network performance through simulations and analysis of existing literature of known attacks, for example [12, 14, 15 & 17]. In Fig.5 *chi-square test(NCM[x])* predicate returns true if the parameter *x* is anomalous in NCM. Similarly the predicate or propositional function *Test (DM[y])* returns true if the test on parameter y of DM fails. This test uses a tool of Statistical Process Control known as variable control chart based on standard deviation σ. In the Attack Identification & Inference module a rule based approach is used in which an interpreter can either employ forward or backward chaining system. A forward chaining system process the rules one by one by checking premises (condition in the rule) to reach conclusions; it can also draw new conclusions. On the other hand backward chaining is goal driven, that is it reaches the conclusion first and keeps looking for rules that would allow the conclusion. In GIDP an interpreter applies forward chaining on the set of rules, Fig.5, at the end looking for the Goal Condition fulfilled as described in fig.4.

c) Intruder Identification &Isolation

a) Identifying intruding nodes

. Obtain known attack Rules for intruder Identification

. for all Goal conditions fulfilled:

Apply intruder identification rule for each detected known attack

add each detected node V_i *to List of Nodes Detected (LND)*

. endfor

b) Response Mechanism

For all nodes V_i *in LND*

. If (V_i *detections in Potential Intruder List(PIL) > Detections_required_To_ Accuse (d))*

CH: Blacklist V_i *& Broadcast Accusation Packet (AP)*

else : enter V_i *in PIL .endif*

.End for

c) Accusation Packet (AP) Handling

. Each CN V_i *maintain its local BlacklistTable (BLT)*

.if CN V_i *receives an AP for CN* V_j

.If CN V_i *has node* V_j *in its BLT then Ignore AP*

else: CN adds node V_j *to its BLT & rebroadcast AP*

.endif

.endif

d) Isolating Intruding Nodes

.if node V_i *receives packet from node* V_j

.If node V_j *is in node* V_i *BLT*

Ignore packet & drop all packets queued from V_j

Else: handle & process packet .endif

.endif

Fig.7 Pseudocode of intruder identification & isolation module.

In case of any known attack detected in the TI, the interpreter applies the Intruder Identification & Isolation module (fig.7) to identify and isolate the intruding nodes. This module first identifies the intruding nodes by applying the known attack rules for intruder identification. For example in case of a SD attack (Fig.5 Rule 4) it employs control chart (explained above) based on σ of RREQ generated by all nodes and adds detected node V_i to the LND. The Response Mechanism (Fig.7(b)) then checks if detection threshold d is reached for any node V_i in the list of nodes detected (LND) in the last p TIs. If so, then it blacklists the node V_i and informs all other CNs by sending an Accusation Packet (AP). When a CN receives an AP it first checks the broadcast id & source address to avoid processing a duplicate AP. If the accused node is already blacklisted the CN will ignore & drop the AP to prevent unnecessary network traffic. Otherwise, the CN will blacklist the accused node and rebroadcast the AP. Finally, to isolate the intruder from the network all nodes will not only drop the packets from a blacklisted node but also immediately ignore all packets in their queue that are from the blacklisted nodes, as shown in Fig.7(d).

If Goal Condition with *POTENTIALUNKNOWNATTACK* is fulfilled during the attack identification process then the interpreter saves this *Rule Trace* and looks for the match of this *Rule Trace* in the current TSW. If a match is found then it confirms the new attack detection by constructing a new rule and appending the new rule in a Set of Rules stored in the knowledge base (Fig.6).

5 Case Study

In this section, we consider a case study with different attack scenarios & analysis of GIDP overhead, to assess the applicability and performance of GIDP. We present the simulation results of these scenarios and some key findings from the analysis of attacks. We used GloMoSim [26] to build the simulation environment, using the simulation parameters shown in Table 1.

Table 1 Simulation Parameters

Number of Nodes	25 & 50
Terrain Dimension	500 * 500 metres & 707 * 707 metres
Node placement	Uniform distribution
Simulation Traffic	CBR (Constant Bit Rate)
Simulation time	2500 seconds
Routing protocol	AODV
MAC protocol	IEEE 802.11
Mobility	Random Way Point Model (RWP)
Nodes mean speed	Varies from 0 to 20 m/s

In this case study GIDP is assessed using its configuration parameters shown in Table 2.

Table 2 GIDP Configuration Parameters

Time interval TI	100 seconds
Training Period (N)	5 Time Intervals
Testing Period	20 Time Intervals
Number of Parameters	NCM =7 & DM=4 parameters
Chi-square test (α)	5% (i.e. 95% confidence interval)
Test Sliding Window	5 Time Intervals
Detections-Required-to-Accuse (*d*)	2 in a Test Sliding Window
Number of Intruders	Varies from 1 to 4

5.1 Scenario 1

In the first scenario we test GIDP with a denial of service attack (sleep deprivation) using malicious RREQ flooding (MRF), as described in Section 3-a. The intruders launch MRF1 or MRF2 attacks. At each tested mean speed and for each network size (either 25 or 50 nodes) we performed 40 runs with no intrusion and 40 runs with intruders, using a mix of both MRF1 and MRF2.

The graph in Fig.8a depicts the success rate (SR) and false alarm (FA) rate of GIDP as a function of the nodes' mean speed in 25 & 50 node networks with SD attack. By SR here we mean the rate of correctly detecting intrusion in the network, identifying the attack type and then identifying & isolating the node which is causing the attack. A false alarm (FA) means that a correctly behaving node has been incorrectly identified and isolated. The graph shows good performance of GIDP in terms of high SR and low FA rates against SD attack. The graph in Fig.8b shows the routing protocol overhead in a 25 node network when there is a) no attack in the network, b) a sleep deprivation attack with no protection and c) a sleep deprivation attack with GIDP in place. The graph shows that GIDP reduces the routing protocol overhead and increases network performance when it is used in a network under sleep deprivation attack.

5.2 Scenario 2

In the second scenario we test GIDP with a mix of black and grey hole attacks caused by initiating a false RREP and then dropping packets as described in section 3-b. In order to launch these attacks, on

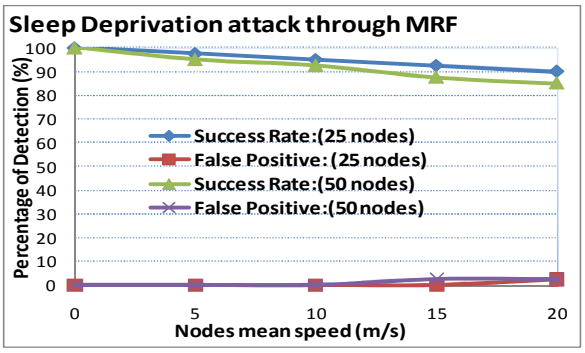


Fig.8a Success & false alarm rate of sleep deprivation attack(SD) as a function of nodes' mean speed

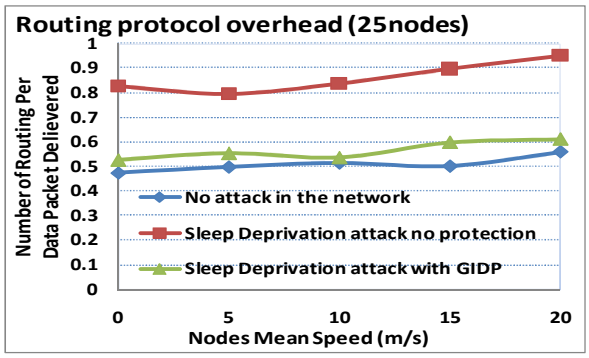


Fig.8b Routing protocol overhead with sleep deprivation attack as a function of nodes' mean speed

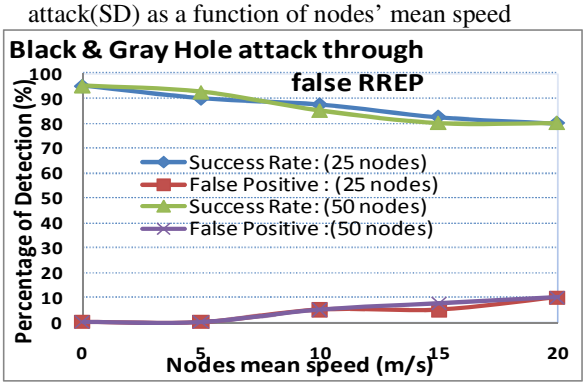


Fig.9a Success & false alarm rate of Black & Grey hole attacks (BH, GH) as a function of nodes' mean speed

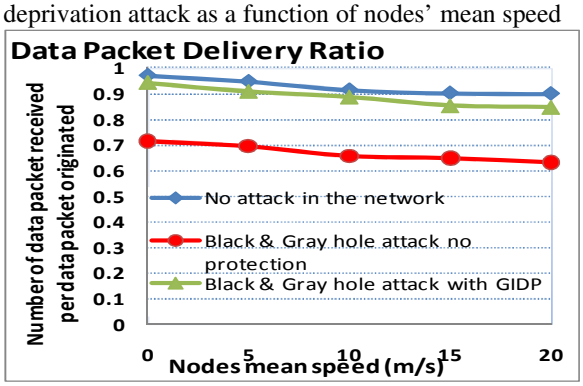


Fig.9b Data packet delivery ratio with Black & Grey hole attacks (BH, GH) as a function of nodes' mean speed

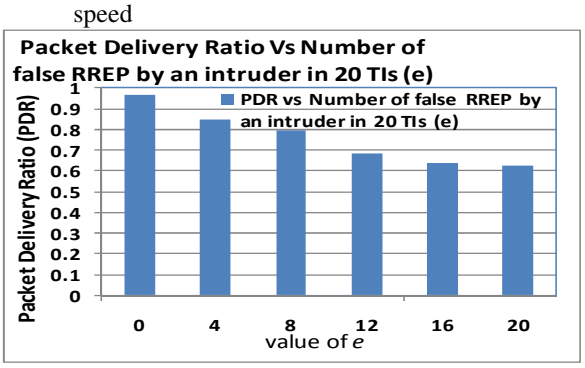


Fig.9c Data packet delivery ratio with Black & Grey hole attack as a function of nodes' mean speed

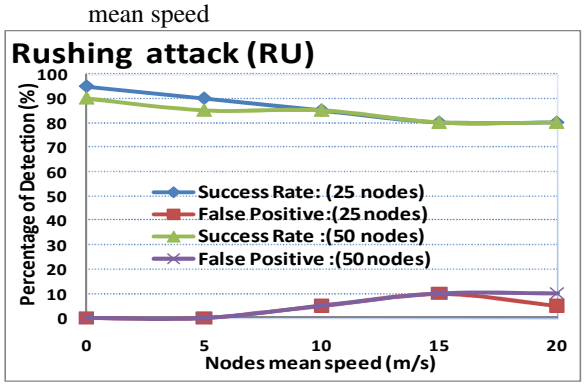


Fig.10a Success & false alarm rate of Rushing attack (RU) as a function of nodes' mean speed (m/s)

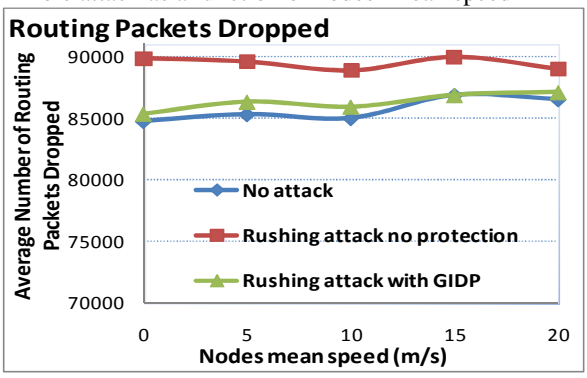


Fig.10b Control packet dropped with rushing attack as a function of nodes mean speed (m/s)

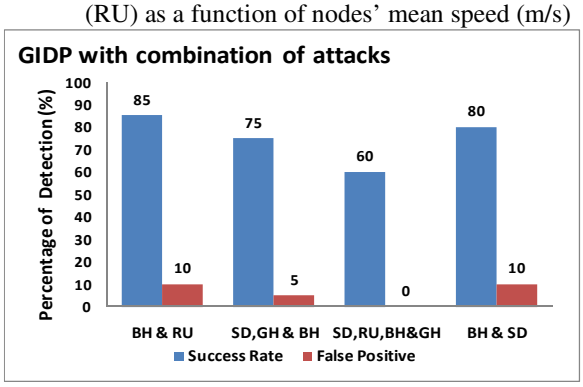


Fig.11 GIDP success & false alarm rates with different combinations of attacks

receiving a RREQ an intruder generates a false RREP packet with *dest_seq=current dest_seq+f*. Through simulations we observed that the value of *f* should be at least *5* in a 25 node network, and higher for larger networks, because some properly behaving nodes have routes fresher than the intruding node for the destination node. We also note that the severity of the attacks depends on the number of paths in the network that the intruder manages to capture. One false RREP packet only allows an intruder to capture the route of one node in the network, because RREP packets are unicast.

A single simulation consists of 20 test TIs. We monitor the number of false RREP packets (*e*) generated by an intruding node in a simulation and its impact on packet delivery ratio. Fig.9c shows that increasing the value of *e* reduces the packet delivery ratio during the BH attack and therefore increases the severity of the attack.

The graph in Fig.9a depicts the SR and FA of GIDP with black & grey hole attacks with $8 \leq e < 20$ and $5 \leq f \leq 30$. The graph in Fig.9b shows the packet delivery ratio with no attack, black & grey hole attack with no protection and black & grey hole attacks with GIDP in place. It shows that GIDP can successfully detect these attacks, and identify & isolate the intruding node and by doing so GIDP also improves the network performance in terms of packet delivery ratio.

5.3 Scenario 3

In this scenario we test GIDP with the rushing attack through forged RREQ as explained in section 3-c. We note that intruders trying to cause rushing attacks by sending a forged RREQ with a higher *src_seq* and minimum delay increase the number of routing packets (i.e. RREQ+RREP+RERR) dropped in the network. Fig.10a shows that GIDP can detect rushing attacks and after isolating, the intruder reduces the number of routing packets dropped as shown in Fig.10b.

5.4 Scenario 4

In the final scenario we assess GIDP with a combination of simultaneous attacks launched by separate intruders in a simulation. We perform 20 runs with each combination of attacks. SR here means that GIDP has detected, identified and isolated *all* the intruders causing attacks. FA means GIDP has detected and isolated a properly behaving node as an intruder. Fig.11 depicts the success rate and false alarm rate of GIDP for each of the attack combinations simulated. The graph shows the ability of GIDP to detect and isolate attacking nodes, and demonstrates the generality of our proposed mechanism. During the simulations GIDP flagged a *POTENTIALUNKNOWNATTACK* on a few occasions but they did not meet the criteria of GIDP attack inferences (i.e. *d* detections of same rule trace in a *TSW*) (Fig.6)) to mark them as a new attack.

5.5 Analysis of GIDP Overhead

We now consider the overhead imposed on the MANET by GIDP. We assess the network overhead, measured in number of packets (evaluated as number of packet generated * number of hop the packet travels) generated by GIDP as a function of the nodes' mean speed, and compare it with (a) the AODV routing protocol overhead and (b) the network traffic produced by the Constant Bit Rate (CBR) connections. CBR traffic generated at the application layer during the simulation results in User Data Datagram Protocol (UDP) traffic at the network layer. GIDP traffic consist of the NCM packets sent periodically from CNs to CHs and the Accusation Packets generated by CHs to inform CNs about the intruders in the network. The AODV overhead consists of all the control packets i.e. RREQ, RREP and RERR packets generated in the network during the simulation. Although packets in these three types of packets differ in size, the comparison still gives us a useful indication of the relative contributions made to the total network traffic.

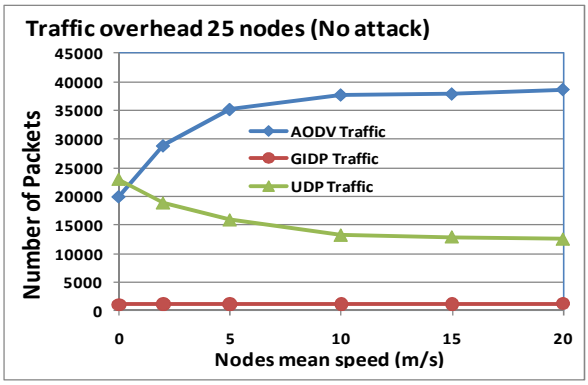


Fig.12 Overhead generated by AODV, GIDP and UDP traffic on the network with no attack.

To analyze the network overhead in terms of AODV, GIDP and UDP traffic we first consider a 25 node network with no attacking node and then 25 nodes with SD attack as an example. We perform 10 runs with nodes' mean speed varying from 0 to 20 m/s. Graphs in Fig.12 & Fig.13 shows the contribution made to the total network traffic by the three components as a function of the nodes' mean speed with no attack and SD attack respectively. We note from the graphs that the AODV overhead rises and the UDP traffic falls with increasing node mean speed, while the GIDP overhead

is independent of node speed. The GIDP traffic on average contributes to 2.6% of the total network traffic, a very low sum.

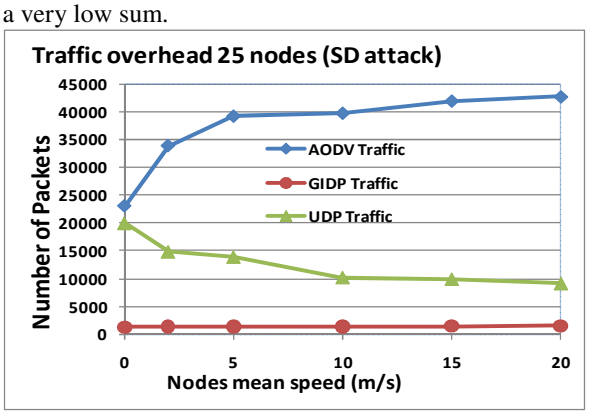


Fig.13 Overhead generated by AODV, GIDP and UDP traffic on the network with SD attack.

6 Impact of Attacks & Intrusion Response

We analyze the simulation results of the case study of Section 5, where we assess the applicability of GIDP to various classes of attacks. We notice that in each scenario with a specific attack a certain parameter of our derived matrix (DM) is affected most. For example in scenario 1 (sleep deprivation attack) the routing protocol overhead (RPO) of the network increased significantly. In scenario 2 (black & grey hole attacks) the data packet delivery ratio (PDR) decreased considerably. In scenario 3 (rushing attack) we observe the increase in the number of control packets dropped (CPD) during the routing operation in the network. This indicates that each attack studied has somehow affected the network performance, but does not give us a clear picture of how severe these individual attacks are for the network. Therefore in this section we investigate the effects of various attacks and then the impact of intrusion response on overall network performance.

6.1 Impact of attacks on network performance

To evaluate the impact of various attacks on the network performance we ensure that the matrix we use illustrates changes and effects that are caused by specific attacks in MANETs. We use all four parameters of our Derived Matrix i.e. Throughput, PDR, RPO & CPD, and we model these parameters when there is no attack taking place in the network and then model them with sleep deprivation (SD), black hole (BH) & grey hole(GH), rushing (RU) and sybil (SY) attack to measure the network performance degradation using equation 3.

$$NPD = w_1 * \Delta Throughput + w_2 * \Delta PDR + w_3 * \Delta CPO + w_4 * \Delta CPD \ldots\ldots(3)$$

where W_i represents the weights, $\sum_{i=1}^{4} W_i = 1$. We analyse the importance of throughput, PDR, RPO & CPD in measuring the overall network performance through literature [17, 20, 27] and simulation results. We observe that throughput and PDR are more significant than RPO and CPD. Therefore, to illustrate the impact of attacks and the impact of intrusion response in the case study in this paper we use the following weights in equation 3: w_1=0.5,w_2=0.3,w_3=0.1 & w_4=0.1. In equation 3, Δ represents the percentage change, for example $\Delta Throughput$ is the percentage change in throughput with and without an attack in the network.

6.1.1 Impact of various attacks

In this study we used the simulation parameters of Table 1. With the 25 node network we first perform simulations with no attack in the network and model the DM parameters. Keeping the same simulation environment we then perform 10 runs with a randomly picked node causing a black hole attack in the network and estimate the network performance degradation (eq.3) when no GIDP is in place. We repeat the same process for sleep deprivation, rushing and sybil attack with a single attacker. Then the entire process is repeated with the 50 nodes network. The graph in Fig.14 shows that some attacks are more severe than others. Specifically, the black hole attacker has the highest impact on network performance. An attacker causing sleep deprivation also has a significant impact while rushing and sybil attacks have the lowest influence on network performance.

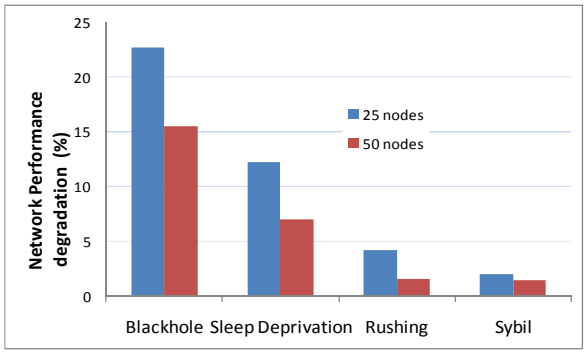


Fig.14 Impact of various attacks on network performance

6.1.2 Impact of combination of attacks

In next set of experiments we evaluate the impact of combinations of attacks with more than one intruders. We experiment with various combinations of simultaneous attacks (section 3) launched by separate intruders. The graph in Fig.15 shows that the overall performance of the network degrades further when more than one intruder is present in the network. We observe that all combinations with black hole attacks have caused more damage to the network than any other combinations. We also notice that when we analyze each attack independently the sybil attack has the least effect on network performance as shown in Fig.14, but when it is used with a combination of other attacks it has caused a significant impact on network performance as shown in Fig.15.

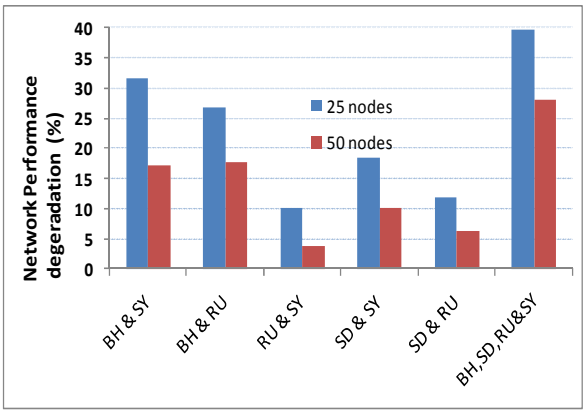


Fig.15 Impact of combination of simultaneous attacks on network performance

6.2 Impact of intrusion response on network performance

When an intrusion is detected and the intruder is identified in GIDP the intrusion response is called by CH as shown in Fig.1. In response to the intrusion GIDP isolates the intruding nodes from the network. To get an estimate of the impact of GIDP intrusion response (isolation) on network performance, we randomly isolate the properly behaving node in the network when there is no attack and no GIDP in place and evaluate the network performance degradation using equation 3.

We first set up a 25 node network using the simulation parameters of Table 1 with no attack and no GIDP in place. Nodes in the network are set to move according to RWP model with mean speed of 5 m/s. We perform 10 runs and in each run we randomly choose a node and isolate it from the network. We then repeat the same process with the 50 node network.

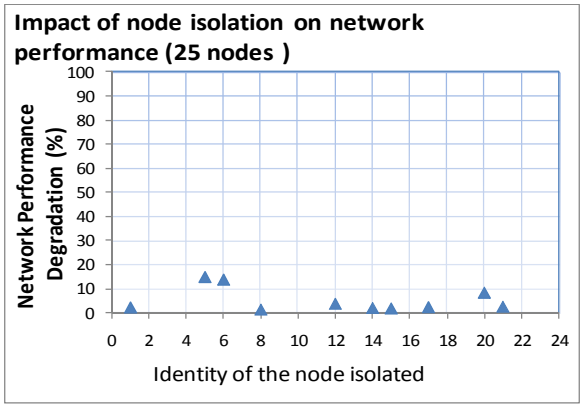


Fig.16 Impact of node isolation on network performance in a 25 node network

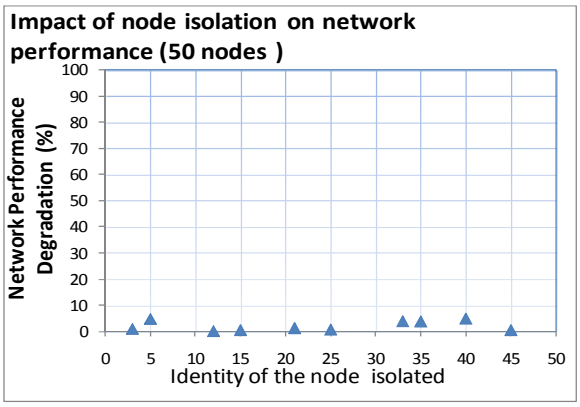


Fig.17 Impact of node isolation on network performance in a 50 node network

The graphs in Fig.16 and Fig.17 depict the impact of isolating a randomly picked node on the overall network performance of the 25 and 50 node networks respectively. In general these graphs illustrate that some nodes in the network are more critical than others because of their location in the network topology, but few nodes have a major role as routing nodes in the network, primarily because the nodes are moving and therefore the critical routing nodes change with time. Isolating the more important routing nodes, for example node 5, 6 & 20 in Fig.16, affects more routes in a network than other nodes and re-routing causes significant routing disruption, which degrades network performance considerably.

We compare the results of the impact of attacks (Fig.14) and the impact of isolating nodes (Fig.16 & Fig.17) on network performance. We note that in some cases when attacks are less severe (for example rushing or sybil attacks, Fig.14) and nodes are more critical (for example nodes 5 or 6, Fig.16), the intrusion response of completely isolating these nodes actually results in a net degradation of network performance. Specifically, in these cases it is actually better *not* to punish an

attacking node by isolating it. In other words, an intrusion response should be more flexible and should be able to tradeoff between the impact of the attack and the impact of isolating the attacker from the network.

7 Conclusions

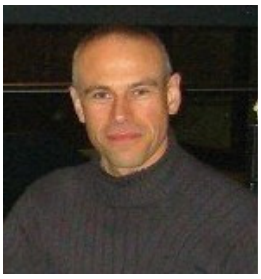
In MANETs considerable interest has recently been devoted to mechanisms that enforce security. Many proposals have been made in the literature to secure MANETs from various attacks, but most are attack-specific. Unlike some mechanisms that provide protection through authenticated routing, the Generalized Intrusion Detection & Prevention mechanism that we have proposed in this paper monitors both network layer characteristics (NCM) and performance statistics (DM). GIDP uses a combination of anomaly-based and knowledge-based ID that can protect MANETs against a variety of attacks. Simulation results of our case study show that our approach can protect MANETs from a wide variety of attacks with an affordable processing overhead. We also investigated the severity of various attacks and their impact on network performance along with the impact of the GIDP intrusion response on network performance. The results shows that in some cases isolating the attacker can cause more harm than good to network, hence an adaptive flexible intrusion response mechanism is required. This will be our focus of research in future.

References

1. E.Perkins and M.Royer, "Ad Hoc On Demand Distance Vector Routing", Sun Micro System Laboratories Advance Development Group, Proceeding of IEEE MOBICOM, pp 90-100, 1999.
2. B.Jhonson and A.Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks", In Mobile Computing Journal, Vol.353, pp 153-181, 1996.
3. P.Jacquet, P.Muhlethaler ,T.Clausen,A.Laouiti and L.Viennot , "Optimized link state routing protocol for ad hoc networks", Proceeding of IEEE INMIC 2001.
4. Z.Li, A.Das, and J.Zhou, "Theoretical Basis for Intrusion Detection", IEEE work shop proceedings on Information assurance and security. pp 184-192, 15-17 June 2005.
5. K.Ilgun, R.A.Kemmerer, and P.A.Porras, "State transition analysis: A rule based intrusion detection approach", IEEE Transactions on software Engineering, Vol.21, No.3, pp 181-199, March 1995.
6. A.Nadeem and M.Howarth, "Adaptive intrusion detection & prevention of Denial of Service attacks in MANETs", Proceeding of ACM 5th International Wireless Communication and Mobile Computing Conference, Germany, June 2009.
7. Y.Zhang and W.Lee," Intrusion Detection in Wireless Ad-Hoc Networks", Proceeding of 6th, ACM MOBICOM, 2000.
8. A.Hijazi and N.Nasser "Using Mobile Agent for Intrusion Detection in Wireless Ad-Hoc Networks", Proceeding of IEEE WCNC 2005.
9. F.Cretu, J.Parekh, Wang and J.stolfo "Intrusion and Anomaly Detection Model Exchange for Mobile Ad-Hoc Networks**",** Proceeding of IEEE Consumer Communication and Networking Conference 2006.
10. H.Jiang and H.Wang "Markov Chain Based Anomaly Detection for Wireless Ad-Hoc Distribution Power Communication Networks", Proceedings of IEEE Power Engineering Conference 2005.
11. E.Padilla, N.Aschenbruck, P.Martini, M.Jahnke and J.Tolle, "Detecting Black Hole Attack in Tactical MANETs using Topology Graph", Proceeding of 32nd IEEE Conference on Local Computer Networks, 2007.
12. S.kurosawa and A.Jamalipour, "Detecting Blackhole Attack on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method", International Journal of Network Security, Vol.5, November 2007.
13. G.Xiaopeng and C.Wei, "A Novel Grey Hole Attack Detection Scheme for Mobile Ad-Hoc Networks", Proceeding of IFIP International Conference on Network & Parallel Computing, 2007.
14. J.Sen ,M.Chandra, Harihara S.G, H.Reddy and P.Balamuralidhar , "A mechanism for detection of Gray Hole attack in Mobile ad hoc network", Proceding of IEEE ICICS 2007.
15. P.Yi, Z.Dai and S.Zhang, "Resisting Flooding Attack in Ad Hoc Networks", Proceeding of IEEE Conference on Information Technology: Coding and Computing", Vol.2, pp 657-662, 2005.
16. W.Yu and K.Ray," Defence against Injecting Traffic Attack in Cooperative Ad Hoc networks", IEEE Global Telecommunication Conference, Globecom, 2005.
17. Y.Hu, A.Perrig and B.Johnson, "Rushing Attack and Defense in Wireless Ad Hoc Networks Routing Protocols", Proceeding of 2nd ACM workshop on Wireless Security, New York, 2003.
18. K.Sanzgiri and M.Belding-Royer, "A Secure Routing Protocol for Ad Hoc networks", Proceedings of 10th IEEE International Conference on Network Protocol 2002, (ICNP' 02).
19. Y.Hu, A.Perrig and B.Johnson, "A Secure On Demand Routing Protocol for Ad Hoc networks", Proceeding of MobiCom, Atlanta, Georgia, USA, pp 23-28, September 2002.
20. J.Joseph, A.Das, B.Seet and B.Lee "CRADS: Integrated Cross Layer approach for Detecting Routing Attacks in MANETs", Proceeding of IEEE WCNC, 2008.
21. Y.Hu, B.Jhonson and A.Perrig, " SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks", Ad hoc Networks, Vol.1, pp 175-192, 2003.
22. M.Pirrete and R.Brooks, "The Sleep Deprivation Attack in Sensor Networks: Analysis and Methods of Defence", International Journal of Distributed Sensor networks, Vol.2, No.3, pp 267-287, 2006.
23. C.Piro, C.Shields and B.Levine, "Detecting the Sybil Attack in Mobile Ad hoc Networks", Proceedings of IEEE International Conference on

Security and Privacy in Communication Networks, 2006.
24. KDD data set, 1999. URL:http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.
25. A.Nadeem and M.Howarth, "A Generalized Intrusion Detection & Prevention Mechanism for Securing MANETs", Proceedings of IEEE International Conference on Ultra Modern Telecommunications & Workshops, St Petersburg Russia 2009.
26. J.Nuevo, "A Comprehensive GloMoSim Tutorial", INRS telecom, 2004.
27. K.Agarwal and W.Wang, "Statistical Analysis of the Impact of Routing in MANETs Base on Real-Time Measurement", Proceedings of IEEE ICCCN 2005.

Adnan Nadeem received his bachelor's degree BSc and master's degree MCS masters in computer science, both from Faculty of Science University of Karachi. He is currently working towards the PhD degree in The Faculty of Engineering and Physical Sciences, University of Surrey, UK. His principal research interest includes security issues in wireless ad hoc networks, intrusion detection & prevention and secure routing. He is a student member of IEEE and IEEE Communication Society.



Michael Howarth received the bachelor's degree in engineering science and the DPhil degree in c electrical engineering, both from Oxford University and the MSc degree in telecommunications from the University of Surrey, United Kingdom. Prior to joining the University of Surrey, he worked for several networking and IT consultancies. He is a lecturer in networking at the Centre for Communication Systems Research (CCSR), University of Surrey. His research interests include traffic engineering, quality of service, security systems, protocol design, and optimization of satellite communications. He is a chartered electrical engineer and a member of the United Kingdom IET.