An Intrusion Detection & Adaptive Response Mechanism for MANETs

Adnan Nadeem[a, b] and Michael P. Howarth[b]

[a] *Department of Computer Science, Federal Urdu University of Arts Science & Technology, Pakistan.* [b] *Centre for Communication Systems Research, Department of Electronic Engineering, University of Surrey, UK*

Abstract

Mobile Ad Hoc Networks are vulnerable to a variety of network layer attacks such as black hole, grey hole, sleep deprivation & rushing attacks. In this paper we present an intrusion detection & adaptive response mechanism for MANETs that detects a range of attacks and provides an effective response with low network degradation. We consider the deficiencies of a fixed response to an intrusion; and we overcome these deficiencies with a flexible response scheme that depends on the measured confidence in the attack, the severity of attack and the degradation in network performance. We present results from an implementation of the response scheme that has three intrusion response actions. Simulation results show the effectiveness of the proposed detection and adaptive response mechanisms in various attack scenarios. An analysis of the impact of our proposed scheme shows that it allows a flexible approach to management of threats and demonstrates improved network performance with a low network overhead.

1. Introduction

Intrusion detection and prevention provides a way to protect mobile ad hoc networks (MANETs) from attacks by external or internal intruders. There are two principal intrusion detection techniques: anomaly based intrusion detection (ABID) and knowledge based intrusion detection (KBID); additionally, specification based intrusion detection techniques (SBID) have also been proposed recently. Once an intrusion has been detected by an intrusion detection system (IDS) it is desirable to take action to thwart attacks or mitigate the damage caused by the attack: this action is referred to as an intrusion response (IR). Although intrusion response is normally a part of the intrusion detection system, it has historically received less attention than the detection component of the IDS.

General protection approaches such as [10][11][14][15][17] do not consider attack responses at all, and some other proposed MANET IDSs, for example

[1][16][9], respond to intrusion in a predetermined fixed way by isolating or banning the detected intruder nodes. However, in some cases authors have focused on the intrusion response and presented new ways of responding to intrusion. For example, an agent-based cooperative response was proposed in [3]. A cost sensitive model for Intrusion Response Systems (IRS) in fixed networks was proposed in [12]. The authors of [13] used this concept to provide a cost sensitive intrusion response for MANETs. In MANETs it is difficult to calculate the intrusion response cost, which we can define as the negative impact on the network resources caused by the response. In [13] the authors first estimate a Topology Dependency Index (TDI) which indicates how much the routing service of nodes in the network will be disrupted if the intruder is isolated. Then they estimate the Attack Damage Index (ADI) that indicates the damage caused by an attack. The ADI calculates the damage in terms of the number of nodes that are affected by the attack. Finally, they respond to the intrusion by isolating the attacker if the ADI is greater than the TDI. This cost-sensitive model was proposed for the proactive routing protocol OLSR where complete network topology information is available for every node. However, this approach is not suitable for reactive routing protocols such as AODV & DSR because they only provide partial topology information; for example, in AODV a node only knows its next hops towards the source or destination of active paths.

In this paper, we present an intrusion detection & adaptive response mechanism (IDAR) that employs a combination of both anomaly based and knowledge based intrusion detection techniques, and takes advantage of both techniques to protect MANETs against a variety of attacks. We consider our previously proposed algorithm [1] that responds to intrusion in all cases by isolating the intruding nodes in a predetermined fixed way. We investigate the impact on a MANET's performance of (a) various attacks and (b) the fixed intrusion response (isolation) of our previous algorithm [1]. The results of this investigation enable us to identify the deficiencies of the fixed response approach. To overcome these deficiencies, in this paper we now propose an adaptive flexible intrusion response scheme. This new scheme selects the intrusion response action based on the severity of the attack, the degradation in network performance and the expected impact of the response action on the network performance. Further, to improve scalability and to estimate the overhead imposed by our scheme we borrow and implement the clustering approach proposed in [2], modifying it for our protection mechanism. Finally, we have conducted a case study to assess the overall effectiveness of the proposed detection and adaptive response scheme.

The rest of the paper is organized as follows. Section 2 describes the architecture and core functionality of the training, intrusion detection, attack identification and intruder identification phases of IDAR. In Section 3 we present in detail the flexible adaptive intrusion response scheme. Section 4 presents the assessment of IDAR and evaluation of the effectiveness of the scheme through a case study, including extensive simulations. Section 5 presents our conclusions.

2. Proposed Protection Mechanism

In this Section, we describe the architecture and core functions of IDAR. For completeness and clarity we first briefly review the data collection, training, intrusion detection, attack identification and intrusion identification phases in this section (the algorithm and technical details of these phases were originally described in [1]). We then present in detail the adaptive flexible intrusion response scheme in Section 3.

2.1. Key Assumptions

Although both physical and data link layer operations are vulnerable to attacks we do not explicitly consider attacks in these layers in this paper. However, we believe that with appropriate matrix selection, IDAR can also be applied effectively for these layers.

We note that ABID requires audit data traces and traffic patterns of normal events to build a training profile that is then used to detect anomalies in the network. However, in contrast with fixed networks, data resources such as [18] that reflect normal activities or events are not currently available for MANETs. Therefore, we assume that the initial behaviour of the network is free from anomalies.

To improve the scalability of IDAR we use a clustered MANET organization, in which all network nodes operate in one of the three roles of manager node (MN), cluster heads (CH) and cluster nodes (CNs). We further assume a security mechanism [4] to protect communication between MN, CHs and CNs. The details of these mechanisms are outside the scope of the work in this paper, because we focus on the intrusion detection & response mechanism.

2.2. Architecture & Core Functionality

In this subsection, we describe the architecture and core functionality of IDAR. Fig.1 (a) illustrates the simplified architecture. It shows that IDAR operates in three main stages: network monitoring & data collection, training, and testing, and we now describe these stages.

2.2.1. Network Monitoring and Data Collection

IDAR monitors the network and periodically collects data for intrusion detection and prevention throughout the network's lifetime. In the data collection phase, after each time interval (TI) the CHs gather data from the CNs within their virtual cluster. The data is stored in the form of two matrices: the network characteristic matrix (NCM) and a performance matrix (PM). The CHs then report these matrices to the MN. The NCM records data that is specific to the network routing protocol. However, IDAR is general, and different NCM parameters can be used for different routing protocols. In this paper we illustrate IDAR using AODV as the routing protocol, and the NCM consists of the following seven parameters:

NCM= { *RREP (route reply), RREQ (route request), RERR (route error), TTL (time to live) values, RREQ src_ seq,, RREP dest_ seq RREQ dest_ seq* }

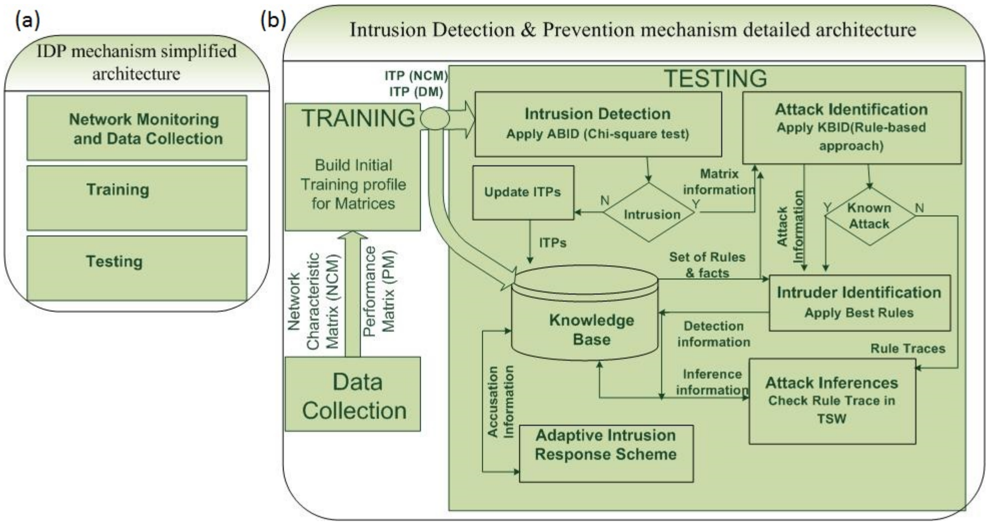


Figure 1: Intrusion detection & response mechanism architecture: (a) simplified, left; (b) detailed, right.

The performance matrix consists of parameters which reflect the network performance and which can be derived from NCM parameters. Here, the PM consists of the following four parameters:

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

NCM is a two dimensional matrix of (r * c) and the number of rows (r) and number of columns (c) depend on its parameters; therefore its storage structure is dynamically assigned by the IDAR monitor. In the PM, the routing packet overhead (RPO) is the ratio of the number of routing packets sent as a fraction of the number of data packets delivered to their destination nodes in the network. The packet delivery ratio (PDR) is the ratio of the number of data packets received at the destination nodes to the number of data packets originated by source nodes. The number of control packets dropped (CPD) is the number of routing packets dropped during the routing process, including route discoveries and route maintenance procedures in the network. The last parameter of the performance matrix, throughput, represents the average network throughput.

2.2.2. Training

In the training phase, CHs continuously gather NCM and PM information, and at fixed time intervals report their collected data to the MN. The MN applies the training module for N for these time intervals. The NCM consists of j parameters, where j=1 to 7 in the case study in this paper. ${}_{j}X_{k}^{i} = X_1, X_2, X_3, \ldots, X_M$ is a set of random variables representing the jth NCM parameter in the ith time interval and k= (1 to M) represents the number of random variables in the jth NCM parameter, where M is the maximum value of the random variables of the NCM's jth parameter in the ith time interval. Similarly, the performance matrix is represented by ${}_jY^i_k$ where j=1 to 4 in this paper's case study. The MN calculates the probability distribution of P(${}_jX^i_k$) for time interval i, and also calculates the PM parameters for the ith time interval. This whole process is repeated for the N time intervals. The MN then calculates the mean NCM of P(${}_jX^i_k$) and the mean PM for N intervals, and these are stored as an initial training profile (ITP) of the NCM and PM. These initial training profiles reflect the normal behaviour of the nodes in the network and the expected network performance.

2.2.3. Testing

Testing operates in four phases: a) intrusion detection; b) attack identification; c) intruder identification; and d) adaptive intrusion response, as shown in Fig.1(b). The first three phases are described in this sub section, while the adaptive intrusion response scheme is presented in detail in section 3.

Intrusion Detection. In the intrusion detection phase the MN considers the network characteristic parameters from the NCM, and uses ABID to identify any intrusion in the network. The ABID uses the chi-square test, because it has a low computational cost and is based on distance measure, as compared to other tests such as Hotelling's T^2. The algorithm first calculates the probability distribution of each NCM parameter, and stores these as observed values. For each time interval (TI) the MN performs hypothesis testing with null hypothesis *Ho[j]*(observed distribution of NCM fits the expected) for each parameter j of the NCM at calculated chi-computed values obtained from Equation 1, where j is the NCM parameter and k(=1 to M) is the number of random variables in each parameter. The MN then performs combined hypothesis testing of all parameters of the NCM.

$$\chi^2[j] = \forall_j(\sum_{k=1}^{M}(\frac{({}_jX^i_k - {}_j\bar{X}^i_k)^2}{{}_j\bar{X}^i_k}))\text{.....(1)}$$

If the combined null hypothesis *Ho* (observed distribution of all NCM parameters fits the expected) is rejected then it assumes intrusion has occured during the TI, and proceeds to the next stage i.e. attack identification. Else, we update the initial training profile of the NCM through an exponentially weighted moving average (EWMA):

$\forall_j(\overline{{}_jX^i_{(q,k_1^M)}} = \beta \star {}_jX^i_{(q,k_1^M)} + (1-\beta) \star \overline{{}_jX^i_{(q,k_1^M)}})$......(2) , where $\overline{{}_jX^i_{(q,k_1^M)}}$ and ${}_jX^i_{(q,k_1^M)}$ represent the expected and observed values of NCM parameter j 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 NCM parameter and $\beta = \frac{2}{(q-1)}$ is the weighting factor. The updated expected profile model therefore reflects the current behaviour of the network.

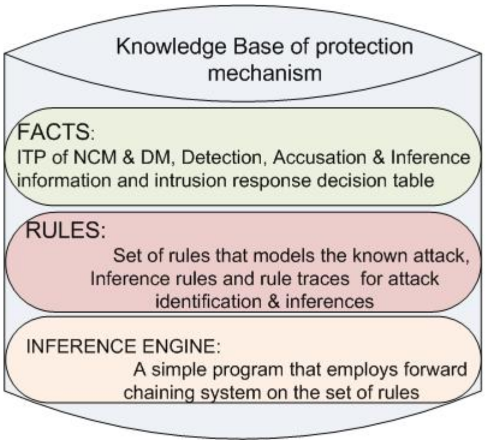

Knowledge Base of protection mechanism
FACTS:
ITP of NCM & DM, Detection, Accusation & Inference information and intrusion response decision table
RULES:
Set of rules that models the known attack, Inference rules and rule traces for attack identification & inferences
INFERENCE ENGINE:
A simple program that employs forward chaining system on the set of rules

Figure 2: Knowledge Base of IDAR

Attack Identification. If network intrusion is detected, the MN proceeds to the second stage, namely attack identification. This uses a rule-based approach to identify the attack that is taking place. IDAR maintains a knowledge base (KB) that is used in all stages of the testing phase. The knowledge base consists of facts, rules and an inference engine as shown in Fig.2. We have constructed a set of rules for attack and intruder identification by analyzing the existing literature of known attacks, for example [5][6][7][8], and through investigating various attacks including their impact on network performance. The KB inference engine employs forward chaining on the set of rules and looks for the goal condition fulfillment that indicates a known attack.

Intruder Identification . Once an attack has been identified, the MN initiates intruder identification. In this phase, the MN applies intruder identification rules that are specific to the the known attack. For example in case of a black hole attack it analyzes the RREP messages received from all the nodes during the latest TI and finds the node that has initiated the false RREP packet with the highest destination sequence number.

Following intruder identification, an IDS should ideally respond to the intrusion. In our original work [1] we employed a fixed intrusion response, in which the intruding node was in all cases isolated. However, as we shall see, this has deficiencies and therefore, to improve the overall effectiveness of the protection mechanism we have introduced an adaptive flexible intrusion response scheme, described in the next section.

3. Adaptive Intrusion Response Scheme

We now present the new adaptive flexible intrusion response scheme. We first describe the response model's internal architecture. We then illustrate a set of possible intrusion response actions suitable for MANETs, three of which are used

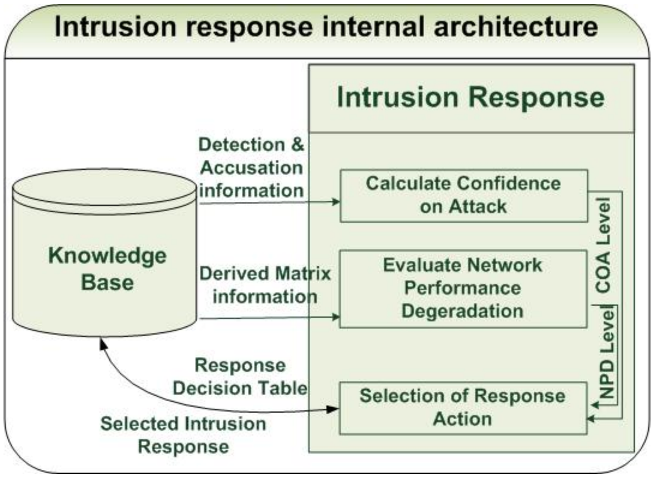

Intrusion response internal architecture
Intrusion Response
Knowledge Base
Detection & Accusation information
Derived Matrix information
Response Decision Table
Selected Intrusion Response
Calculate Confidence on Attack
Evaluate Network Performance Degeradation
Selection of Response Action
COA Level
NPD Level

Figure 3: Adaptive intrusion response scheme

in the case study described in Section 4. We also present the technical details of the adaptive intrusion response scheme. Finally, we give a time complexity analysis of this proposed scheme.

3.1. Adaptive Intrusion Response Scheme Architecture

Fig. 3 shows the architecture of the adaptive intrusion response scheme. Following a successful intruder detection, attack identification, and intruder identification, the intrusion response is implemented by the manager node, which primarily performs three tasks as shown in Fig. 3. In the first task the MN calculates the confidence level of the attack that has been detected, using the detection information. The MN then evaluates the network performance degradation (NPD) since the attack was launched using information in the performance matrix; this gives a measure of the severity of the attack. Finally a response action is selected and the necessary actions required to implement the intrusion response action (IRA) are taken. The selection of the response action requires the network administrator to model in advance the response decision process in the form of a decision table. This defines the criteria for selection of the IRA, taking into account the appropriateness of the IRA in the current context by considering the level of confidence in the detected intrusion and the degree of NPD.

3.2. Intrusion Response Action

Most of the IDSs in the literature respond to an intrusion in a predetermined fixed manner without considering the negative impact of the response or the side effects of the IRA on the network. To enhance the effectiveness of the intrusion response and to reduce its adverse effects on the network, we first consider possible IRAs (i.e. a range of punishments suitable for the intruding node) that are appropriate for MANETs. Then after analyzing the appropriateness of each IRA, we select a set of IRAs for the IDAR case study presented in Section 4.

3.2.1. List of Intrusion Response Actions

An example list of possible IRAs based on the various operations each network node performs on data and routing packets is as follows:

Isolation. In this response action all nodes in the network punish the intruding node by completely isolating it from the network immediately, that is, simply treat the intruder as non-existent. To employ this IRA, nodes impose the following restriction in terms of data forwarding and routing service.

- Network nodes do not forward any data packets originating from or destined to the intruding node.
- Network nodes do not route any data packets through the intruder.
- Network nodes do not send any routing packets to or through the intruder.
- Network nodes ignore all routing packets originating from the intruding node.

Probabilistic Isolation. In this IRA, nodes do not isolate the intruder completely; instead they apply some restriction in terms of forwarding its data. Specifically, nodes perform the following actions:

- Network nodes only forward some of the intruding node's data packets, with a specified probability.
- Network nodes do not send any routing packets through the intruder.

This ensures the intruder is not able to initiate further routing attacks, but is still able to forward data packets for other nodes in the network.

Route Around Attacker. In this IRA, nodes route data packets around the intruding node to stop further attacks from the intruding node while still allowing the intruder to forward data packets for other nodes. To employ this intrusion response nodes perform the following actions:

- Allow the intruder to forward data packets for other nodes in the network for existing routes. Nodes process these data packets so that they will reach their destinations.
- Do not include the intruder in new route discoveries, i.e. route the packets around the intruding node.
- Ignore all routing packets generated and forwarded by intruder (i.e. to prevent further attacks).

Thus, although packets will initially continue to be routed through the intruder, eventually the routes will expire as the network topology changes and eventually the intruder will not be included in any paths in the network.

Service Denial. In this response, network nodes deny services provided to or offered by the intruder while using the intruder as an intermediate router. For this intrusion response nodes perform the following tasks:

- Network nodes do not forward any data packets originating from or destined to the intruding node.
- Network nodes ignore any further services the intruder provides to other nodes in the network, for example providing internet access.
- Allow data packets to be routed through intruder nodes i.e. still use the intruder as an intermediate router in the network.

No Punishment. In some cases when the attack is not severe, i.e. the performance of the network is not significantly affected, it is possible that implementing any intrusion response will cause a worse degradation of the network performance than simply ignoring the attack. In these cases, the attack is simply ignored.

Relocation. Another response action is to physically move a node so that it is closer to the intruder node before isolating the intruder. This approach requires the availability of network topology information to find critical nodes in the network, and also requires the network to be able to command its nodes to move as required. For example, if isolating the intruder causes network partitioning due to its location in the network then a different node can be relocated close to the intruder node first to maintain the network connectivity, and then the intruder can be isolated from the network.

3.2.2. Proposed Intrusion Response Actions

We consider the appropriateness of each response action in the above list of possible IRAs in terms of their side effects or any adverse impact they might have on network performance. In addition, we further analyze the appropriateness of these response actions in terms of their practical effectiveness in combating attack, mitigating damage cause by attack and stopping further attacks from the intruding node. We then propose three IRAs for our response scheme and case study based on confidence on detected attacks and the impact of the attacks on network performance. This selected set of IRAs is as follows:

Isolation. This response action is used when the confidence in a detected attack is high, and the attack is severe, and the network performance has degraded considerably since the attack was launched. By isolating the intruder, nodes in the network will treat the intruder as non-existent. Although this will cause a rerouting overhead it still improves the overall network performance significantly.

Route Around Attacker. When the confidence in the detected attack is reasonably high and the NPD is noticeable then the response scheme will employ Route Around Attacker. This stops further attacks from the intruder while still maintaining the data forwarding service in the network.

No Punishment. When the COA is not high or the attack is not severe and NPD is tolerable then our response scheme will simply ignore the attack. This avoids reasonable adverse effects on the network performance.

3.3. Technical Details

We now consider in detail the functional of each process involved in the adaptive flexible intrusion response scheme. We observe that given the probabilistic nature of intrusion detection an intrusion response based on a single detection of an intruding node is not sufficient. Consequently, to optimise the probability of identifying intruders correctly (i.e. with a low level of false positives), the MN maintains a test sliding window (TSW). IDAR will therefore respond to the intrusion only when the intruding node has been identified in a number of time intervals (TIs). Specifically, an intrusion response only occurs if a given intruder node is identified in at least d detections out of p TIs of the TSW. To select the appropriate values of p (representing the size of the TSW in units of TIs, i.e. the number of checks considered) and d (the minimum number of detections required to confirm a detected node as an attacker), we note that the detection of an intruding node within a TSW is a Bernoulli trial (i.e. the trials during the TSW are identical and independent repetitions of the experiment with two possible outcomes: detection or no detection). The probability of confirmation of intrusion in a sequence of Bernoulli trials is therefore given by

$$P_c = \sum_{i=d}^{p} C_i^p \star (P)^i \star (1-p)^{(p-i)} ......(3)$$

In Equation 3, P represents a probability of a single detection, P_c is the probability of at least d detections in a TSW of size p, thereby confirming a node as an intruder, and $C_i^p = \frac{p!}{i!(p-1)!}$ is the binomial coefficient. The curves in Fig. 4 show how the probability of a node being confirmed as an intruder varies as a function of the number of detections required, d, with different values of probability of a single detection, P, for a TSW of size p=5. The graph shows that the probability of identifying a node as an intruder is over 90% when P=80% and d=1, 2 or 3. However, it is also important to avoid false positives: for example, if P=20% represents the probability of wrongly identifying a node that is in fact correctly behaving, then the probability of a false positive is below 10% when d=3, 4 or 5. In this example, d=3 provides a balance between correctly identifying an intruding node and avoiding a false positive.

The MN runs the adaptive intrusion response scheme for all nodes that have been identified as intruders in the current test sliding window, using Algorithm 1. The MN first estimates the confidence on attack detected (COA) value, based on the detection and accusation information:

$$COA = w_1 \star CI + w_2 \star P_c....(4)$$

In Equation 4, Wi represents a weighting factor, where the sum of these weights equals one. CI represents the confidence interval of the chi-square test

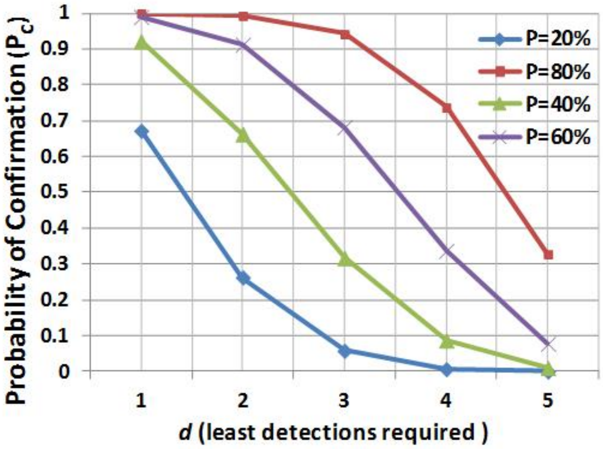


Figure 4: Probability of confirmation (Pc) as a function of d (least number of detections required) with varying probability of single detection P (p=5).

during the intrusion detection phase and P_c is the probability of confirmation. Equation 3 returns a confidence value for P_c between 0 and 1.

The MN then evaluates the NPD value using equation 5. This is a weighted sum of the changes in the performance matrix parameter values (i.e. throughput, packet delivery ratio, routing protocol overhead and routing packets dropped) from when there was no attack in the network to their current values, as follows:

$$NPD = w_1 \star \triangle Throughput + w_2 \star \triangle PDR + w_3 \star \triangle RPO + w_4 \star \triangle RPD.......(5)$$

where Δ represents the percentage change in the parameter between the average value in the current test sliding window and the average value of the parameter when there was no attack in the network. Once the COA and NPD values have been calculated, the MN assigns confidence levels to the COA and NPD. For illustration in this paper we define four COA levels, as shown in Table 1, namely low, medium, high or very high. For the NPD we again use four levels, but the precise mapping of NPD value to NPD level varies as will be seen in Section 4. These levels are then used in the decision table, Figure 5, (from the knowledge base constructed by the network administrator) to select the intrusion response. Modelling the intrusion response selection through a decision table allows the network administrator to configure and modify the intrusion response selection process for different network environments.

Fig. 5 shows the decision table used in our case study. The first two rows represent the conditions (i.e. COA and NPD levels) and the last three rows represent the actions (i.e. isolation, route around attacker and no punishment). If the selected intrusion response is isolation or route around attacker then the MN informs all nodes of the IRA by broadcasting an accusation packet (AP). In the case of no punishment the MN ignores the attack. When a CN receives an AP it first checks the broadcast id and source address of the packet to avoid

Table 1: Mapping ranges of COA values to COA levels

COA Level	Range of COA values (%)
Low	$0 <$ COA value (%) ≤ 25
Medium	$25 <$ COA value (%) ≤ 50
High	$50 <$ COA value (%) ≤ 70
Very High	COA value (%) > 70

Algorithm 1 Intrusion Response Mechanism

```
For all detected and identified nodes $Vi$ in a Test Sliding Window

  • Calculate Confidence On Attack (COA) value using Equation 4.

  • Calculate Network Performance Degradation (NPD) value using Equation
    5.

  • Assign COA level based on calculated COA value (Table 1).

  • Assign NPD level based on calculated NPD value.

  • Search Decision Table (Figure 5) using COA & NPD levels and identify
    Intrusion Response Action (IRA)

  • If (IRA== ISOLATION)

      MN blacklists $Vi$ & broadcasts Accusation Packet (AP)
      with IRA=ISOLATION
    Else    If (IRA== ROUTE_AROUND_ATTACKER)
              MN temporarily blacklists $Vi$ & broadcasts AP
              with IRA=ROUTE_AROUND
             Else: MN sets IRA to no punishment
             EndIf
     EndIf
EndFor
```

Algorithm 2 Accusation Packet Handling

```
• Each CN $Vi$ maintains its local BlacklistTable (BLT) & Temporary Blacklist (TBLT).
• If CN $Vi$ receives an AP for CN $Vj$.
• If CN $Vi$ has node $Vj$ in its BLT or TBLT then Ignore AP
      Else: CN $Vi$ checks IRA in AP.
        If (IRA== ISOLATION)
          CN adds node $Vj$ to its BLT & rebroadcasts AP.
          CN isolates intruding node $Vj$ (Algorithm 3(a))
      Else CN adds node $Vj$ to its TBLT & rebroadcasts AP.
              CN routes around intruder $Vj$ (Algorithm 3(b))
        EndIf
  EndIf
EndIf
```

processing duplicate APs. If the accused node Vj is already blacklisted either permanently or temporarily then the CN will ignore and drop the AP to prevent unnecessary network traffic (Algorithm 2). Otherwise, the CN will check the IRA specified in the AP for the node Vj. In the case of isolation, the CN will first add intruder Vj to its blacklist table, and then to isolate the intruder from the network all nodes will drop all packets from blacklisted node and also immediately ignore all packets in the queue that are from the blacklisted nodes as shown in (Algorithm 3(a)). If the IRA is route around attacker then the CN will first add the intruder Vj to its temporary blacklist table. To implement this IRA, all nodes will ignore and drop routing packets i.e. RREQ, RREP, & RERR packets generated or forwarded by the intruder node Vj to prevent further attacks from the intruder. All nodes exclude intruder Vj in their new route discoveries, i.e. they select routes that do not include Vj in the path. However, nodes will still forward the data packets received from Vj for existing routes to maintain the current level of data forwarding service (Algorithm 3(b)). Allowing a data-forwarding service from Vj reduces the possibility of adverse affects on network performance for a period, until the nodes find new routes around Vj. This response action is also effective when node Vj is a critical node in terms of its location in the network topology, when isolating node Vj could have significant side effects on network performance.

3.4. Time Complexity Analysis of Adaptive Intrusion Response Scheme

We now consider the complexity of the adaptive intrusion response mechanism. We assume that a single non-iterative task takes t seconds to complete. The intrusion response module is called for each node j that has been detected as an intruder in a TSW and we further assume that the proposed mechanism consists of n TSWs. The intrusion response phase can be further divided into

Algorithm 3 Intrusion Response Action

```
a) Isolate Intruding Node
• If node $Vi$ receives packet from node $Vj$
•     If node $Vj$ is in node $Vi$ BLT ignore all packets & drop all packets
      queued from $Vj$
• Else: handle & process packet
•     EndIf
• EndIf
b) Route Around Attacker
• If node $Vj$ is in node $Vi$ TBLT
•     If node $Vi$ receives routing packet from node $Vj$
•         Ignore & drop RREQ, RREP & RERR packets from $Vj$
•     EndIf
•     If node $Vi$ receives data packet from node $Vj$ destined for node $Vk$.
•         Node $Vi$ forwards data packets towards node $Vk$
•     EndIf
•     Node $Vj$ removes route entries including node $Vj$ from its route table
•   Else:
•   Handle & process packet
•   EndIf
• EndIf
```

four tasks:

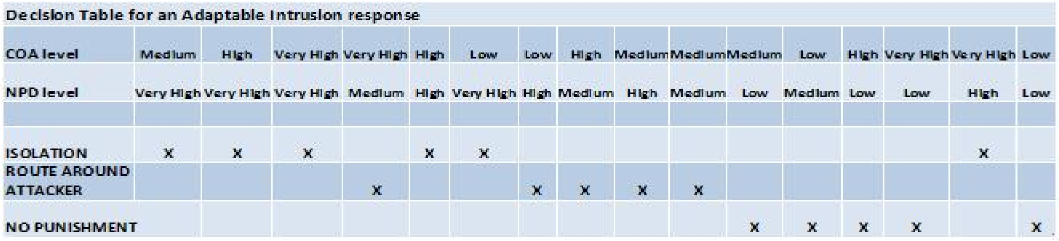
1. Calculating the COA for each node j (Eq. 4). This involves calculating CI and P_c, and so takes time $2 * j * t$ seconds.
2. The intrusion response module then evaluates the NPD for each detected node$j$ (Eq. 5), which is based on M performance matrix parameters. This takes time $j * M * t$ seconds.
3. The MN searches its decision table (i.e. a decision tree) with the calculated COA & NPD values, to identify the intrusion response action. The complexity of building a decision tree is *2d-1,* where d is the depth of the decision tree. We assume as a worst case that the complexity of searching a decision tree is the same as building it, so the time it takes to build and search the decision tree for the intrusion response will be $(2d -1) * 2t$ seconds.
4. The MN compares the intrusion response selected from the decision table and performs this intrusion response action. The comparison will take t sec. Let us assume the selected response is isolation, and that this includes blacklisting node j and broadcasting an AP; this will take $2 * j * t$ seconds.

Now combining tasks 1), 2) , 3) and 4) gives a running time complexity $= n\{ 2 * j *t +j * M * t + (2d -1) * 2 * t + t + 2* j * t\}$, which can be simplified to $n\{j(4+M)+4d-1\}t$. Now if we focus on the non-constant portion of the running time and ignore the constant then the above expression in big O notation will be $\boldsymbol{O(nj(M+d))}$. This expression suggests that the running time complexity of the intrusion response module depends on n (number of test sliding windows over which the module is applied), j (the number of nodes for which the intrusion response is required), M (the number of performance matrix parameters) and d (the depth of the decision tree used by the IR module).

4. Case Study and Evaluation

We have described the principles of IDAR in Sections 2 and 3, and we now present a case study that illustrates and assesses the proposed adaptive protection mechanism. We first evaluate the performance of the protection mechanism in different attack scenarios. In the second part of the case study, we investigate the IRA selected by IDAR under various attack scenarios and assess the scalability of the proposed mechanism. In the third part of the case study we consider the impact of IDAR on the network performance. We do this by evaluating three response approaches, namely a) the effectiveness of the mechanism with no intrusion response, b) the effectiveness of the mechanism with fixed intrusion response, and c) the effectiveness of our proposed mechanism with adaptive flexible response. We analyze the impact of the three response approaches on the routing protocol overhead (when using AODV) and the direct IDAR overhead. Finally in this Section, we compare IDAR with two other mechanisms described in the literature.

Our assessment of IDAR under various attack scenarios is based on simulations. We use GloMoSim version 2.03 to build the simulation environment,



Decision Table for an Adaptable Intrusion response

COA level	Medium	High	Very High	Very High	High	Low	Low	High	Medium	Medium	Medium	Low	High	Very High	Very High	Low
NPD level	Very High	Very High	Very High	Medium	High	Very High	High	Medium	High	Medium	Low	Medium	Low	Low	High	Low
ISOLATION	X	X	X		X	X									X	
ROUTE AROUND ATTACKER				X			X	X	X	X						
NO PUNISHMENT											X	X	X	X		X

Figure 5: Decision Table for an adaptive flexible intrusion response scheme

Table 2: Simulation Parameters

Number of Nodes	25	50	100	150	200
Terrain Dimensions (metres)	500*500	707 *707	1000*1000	1225*1225	1415*1415
Node placement	Distributed uniformly				
Routing protocol	AODV				
Simulation time	2000 seconds				
Simulation traffic	Constant Bit Rate				
MAC protocol	IEEE 802.11				
Nodes' mean speed	Varies from 0 to 20 m/s				

using the parameters shown in Table 2. IDAR is assessed using the configuration parameters shown in Table 3. More generally, these parameters can be changed to enable the mechanism to cope with different protection requirements. For example in order to adapt to a network where intrusion detection and response time is critical, the network administrator can adjust configuration parameters such as time interval and test sliding window size.

4.1. Evaluation of Proposed Mechanism: Attack Identification

In this sub-section, we assess IDAR in terms of its success rate (SR) and false alarm rate (FA) in different attack scenarios. By SR, we mean the percentage of intrusions that are correctly detected, and where both the attack type and the intruder node are correctly identified. FA means a properly behaving node that has been incorrectly identified as an intruder.

We first test IDAR's performance with a number of attack types. At each tested mean speed and for each network size (25, 50 or 100 nodes) we perform 40 runs with no intrusion and 40 runs with intruders. The graph in Fig. 6 depicts the SR and FA rate of IDAR as a function of the nodes' mean speed in 25, 50 & 100 node networks with a sleep deprivation (SD) attack by malicious RREQ flooding (i.e. a Denial of Service (DoS) attack). The figure shows good performance of IDAR in terms of high success rate and low FA rates against the SD attack. The graph in Fig. 7 illustrates the performance of IDAR wagainst black and grey hole attacks, and again shows good performance in terms of high success and low false alarm rate.

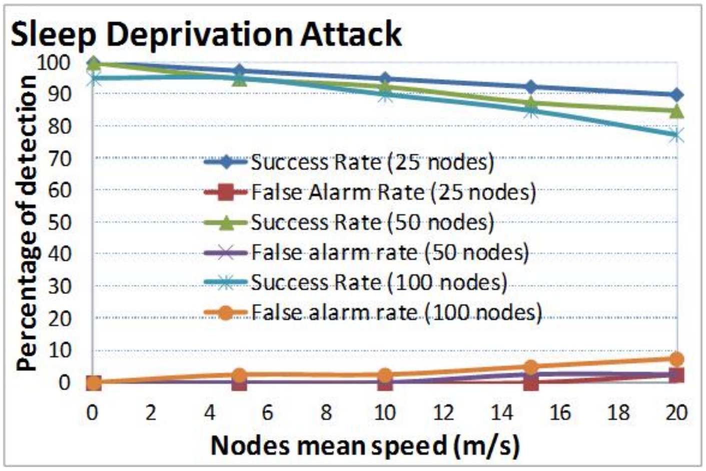


Figure 6: Success & false alarm rate of SD attack

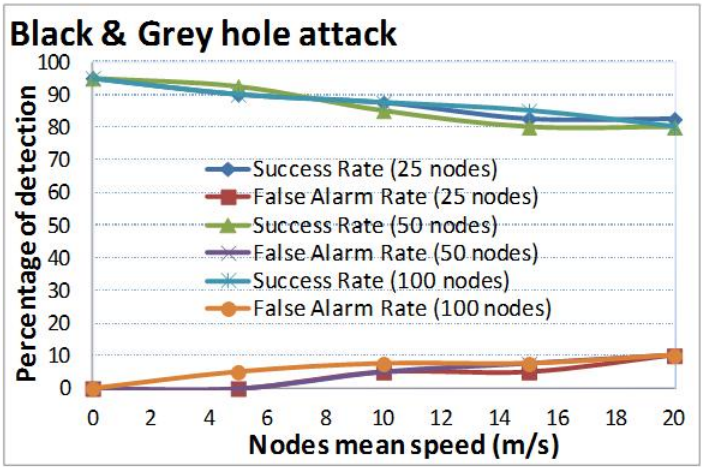


Figure 7: Success & false alarm rate of Black & Grey hole attack

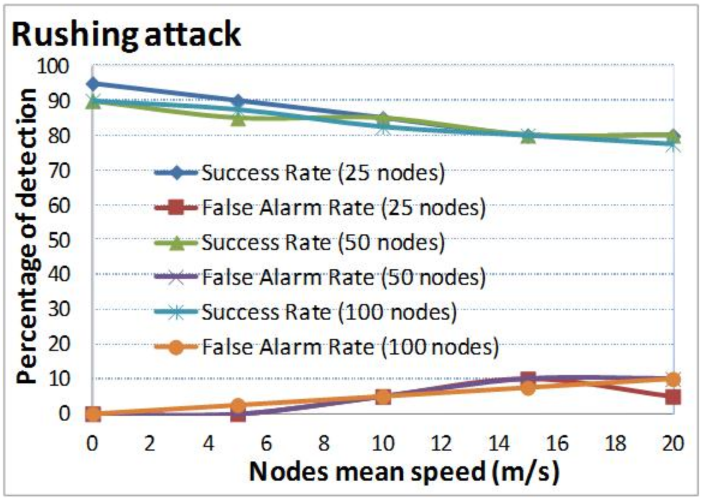


Figure 8: Success & false alarm rate of Rushing attack

Finally the graph in Fig. 8 shows the performance of IDAR under a rushing attack through forged RREQ packets, and again illustrates IDAR's good performance. In all three attacks considered, the performance drops slightly when the nodes are moving with high speed because with high mobility causes an increased frequency of link failure, and route discovery takes a longer time.

4.2. Evaluation of Intrusion Response Action

We now consider how the intrusion response selected by IDAR varies in different attack scenarios and analyze the appropriateness of the IRA. We assess IDAR's scalability by including experiments with 100, 150 & 200 node networks.

4.2.1. Black Hole Attack

In this scenario, we test IDAR when one randomly selected intruding node launches a BH attack. We perform 20 runs for each network size (25, 50, 100, 150 & 200 nodes), and use the simulation parameters, configuration parameters and COA mapping range shown in Tables 2, 3 and 1 respectively. The adaptive response scheme uses the decision table shown in Fig. 5 to determine the IRA selection.

The graph in Fig. 9 shows (expressed as a percentage) the fraction of total identified intrusions for which each IRA (i.e. isolating intruder, route around attacker or no punishment) was selected. The intrusion response scheme used the NPD level settings shown in the graph. The graph shows that for the BH attacks IDAR isolated the intruding node in most cases. It can be seen that in larger networks (i.e. more than 100 nodes in the network) IDAR responded to an intrusion by routing around the intruder node in (on average) 26% of cases and opted to ignore the attack on average 23% of the time. IDAR isolated the

Table 3: Configuration Parameters

Time interval (TI)	100 seconds
Training period (N)	5 TIs
Testing period	15 TIs
Number of intruders	Varying from 1 to 5
Chi-square test (α)	5% (i.e. 95% confidence interval)
Test Sliding Window (TSW)	5 TIs
Number of Parameters	PM=4 & NCM=7 parameters
Intrusion Response Actions	Complete Isolation, Route Around Attacker & No Punishment
Confidence on attack	Function of: Confidence Interval of
Network Performance Degradation	Function of: Network Throughput, PDR, RPO & RPD
COA & NPD levels	4 (Low, Medium, High & Very High)

intruder 90% of times on average in smaller networks (25 & 50 nodes) but this percentage drops to 53% in the larger networks.

To investigate the difference in intrusion response selection in the smaller and larger networks, we repeated the same set of experiments by first adjusting the NPD level settings and then slightly modifying the way an intruder launches a BH attack in a larger network. To launch a BH attack, an intruder first sends a forged RREP with an incremented destination sequence number (indicating freshness of route). This increment is f. We used f in the range $5 \leq f \leq 30$ for the smaller network sizes (25 & 50 nodes), and $15 \leq f \leq 40$ in the larger networks (100, 150 & 200 nodes), because a larger value of f allows an intruder to capture more routes by falsely claiming to nodes that it has fresher routes. The graph of Fig. 10 shows the results with the modified NPD level settings shown on the graph. The figure shows that the selected IRA does not vary much between smaller and larger networks: IDAR responds to the BH attack by isolating the intruder most of the time in larger networks as well as the smaller ones. We infer from Figs. 9 and 10 that an intruder has to adjust the way it launches the BH attack in MANETs to cause the same amount of damage in larger networks as in smaller networks, and that consequently the protection mechanism has to adopt in order to improve its performance in larger networks (scalability).

4.2.2. Sleep Deprivation Attack

In this scenario, we test our intrusion response scheme with a SD attack. Following the approach used in the black hole attack, we use the modified NPD level settings. The graph of Fig.11 shows that IDAR has responded to a SD attack by isolating the intruding node 78% of the time on average across all network sizes. In general it shows the appropriateness of the action taken by IDAR intrusion response scheme, because we know from the literature, and from our previous work [1], and by analyzing the impact of various attacks that SD is

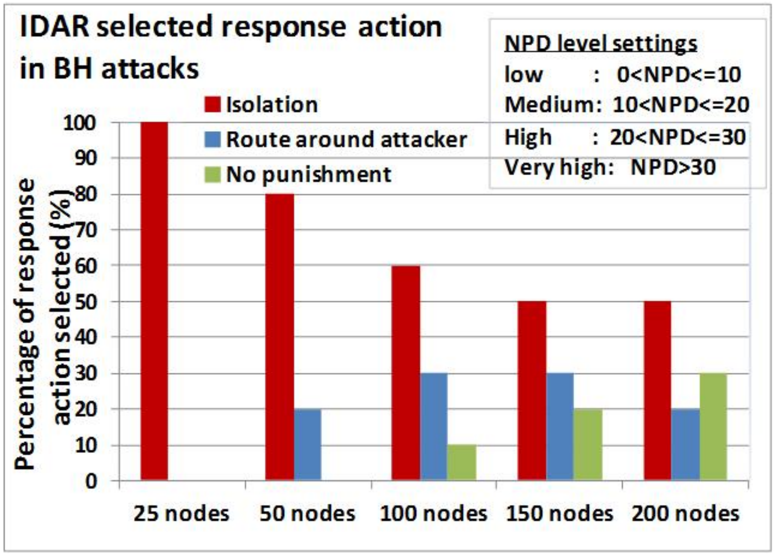


Figure 9: Intrusion response action selected in BH attack

a severe DoS attack that generally causes considerable damage to the network. It can also be seen from the graph that the proposed intrusion response scheme scales well for larger networks.

4.2.3. Rushing Attack

Our final evaluation of the intrusion response action considers a rushing attack. We believe rushing is a mild attack, and this is confirmed by tests where we have inspected the impact of various attacks on MANETs' performance. The graph of Fig.12 shows that in response to a rushing attack IDAR's response scheme has elected not to punish the intruding nodes most of the time, and this result is approximately independent of network size. This is because the impact of a rushing attack on network performance is generally low, and we have observed that taking strict action such as isolation when the attack is minor can actually result in a net degradation of network performance. The main reason for this degradation is that isolating a node that might perform an important routing function can affect a number of routes in the network, with rerouting causing disruption and degradation in network performance. This result shows the flexibility and effectiveness of our protection mechanism. However, it can be seen from the graph that in some cases the rushing attack significantly degrades the network performance, and in these cases IDAR has responded by either isolating or routing around the intruder node.

4.3. Impact of the Adaptive Flexible Intrusion Response Scheme

We now consider the impact of the intrusion response scheme on network performance. We initially estimate the effectiveness of the intrusion response

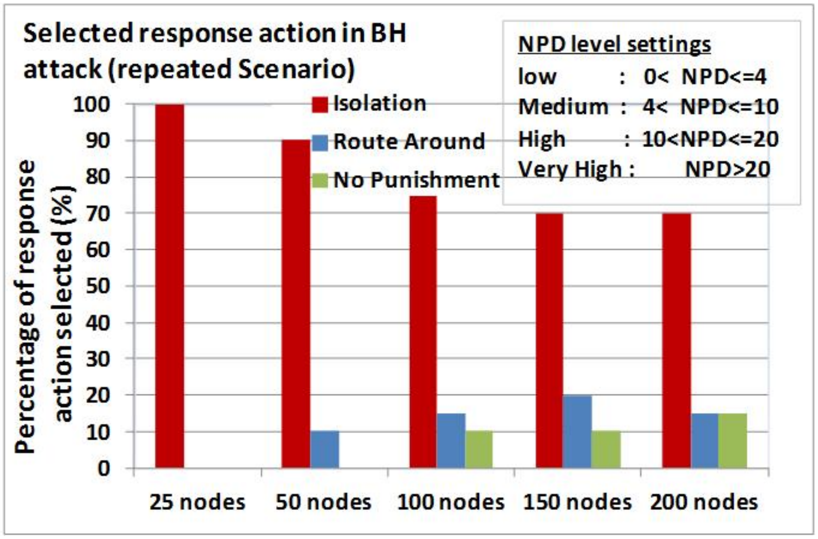


Figure 10: IRA selected in BH attack with modified NPD level settings

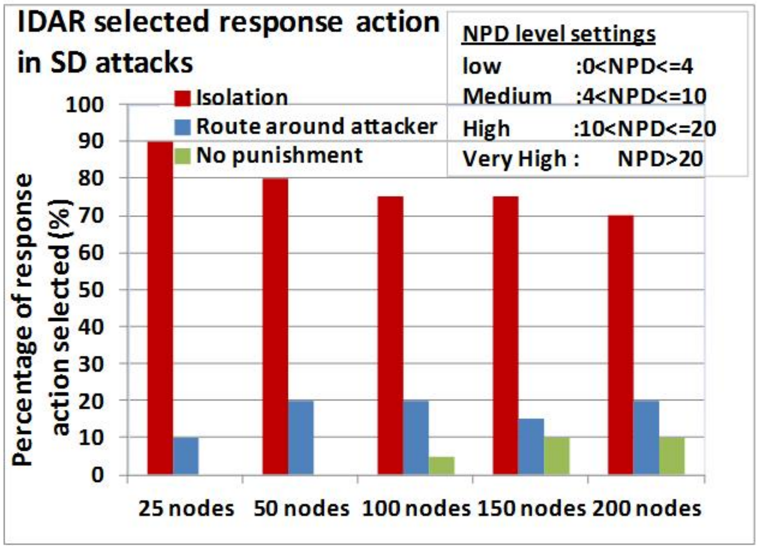


Figure 11: Intrusion response action selected in SD attack

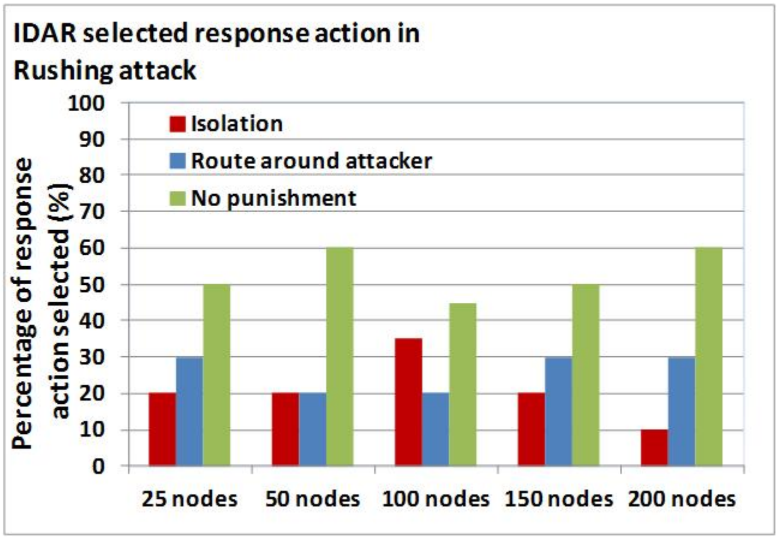


Figure 12: Intrusion response action selected in Rushing attack

scheme in terms of degradation in network performance when either a) no intrusion response, b) a fixed intrusion response, or c) an adaptive flexible intrusion response scheme is employed in the various attacks. We then analyze the impact of the IDAR response action on the AODV and IDAR overhead .

4.3.1. Impact of Intrusion Response Action on Network Performance

In this scenario, we analyze the effectiveness of the intrusion response scheme using NPD as a metric. We consider 25 and 50 node networks, both with various attacks and also with combinations of different simultaneous attacks. We performed 30 runs with each attack in a 25 node network, these being 10 runs when IDAR does not respond to intrusion, 10 runs when IDAR responds with a fixed response (isolate intruder) in all cases, and 10 runs when IDAR employs the adaptive flexible intrusion response scheme. We also repeated these tests with a 50 node network.

The graphs of Fig. 13 and 14 shows the effectiveness of the intrusion response scheme in terms of the NPD, for 25 and 50 node networks respectively. They show the NPD in various attack situations when there is no response to intrusion by IDAR, when the response is intruder isolation, and in the adaptive response case. It can be seen from the graphs that the average network degradation is minimised when IDAR is used with the adaptive flexible intrusion response scheme proposed in this paper. Although IDAR minimises the damage to network performance in all attacks, we observe that in the case of mild attacks such as rushing or some GH attacks, the adaptive response significantly reduces the network degradation.

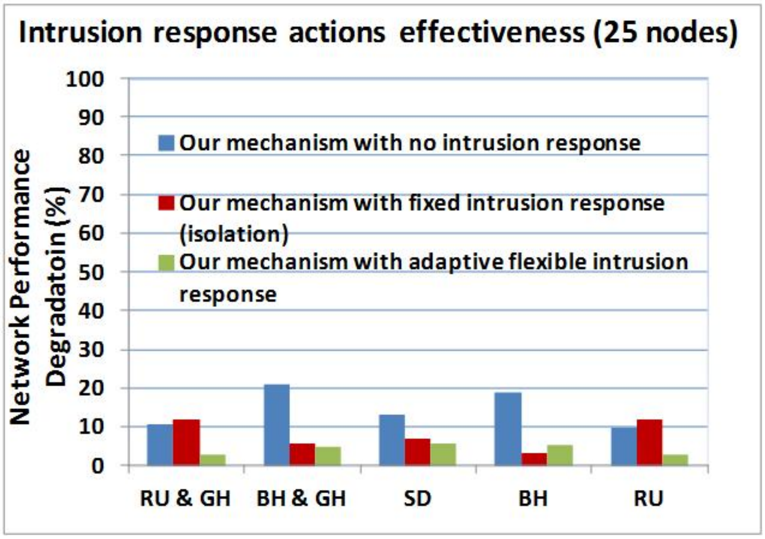


Figure 13: IRA effectiveness with various attacks in 25 node scenario.

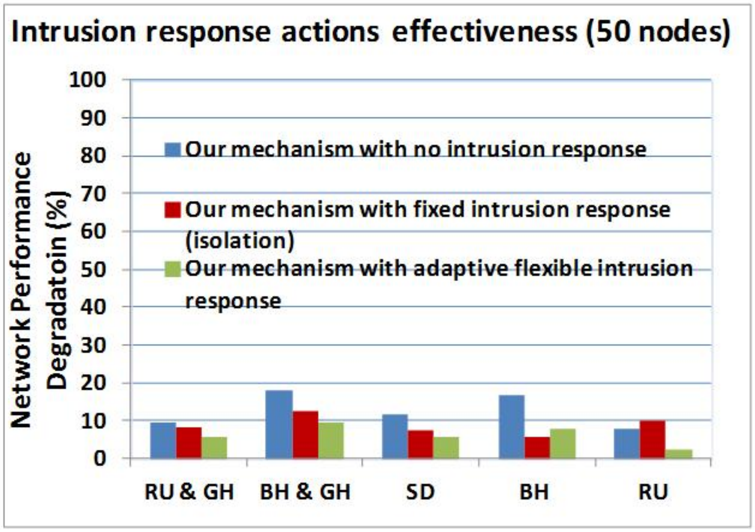


Figure 14: IRA effectiveness with various attacks in 50 node scenario.

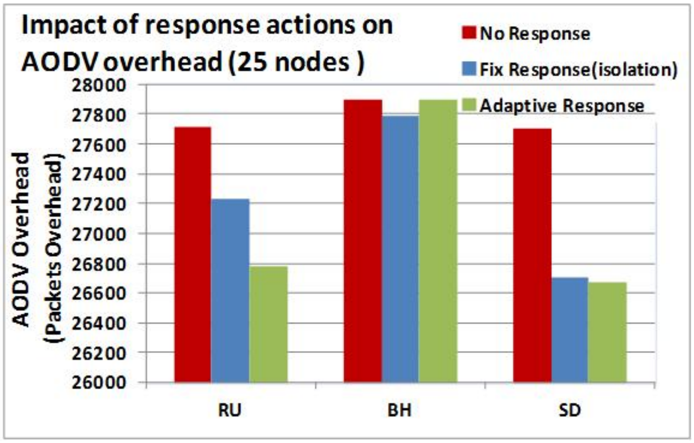


Figure 15: Impact of IRA on AODV overhead with BH, RU and SD attack in 25 node network

4.3.2. Impact of Intrusion Response Action on AODV & IDAR Overhead

Finally, we analyze the packet overhead imposed on the MANETs by the AODV routing protocol and our protection mechanism. We define the packet overhead as the number of packets generated multiplied by the number of hops each packet travels. We consider a 25 node network and analyze the AODV and IDAR overheads by performing 30 runs (10 runs each with no response to intrusion, fixed intrusion response, and adaptive intrusion response) with rushing, BH, and SD attacks.

Fig. 15 shows the impact of IRA on the AODV overhead when a) there is no response to intrusion, b) fixed response and c) adaptive intrusion response, in the cases of BH, SD, and rushing attacks. The AODV overhead comprises all control packets i.e. RREQ, RREP and RERR packets generated in the network during the simulation. The graph shows that as a result of employing the proposed intrusion response scheme the AODV overhead decreases by 6.8% & 6.4% in the cases of sleep deprivation & rushing attacks respectively. Fig. 16 shows the impact of IRA on the IDAR traffic overhead, including the overhead of the clustering approach used. This overhead comprises the network characteristic packets reported periodically from CNs to the CHs and the MN, cluster configuration packets and the AP generated by the MN and CHs to inform CNs of the required IRA. It can be seen from the graph that the IDAR overhead is least when there is no response to intrusion, simply because there are no APs in the network. The overhead is similar when either fixed or adaptive intrusion response is used; however, using adaptive intrusion response in case of rushing attack has the least overhead.

Analyzing the two graphs in Fig. 15 and Fig. 16 we can see that on average the AODV traffic is approximately 8 times the IDAR traffic. If we consider all the network traffic, i.e. AODV, IDAR and the UDP-based CBR traffic that the network is intended to carry, then the IDAR traffic, including clustering packets,

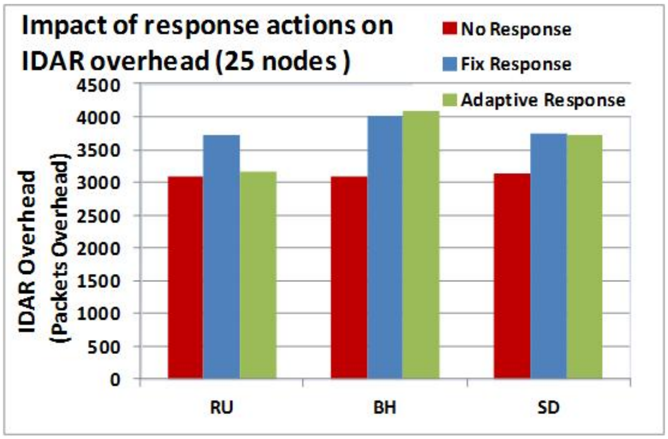


Figure 16: Impact of IRA on IDAR overhead on RU, BH and SD attack in 25 node network

contributes on average less than 5% of the total network traffic, a very low sum.

4.4. Comparison

We now compare the proposed IDAR mechanism with two existing techniques, [1] and [13]. In [1], we proposed a generalized intrusion detection and prevention mechanism, which responds to the detected intrusion by isolating the intruder in all cases (i.e. fixed response). Fig. 13 shows the degradation in the network performance with fixed intrusion response of isolation as proposed in [1], and compares it with IDAR's adaptive response as proposed in this paper. The graph indicates that with a fixed intrusion response the network performance degrades by an amount in a range from 2% to 12%. However, when adaptive intrusion response is selected, the NPD is in the range of 3% to 5%.

Table 4 compares the cost sensitive intrusion response model [13] with IDAR. It can be seen that in [13] the authors proposed the calculation of a Topology Dependency Index (TDI) and an Attack Damage Index (ADI). The TDI represents the routing dependency of nodes on the intruder and the ADI represents the damage caused by an attacker. However, IDAR relies on confidence in detected attack and its impact on network performance. The model in [13] operates using five types of intrusion response (normal, recovery, full isolation, temporary isolation and relocation). IDAR responds to the intrusion by adaptively selecting one of the three responses (isolation, route around attacker, and no punishment). Both the cost sensitive model [13] and IDAR have been implemented using GloMoSim. In general the comparison shows that IDAR is better in terms of improving network performance in various attacks with minimum overhead on the network.

Table 4: Comparison of Cost Sensitive Intrusion Response Model [13] and IDAR.

Comparing Parameter	**Cost Sensitive Model [13]**	**IDAR**
Intrusion response selection criteria	Topology Dependency Index (TDI) and Attack Damage Index (ADI)	Confidence on Attack (COA) and Network Performance Degradation (NPD)
Intrusion response actions	Normal, Recovery, Full isolation, Temporary isolation and Relocation	Isolation, Route around attacker, and No punishment
Types of attacks considered	Authenticity, Integrity and Availability	Black hole, Sleep deprivation, Rushing, Grey hole
Parameter for intrusion response impact assessment	Packet Delivery Ratio (PDR)	Network Performance Degradation (Eq.5)
Impact of intrusion response scheme	Max PDR reduction = 13 %	Maximum NPD = 7 %
Scalability	Simulated up to 50 nodes	Simulated up to 200 nodes
Network overhead	Not considered	Less than 5 % of total network traffic

5. Conclusion

IDAR can not only detect a number of attacks but can also adaptively respond to the detected attacks to halt the attack and / or mitigate the damage caused by the attack and prevent further attacks from the intruding nodes. Our intrusion response scheme has a reduced impact on network performance, and works by adaptively selecting the intrusion response action based on the level of confidence in the detection of the attack, the attack severity and the degradation in network performance. The use of a decision table to represent the intrusion response action selection criteria allows a flexible approach to management of threats and can accommodate the different security requirements of the network. IDAR demonstrates the importance of a flexible response that takes account of network conditions and attack type.

The first part of the case study shows the good performance of IDAR in terms of its high success and low false alarm rate in a range of attacks. The second part of the case study results reveal that each of proposed response actions are appropriate in different attack scenarios. For instance, in severe attacks such as BH and SD, IDAR can elect to completely isolate the intruding node (81% and 78% of the time respectively). On the other hand, in mild attacks such as rushing the mechanism generally opts not to punish the intruding node because the impact on the network of the attack is low and taking severe action (complete isolation) could result in an overall degradation in network performance. The last part of the case study analyzes the impact of proposed IRAs on network performance, and shows that our mechanism with adaptive flexible intrusion response is more efficient than either a fixed intrusion response or no response in all attack scenarios. We finally note that IDAR incurs a low overhead on the network, being less than 5% of the overall network traffic.

References

[1] A.Nadeem and M.Howarth, Protection of MANETs from a Range of Attacks using an Intrusion Detection and Prevention System, Telecommunications Systems Journal, Springer, Vol.52, No.4, pp.2047-2058, Apr. 2013.

[2] O.F.Gonzalez Duque, A.M.Hadjiantonis, G.Pavlou and M.Howarth, Adaptable misbehaviour detection and isolation in wireless ad hoc networks using policies, Proc. IFIP/IEEE International Symposium on Integrated Network Management (IM), 1-5 June 2009.

[3] Y.Ping, Z.Futai, J.Xianghao and L.Jianhua, Multi-agent Cooperative Intrusion Response in Mobile Ad Hoc Networks, Elsevier Journal of System Engineering and Electronics, Vol.18, No.4, pp.785-794, 2007.

[4] Y.Zhang, W.Liu, W.Lio and Y.Fang, Securing Mobile Ad Hoc Networks with Certificateless public keys, IEEE Transactions on Dependable and Secure Computing, Vol.3, No.4, pp.386-399, Oct.- Dec. 2006.

[5] S.Kurosawa and A.Jamalipour, Detecting Blackhole Attacks on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method, International Journal of Network Security, Vol.5, Nov. 2007.

[6] J.Sen, M.Chandra, S.G.Harihara, H.Reddy and P.Balamuralidhar, A Mechanism for Detection of Gray Hole Attack in Mobile Ad Hoc Network, Proc. IEEE ICICS 2007.

[7] P.Yi, Z.Dai and S.Zhang, Resisting Flooding Attack in Ad Hoc Networks, Proc. IEEE International Conference on Information Technology Coding & Computing (ITCC), April 2005.

[8] Y.Hu, A.Perrig and B.Johnson, Rushing Attack and Defense in Wireless Ad Hoc Networks Routing Protocols, Proc. 2nd ACM Workshop on Wireless Security, New York, 2003.

[9] Y.A.Mohamed and A.B.Abdullah, Implementation of IDS with Response for Securing MANETs, Proc. IEEE International Symposium in Information Technology (ITSim), Vol.2, pp.660-665, 2010.

[10] K.Sanzgiri and M.Belding-Royer, A Secure Routing Protocol for Ad Hoc networks, Proc. 10th IEEE International Conference on Network Protocols (ICNP '02), 2002.

[11] J.Joseph, A.Das, B.Seet and B.Lee, CRADS: Integrated Cross Layer approach for Detecting Routing Attacks in MANETs, Proc. IEEE Wireless Communication and Networking Conference (WCNC), 31st March - 3rd April 2008.

[12] N. Stakhanova, S. Basu and J. Wong, A Cost-Sensitive Model for Preemptive Intrusion Response Systems, Proc. IEEE 21st International Conference on Advance Networking and Applications (AINA), 2007.

[13] S. Wang, C.H. Tseng. K. Levitt and M. Bishop, Cost-Sensitive Intrusion Response for Mobile Ad Hoc Networks, Proc. 10th International Conference on Recent Advances in Intrusion Detection, 2007.

[14] F.Bu, F.R.Yu, P.Liu and H.Tang, Structural Results for Combined Continuous User Authentication and Intrusion Detection in High Security Mobile Ad Hoc Networks, IEEE Transactions on Wireless Communications, Vol.10, No.9, pp.3064-3073, Sep. 2011.

[15] J.Liu, F.R.Yu, C. Lung, H. Tang, Optimal Combined Intrusion Detection and Biometric-based Continuous Authentication in High Security Mobile Ad Hoc Networks, IEEE Transactions on Wireless Communications, Vol.8, No.2, pp-806-815, Feb. 2009.

[16] A. Hasswa, M.Zulkernine and H.Hassanein, Routeguard: an Intrusion Detection and Response System for Mobile Ad Hoc Network, Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2005), Vol.3, pp.336-343, 2005.

[17] A. Mitrokotsa and C. Dimitrakakis, Intrusion Detection in MANET using Classification Algorithms: The effects of Cost and Model Selection, Elsevier Journal of Ad Hoc Networks. Vol.11, No.1, pp.226-237, 2013.

[18] KDD Data Set used in 3rd International Knowledge Discovery and Data Mining Tool Competition, 1999, available at URL http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.