Adaptive Intrusion Detection & Prevention of Denial of Service attacks in MANETs

Adnan Nadeem
Centre for Communication Systems Research
University of Surrey, UK
a.nadeem@surrey.ac.uk

Michael Howarth
Centre for Communication Systems Research
University of Surrey, UK
m.howarth@surrey.ac.uk

ABSTRACT
Mobile ad-hoc networks (MANETs) are well known to be vulnerable to various attacks, due to features such as lack of centralized control, dynamic topology, limited physical security and energy constrained operations. In this paper we focus on preventing denial-of-service (DoS) attacks. As an example, we consider intruders that can cause DoS by exploiting the route discovery procedure of reactive routing protocols. We show the unsuitability of tools such as control chart, used in statistical process control (SPC), to detect DoS and propose an anomaly-based intrusion detection system that uses a combination of chi-square test & control chart to first detect intrusion and then identify an intruder. When the intruder is isolated from the network we show reduced overhead and increased throughput. Simulation results show that AIDP performs well at an affordable processing overhead over the range of scenarios tested.

Categories and Subject Descriptors
D.2 [**Computer-Communication Networks**]:
Wireless communication, Security and protection.

General Terms
Security, Algorithms.

Keywords
Ad-hoc network security, intrusion detection & prevention.

1. INTRODUCTION
MANETs have a decentralized architecture and lack of centralized control; consequently, mechanisms that enforce security present a particular challenge. In fixed networks, intrusion detection and prevention (IDP) [1] acts as a second layer of defence beyond a firewall; whereas in MANETs IDP becomes the front line of defence to protect nodes from attacks. There are two ID techniques known as misuse-based intrusion detection (MBID) and anomaly-based intrusion detection (ABID). MBID maintains a knowledge base containing signatures or patterns of known attacks and looks for these patterns in an attempt to detect them. MBID has a potentially low false detection rate but it can only detect attacks whose signatures are in the database. On the other hand ABID can flag observed activities that deviate significantly from the established normal profile. ABID not only provides early warnings of potential intrusions but also can detect attempts to exploit new and unforeseen vulnerabilities; however it is more prone to generate false positives than MBID.
Routing protocols in MANETs are generally classified as either proactive or reactive. Proactive routing protocols such as DSDV and WRP are not efficient because of their routing traffic overhead and therefore reactive routing protocols such as AODV and DSR are most frequently used in MANETs. However, both AODV and DSR operate on the assumption that all nodes in the network can trust each other and there are no malicious intruder nodes. This is not true in all cases, and therefore we believe that there is a need to use an IDP system to provide secure routing in MANETs. In this paper we illustrate how intruders can exploit the route discovery procedure of reactive routing protocol to cause certain DoS attacks in MANET. We then look at the detection of DoS; we assess control chart, a tool used in statistical process control (SPC)[2] and show that it generate low detection & high false alarm rates. We therefore consider a two stage process: we employ the chi-square goodness of fit test [3] as an ABID mechanism to initially check the overall behaviour of the network and indicate intrusion; in the event of a positive result we then use control chart to identify intruding nodes. Finally we isolate the nodes from the network to prevent intrusion. We call our algorithm Adaptive Intrusion Detection & Prevention (AIDP).

Section 2 of this paper describes related research & challenges in ID and securing MANETs. Section 3 describes DoS attacks and indicates how an intruder can cause DoS in MANETs. Section 4 presents our proposed algorithm. Simulations and their result are illustrated in Section 5. We summarize our results and future work in Section 6.

2. RELATED RESEARCH & CHALLENGES

2.1 Securing MANETs
SEAD was proposed in [4] as a secure routing protocol that uses a one-way hash function to provide authentication for the proactive routing protocol DSDV. The secure routing mechanism ARAN was proposed in [5]. A similar approach Ariadne [6] has been proposed for end-to-end authentication based on shared key pairs. These methods provide authenticated routing & ensure integrity of routing information mainly to prevent routing attacks caused by modification of control packets or forged routing information. However, a MANET node can, without modifying any control packets exploit the route discovery procedure of reactive routing protocols to cause DoS attacks as we describe below in section 3.

Wang, Lu and Bhargava [7] performed a vulnerability analysis of AODV in which they observed that on-demand route queries enable real time attacks. Some researchers have proposed methods to detect this: for example Ping and Zhang [8] considered a route request (RREQ) flooding attack in MANETs. They proposed a RREQ flooding prevention mechanism based on neighbour's supervision that maintains a priority queue of the incoming RREQs. This mechanism reduces the priority of RREQ generated by a specific node if a higher rate of incoming queries from that particular node is observed. However in some applications of MANETs there can be specific nodes that generate more traffic; for example, in on-the-fly networks formed for a seminar and yet Ping & Zhang's method will remove requests from the queue above a certain incoming request rate

in all cases. In another example Yu and Ray [9] defined two types of injecting traffic attack in MANETs as query and data packets 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 which in our opinion does not cope well with the dynamically changing environment of MANETs. In addition, there is also the need to address the issue of isolating a node once it is detected as an attacker.

2.2 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 dynamically changing topology and limited computational ability of nodes. Zhang and Lee argue [10] that many ID techniques developed for fixed wired networks are not applicable in MANETs and they proposed 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. Nguyen *et al*. [11] proposed ID through a statistical anomaly detection approach called Principal Component Analysis (PCA) which is used as an outlier detector method for high speed fixed networks. Ye *et al*. [12] used a probabilistic technique for anomaly detection in fixed networks. They investigated audit data by using various probabilistic techniques including decision tree, Hotelling's T^2 test, chi-square test and Markov chain for detecting intrusion into the information system on fixed networks and concluded that chi–square test based on frequency property provides good ID performance. Ye and Chen [13] also proposed an anomaly detector based on the chi-square test for detecting intrusion in fixed networks. They concluded that the results demonstrate promising performance in terms of high detection and low false alarm rate. The chi-square test has been successfully used for ABID in fixed network but in a MANET where there is no existing knowledge of normal behaviour we have extra challenges to apply these ABID techniques.

3. DENIAL OF SERVICE ATTACKS

DoS is an attempt to make resources or services unavailable to their intended users. Distributed DoS is a severe threat for MANETs because they can be crashed due to their limited battery power or their network can easily become congested due to its relatively limited bandwidth compared to fixed networks. Some examples of DoS attack on reactive routing protocol such as AODV and DSR are as follows:

Sleep Deprivation: here [14] the attacker interacts with the node in a manner that appears to be legitimate; however the purpose of interaction is to keep the victim node out of its power conserving sleep mode. An intruder can cause sleep deprivation by exploiting the vulnerability of the route discovery process of protocols such as AODV and DSR, for example, by sending a RREQ packet periodically so that the victim node has to process these packets causing exhaustion of its battery power.

Rushing Attack: In order to limit the control packet overhead an on-demand protocol only requires nodes to forward the first RREQ that arrives for each route discovery. An intruder can exploit this property by spreading RREQ packets quickly throughout the network so as to suppress any later legitimate RREQ packets [15]. In the work described in this paper we have initially considered vulnerabilities in AODV but AIDP can be applied to other routing protocols.

3.1 Vulnerabilities in AODV

AODV is designed for use in networks where the nodes can all trust other nodes and can assume there is no malicious intruder node. Considering the operation of AODV [16], specifically its route discovery process, it is highly vulnerable to DoS attacks such as sleep deprivation and the rushing attack. In the route discovery procedure of AODV when a node needs a route to a destination, it broadcasts a RREQ packet containing a broadcast id, source & destination addresses, hop count and destination sequence number. After broadcasting a node is required to wait for a specific time for a RREP or other control packet. The node may try again once this time expires. The source node is expected to use an expanding ring search technique for controlled dissemination of RREQs in the network. This means that after sending a RREQ with Time to Live (TTL) field set to one the node can resend the RREQ with an incremented TTL value after waiting for *ring traversal time.* The node can repeat this process until either a RREP or route error is received or the TTL value reaches its maximum value. After that the node can retry the same path discovery for a specific destination up to some defined maximum number of RREQ retries.

3.2 Malicious RREQ Flooding

We initially focus on malicious RREQ flooding (MRF) which can cause DoS attacks such as sleep derivation & rushing attack in MANETs. In MRF an intruder exploits the route discovery process of a reactive routing protocol such as AODV to cause DoS in the network. An intruder can flood the network with malicious RREQs without being detected by other nodes in the following ways:

Malicious RREQ Flooding 1: an intruder broadcasts a RREQ with a destination IP address that is within the domain but 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 same destination with higher TTL value.

4. AIDP

We now describe AIDP, which uses ABID to detect DoS attacks caused by MRF in MANETs. AIDP consists of two modules: a training and a testing module as explained later in this section.

4.1 Model Assumptions & Terminologies

We disregard attacks aimed at the physical and link layers. We consider applications of MANETs formed on-the-fly for group collaboration such as rescue operations or seminars. We note that an ABID requires data of only normal activities containing audit traces and traffic patterns of normal events to build a training profile. However, in contrast with fixed networks, data resources such as [17] that reflect normal activities or events are not available for on-the-fly MANETs applications. In general the normal operation of MANETs is not known. Therefore we assume that the initial behaviour of the network formed on-the-fly during the training phase is free from anomalies. To illustrate the implementation of AIDP we assume a clustered MANET organization. We select the most capable nodes in terms of their processing abilities as cluster head (CH) and the others nodes becomes cluster nodes (CN). At present we assume secure communication between CH and CNs.

The operation of AIDP is illustrated in Fig.1. When the network is established, the CH continuously gathers information and applies the AIDP training module for *N* time intervals (TI), resulting in an initial training profile (ITP). The ITP reflects the normal behaviour of the nodes in the network. In the testing phase the CH then applies the

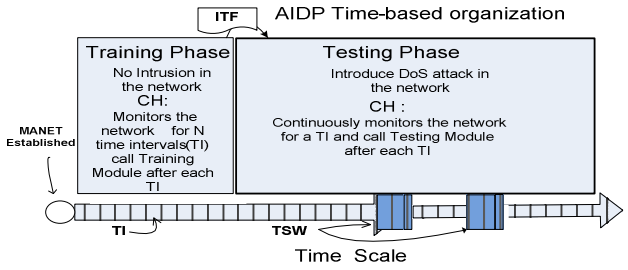

Figure 1. Time-based operation of AIDP.

testing module after each TI. This test consists of several tasks, the first of which detects intrusion. If there is no intrusion then it updates the ITP 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 intruding nodes. To optimise the probability of identifying intruders correctly with a low level of false positives, it maintains a test sliding window (TSW), 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 (BL) the node and isolate the node by informing all CNs.

4.2 AIDP Algorithm

We now explain the training and testing module of AIDP.

Training Module:

While i is less than equal to N
.CH collects the number of RREQ received X_i by CN
from all other CNs taking into account TTL values in TI.
. Calculate the probability distribution P(X_i)
end while
.calculate mean $\overline{X}_i$ of P(X_i) for i=1 to N
.Store results as ITF.
. Exit

Figure 2. Pseudocode of training module.

In Fig.2 X_i ={X_1, X2, X3,…,X_M} is a set of random variables representing the number of RREQs received by all CNs in the *ith* TI, where M is the maximum number of RREQs received in a TI. This includes both the RREQ packets generated by the source nodes and those RREQ packets forwarded by intermediate nodes. The probability distribution of X_i is calculated for the TI, and this process is then repeated for the *N* time intervals in the training phase. We then calculate mean $\overline{X}_i$ of P(X_i) for *N* intervals, which is stored as an ITP containing the expected values for that particular network observed for the total time of *N*TI* seconds. The training module pseudocode can of course be generalized to collect other parameters.

Testing Module:

a) *Detecting Intrusion & Calling other Modules*

.CH sets TSW to P number of TI
.CH Monitor the network for TI
Do after each TI
. CH collects number of RREQ received X_i from all other CNs in TI.
. Calculate the probability distribution P(X_i)
.Calculate average of P(X_i)
.Store X_i as Observed values.
. End do
.CH Applies the chi-square test by first Calculating Chi computed (χ^2)
.Hypothesis Testing
H_o: Observed distribution of X_i fits the expected
H_a: Observed distribution of X_i does not fit expected
.If (chi-computed (α.d.f) > P-value (α.d.f)) then
Reject H_o & call: LND= Intruder-Identification(nodes V_i)
For all nodes V_i in LND (List of Nodes Detected)
.If (V_i detections in PIL > Detections_To_ Accuse)
CH: Blacklist V_i & Broadcast AccusationPacket(AP)
else : enter V_i in PIL
endif. else : Update Expected values $\overline{X}_i$ (TP)
.endif
.Exit

b) *Intruder-Identification (nodes V_i)*

.calculate RREQ generated by V_i for i=1 to n (n=number of nodes)
.calculate standard deviation σ of V_i
.set Contol Line (CL)
.set Upper Control Limit (UCL) & lower control limit(LCL)
For V_i i= 1 to n
If (RREQ generated by V_i)> UCL
add V_i to LND
. endif
.endfor
.return (LND) & Exit

c) *Accusation Packet (AP) Handling*

. each CN V_i maintain its local BlacklistTable (BLT)
.if CN V_i receive an AP for CN V_j
.If CN V_i has node V_j in its BLT
Ignore AP
else
CN add node V_j to its BLT & rebroadcast AP
.endif
.endif

d) *Isolating Intruding Node*

.if node V_i receive packet from node V_j
.If node V_j is in node V_i BLT
Ignore packet & drop all packet queued from V_j
Else : handle & process packet
.endif
.endif

Figure 3. Pseudocode of testing module.

Fig.3 shows the pseudocode of the testing module. After monitoring the network for one TI, the CH uses the chi-square test to identify any intrusion. This test determines how well the observed model fits with the expected.

$$\chi^2 = \sum_{i=1}^{N} \frac{(X_i - \overline{X}_i)^2}{\overline{X}_i} \qquad \textbf{(1)}$$

Equation 1 is the specific form of the test in which X_i is the observed and $\overline{X}_i$ is the expected value of the *i*th variable. After calculating the chi-computed value the CH performs hypothesis testing by setting the null hypothesis H_o and alternative hypothesis H_a as shown in Fig.3a. The critical P-value is calculated at given level of significance (α) and degree of freedom (d.f). To illustrate the operation of the algorithm we have chosen the standard value of α=5% (i.e. a confidence interval of 95%). The d.f is the number of classes of X_i (i.e. the number of groups in which the frequency of RREQ is divided) being tested which is in our case is determined by the testing module of AIDP for each TI at run time. If calculated chi-computed value is larger than the critical value then we reject H_o ,and assume intrusion in the TI. We then use intruder-identification task (Fig.3b) to identify the individual intruding node. This uses variable control chart based on standard deviation σ. We calculate σ of RREQ generated by all nodes then set the CL= σ, UCL=CL + 3σ & LCL= minimum [0 , CL - 3σ]. We choose 3σ limits because we know that for a normal distribution 99.7% of the observation lies within ± 3σ limits. We consider node V_i to be a detected intruder if it initiates more RREQs than the UCL, in which case we add node V_i to the potential intruder list (PIL) maintained by CH. If any node V_i is detected more than

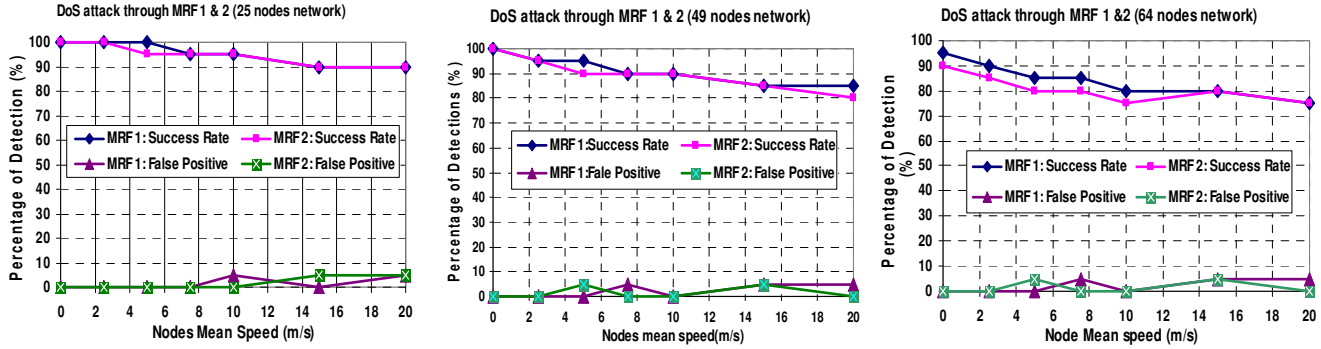

Figure 4. Success and false alarm rate as a function of node mean speed in a 25, 49 & 64 nodes network.

d times in *P* intervals (the threshold for accusation *Detection_to_Accuse*) then the CH blacklists the node and inform 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 form the network all nodes will not only drop the packets from a blacklisted node but also immediately ignore all packets in their queue from the blacklisted nodes as shown in Fig.3d. If no intrusion is detected by the chi-square test (Fig.3a) then we updates the training profile using an exponentially weighted moving average (EWMA) as given in equation 2,

$$\overline{X}_{(J,I_1^k)} = \alpha * X_{(J,I_1^k)} + (1-\alpha) * \overline{X}_{(J-1,I_1^k)} \qquad (2)$$

where $\overline{X}_{(J,I_1^k)}$ and $X_{(J,I_1^k)}$ represents the expected and observed value for update period number (*J*) respectively. The value of *J* is incremented in the TI when no intrusion in the MANET is detected. *I* represent the random variable from 1 to k and $\alpha = \frac{2}{(J-1)}$ is the weighting factor. As *J* increases the weighting for older data points decreases exponentially giving more importance to the current observation. The updated expected profile model reflects the current behaviour of the network. This is important for adaptive ID in MANET where overall behaviour of the network changes with time.

5. EVALUATION

5.1 Simulation Environment

We use GloMoSim to assess the performance of AIDP. We build the simulation environment by assuming that the MAC and Physical layer are reliable in their operations. Table 1 shows the simulation parameters for all scenarios. The nodes are initially placed at the start of simulation in a rectangular grid. The terrain dimension values in Table 1 ensure node density is constant between all three scenarios. We assume a single cluster in our simulations, and we use the random way point (RWP) as mobility model.

5.2 Assessment of control chart as an Intrusion Detector

In first set of experiments we use only control chart based on standard deviation σ to detect DoS. The CH monitors the network for a TI and then applies control chart based on σ on the number of RREQs generated by all nodes. The CH calculates the CL, UCL & LCL (Fig.3b). The CH will detect as intruder any node V_i that generates more RREQ than the UCL. This process is repeated by the CH for each TI. We perform 20 runs first with normal traffic (i.e. no intrusion in the network) and then a further 20 runs with one intruder picked randomly from the nodes. This intruder launches MRF attacks in order to cause DoS by sleep deprivation and rushing attack. Simulation with 25 , 49 & 64 nodes results in an average successful identification rate of 70%, but also has a very high average false alarm rate (i.e. detecting a node as an intruder when there is no intrusion in the network) of 55%.

Table 1. Simulation Parameters

Number of nodes	25	49	64
Terrain dimensions	400*400 m	560*560 m	640*640 m
Number of intruders	1 or 2		
Node placement	Grid with grid unit=10 metres		
Time interval TI	100 seconds		
Simulation time	Training + Testing =500+2000=2500 seconds		
Routing protocol	AODV		
MAC protocol	IEEE 802.11		
Mobility	Random Way Point Model (RWP)		
Pause time	Varies from 10 to 60 seconds		
Mean speed	Varies from 2 to 20 m/s		

5.3 Evaluation of AIDP

In this section we present the results from AIDP. We experimented with all three scenarios (25, 49&64 nodes) using the parameters of Table 1.

By introducing the chi-square test in addition to control chart the individual identification rate rises to 86% and the false alarm rate drops to 15% for a test in a single TI. This significantly improves the detection rate.

In simulation for AIDP the CH applies the training module (Fig.2) for *N*=5 TI, and then applies the testing module (Fig.3) for 20 TI each of 100 seconds. We illustrate results here with TSW=5 and *Detection_to_Accuse=3*. We perform 20 runs with each scenario (25, 49 & 64 nodes) with normal traffic using the simulation parameters of Table 1 and then with intruders picked randomly from the nodes. These intruders launch MRF 1 & 2 attacks in order to cause DoS by sleep deprivation and rushing attack. At each tested mean speed we perform 20 runs with no intruders and 20 runs with one intruder using MRF1 and 20 runs with on intruder using MRF2.

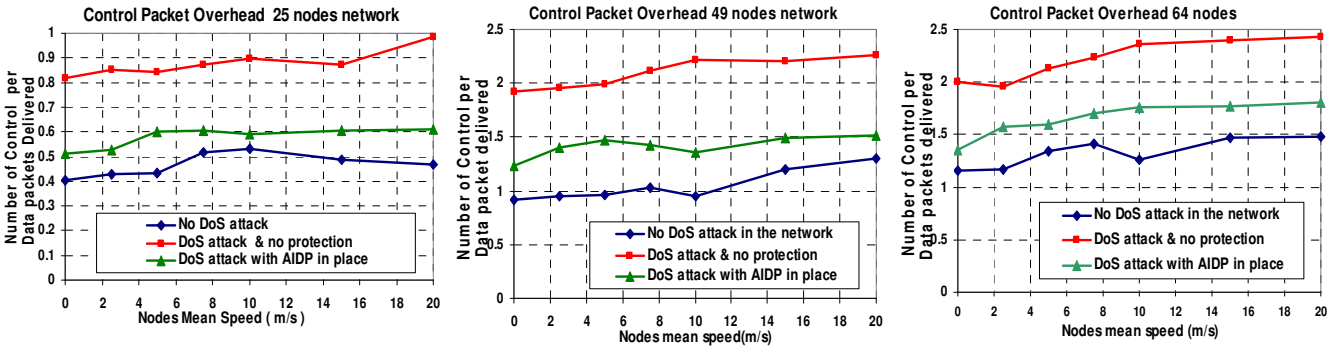

Figure 5. Control packet overhead Vs nodes mean speed in a 25, 49 & 64 nodes network.

Fig. 4 depict the success rate (SR) and false alarm (FA) rate of AIDP as a function of the nodes' mean speed in 25 ,49 & 64 node networks .By SR here we mean the rate of correctly indicating intrusion in the network and then identifying & isolating the node which is causing DoS. A false alarm (FA) means that a correctly behaving node has been incorrectly identified and isolated. When there is no intrusion in the network the FA are zero in all three scenarios. The graphs show good performance of AIDP in terms of high SR and very low FA rate against DoS attacks through MRF 1 & 2. The SR drops slightly in the 64 node network when the nodes are moving with a higher mean speed.

Effects on network performance

To analyze the performance impact of AIDP on the network we monitor control & data packets during our simulations. Fig 5 depicts the control packet overhead as functions of increasing mean node speed in 25, 49 & 64 nodes network. By control packet overhead we mean the ratio of the number of control packets to the delivered data packets during the simulations. Each graph displays the control packet overhead when there is a) no DoS attack in the network b) intrusion in the network (DoS attack) but no means of defending these attacks, and c) intrusion (DoS attacks) with AIDP in place to protect the network . As can be seen from the graphs AIDP reduces the control packet overhead & conversely increases the network throughput when it is used in a network under attack by intruder causing DoS. However, the control packet overhead is not as low as that of a network when there is no intrusion because AIDP also needs control packets for IDP in the network.

CONCLUSION & FUTURE WORK

The on-demand nature of MANET routing protocols makes them susceptible to DoS attacks, such as sleep deprivation and rushing attack. In this paper we have illustrated how intruders can cause DoS attacks in MANETs. We consider the suitability of using only control chart to protect against these attacks, and demonstrated that this method based on static threshold similar that proposed in [8] & [9] is not suitable because it does not cope well with the dynamics of MANETs. We then proposed an adaptive intrusion detection & prevention mechanism AIDP. It employs ABID which first use chi-square test to check the overall behaviour of the network and indicate intrusion, and then uses control chart to identify intruding nodes. Finally we isolate the intruding nodes.

Simulation results show that AIDP successfully detects identifies & isolates the intruding nodes attempting to cause DoS attacks. AIDP exhibits a high success rate and very low false alarm rate with an affordable processing overhead on the network over a range of scenarios tested.

In our ongoing work we are focusing on generalizing AIDP by including other related parameters to cover all routing attacks in MANET.

6. REFERENCES

[1] Z.Li, A.Das, and J.Zhou, "Theoretical Basis for Intrusion Detection", IEEE *Proc, Information Assurance and Security,* pp 184-192, 15-17 June 2005.

[2] Leonard.A.Doty, " Statistical Process Control ", 2nd ed, 1996.

[3] H.O.Lancaster, "The Chi-Squared Distribution", Willey Publications in Statistics 1969.

[4] Y.Hu, B.Johnson and A.Perrig, " SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks", *Ad hoc Networks*, Vol.1, pp 175-192, 2003.

[5] K.Sanzgiri and M.Belding-Royer " A Secure Routing Protocol for Ad Hoc networks", *Proc, 10th IEEE ICNP, 2002*.

[6] Y.Hu, A.Perrig and B.Johnson, "A Secure On Demand Routing Protocol for Ad-Hoc networks",*Proc, MobiCom,* pp 23-28, September 2002.

[7] W.Wang, Y.Lu and K.Bhargava," On Vulnerability and Protection of Ad Hoc On Demand Distance Vector Protocol", IEEE *Proc, ICT,* Vol.1, pp 357-382, 2003.

[8] P.Yi, Z.Dai and S.Zhang," Resisting Flooding Attack in Ad Hoc Networks",*IEEE Conference on Coding & Computing,* April 2005.

[9] W.Yu and K.Ray," Defence against Injecting Traffic Attack in Cooperative Ad Hoc networks", *IEEE Global Telecommunication Conference,Globecom,* 2005.

[10] Y.Zhang and W.Lee," Intrusion Detection in Wireless Ad-Hoc Networks", *Proc, 6th, ACM MOBICOM,* 2000.

[11] D.Nguyen, Das, Memik and Choudhary," A Reconfigurable Architecture for Network Intrusion Detection Using Principal Component Analysis", *ACM Int Symposium on Field Programmable Gate Arrays,* 2006.

[12] N.Ye, X.Li, ,M.Emran and M.Xu," Probabilistic Techniques for Intrusion Detection Based on Computer Audit Data", *IEEE Transaction on Systems, Man & Cybernetics*, 2001.

[13] N.Ye and Q.Chen," An Anomaly Detection Techniques based on a CHI-SQUARE Statistics for Detecting Intrusion into Information System" *Quality and Reliability Engineering International,* 2001.

[14] M.Pirrete and R.Brooks," The Sleep Deprivation Attack in Sensor Networks: Analysis and Methods of Defence", *Int Journal of Distributed Sensor Networks,* Vol.2, No.3, pp 267-287, 2006.

[15] Y.Hu, A.Perrig and B.Johnson," Rushing Attack and Defence in Wireless Ad Hoc Networks Routing Protocol", *2nd ACM Workshop on Wireless Security*, pp 30-40, 2003.

[16] C.Perkins," Ad Hoc On Demand Distance Vector (AODV) Routing", RFC 3561, July 2003.

[17] KDD data set, 1999, can be accessed at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.