Real-time Intrusion Detection for Ad hoc Networks

Ioanna Stamouli, Patroklos G. Argyroudis, and Hitesh Tewari
Department of Computer Science
University of Dublin, Trinity College, Ireland
{stamouli, argp, htewari}@cs.tcd.ie

Abstract

A mobile ad hoc network is a collection of nodes that is connected through a wireless medium forming rapidly changing topologies. The widely accepted existing routing protocols designed to accommodate the needs of such self-organised networks do not address possible threats aiming at the disruption of the protocol itself. The assumption of a trusted environment is not one that can be realistically expected; hence several efforts have been made towards the design of a secure routing protocol for ad hoc networks. The main problems with this approach are that it requires changes to the underlying routing protocol and that manual configuration of the initial security associations cannot be completely avoided. In this paper we propose RIDAN, a novel architecture that uses knowledge-based intrusion detection techniques to detect in real-time attacks that an adversary can perform against the routing fabric of a mobile ad hoc network. Our system is designed to take countermeasures minimising the effectiveness of an attack and maintaining the performance of the network within acceptable limits. RIDAN does not introduce any changes to the underlying routing protocol since it operates as an intermediate component between the network traffic and the utilised protocol with minimum processing overhead. We have developed a prototype that was evaluated in AODV-enabled networks using the network simulator (ns-2).

1. Introduction

Mobile ad hoc networks consist of nodes that are able to communicate through the use of wireless media and form dynamic topologies. The basic characteristic of these networks is the complete lack of any kind of infrastructure, and therefore the absence of dedicated nodes that provide network management operations. In order to maintain connectivity in a mobile ad hoc network all participating nodes have to perform routing of network traffic. The cooperation of nodes cannot be enforced by a centralised administration since one does not exist. Therefore, a network layer protocol designed for such self-organised networks must enforce connectivity and security requirements in order to guarantee the undisrupted operation of the higher layer protocols. Unfortunately all of the widely used ad hoc routing protocols have no security considerations and trust all the participants to correctly forward routing and data traffic. The routing protocol sets the upper limit to security in any packet network. If routing can be misdirected or modified the entire network can be paralysed [12]. Several efforts have been made towards the design of a secure routing protocol for ad hoc networks. The main problems with this approach are that it requires changes to the underlying routing protocol and that manual configuration of the initial security associations cannot be completely avoided.

The Real-time Intrusion Detection for Ad hoc Networks (RIDAN) system is based on previous research proposed to detect link-state routing attacks against OSPF, a routing protocol that is widely used in wired networks [2]. We have adopted the successful approach of employing timed finite state machines for detecting attacks in real-time and have applied it in the domain of ad hoc routing. RIDAN can be characterised as an architecture model for intrusion detection in ad hoc networks, while its implementation targets specifically AODV [11]. We classify our system as an architecture model since it does not perform any changes to the underlying routing protocol but it merely intercepts traffic and acts upon recognised patterns.

In the remainder of this paper we start by briefly presenting the related work on this area in section 2. In section 3 we describe the AODV routing protocol and the threat model associated with it. Section 4 presents in detail our proposed architecture and the design of RIDAN for AODV-based networks. In section 5 we evaluate our prototype that has been implemented using ns-2. Section 6 concludes by discussing the strengths and the shortcomings of our proposal identifying directions for future work.

2. Related work

Knowledge-based intrusion detection systems accumulate knowledge about attacks, examine traffic and try to identify patterns indicating that a suspicious activity is occurring. This approach can be applied against known attack patterns only and the utilised knowledge base needs to be updated frequently [2]. Knowledge-based systems are particularly attractive due to their low false alarm rates and high accuracy. A real-time knowledge-based network intrusion detection model for detecting link-state routing protocol attacks has been developed specifically for OSPF [2]. The model is composed of three main layers; a data process layer, an event abstractor layer and an extended finite state machine layer. The data process layer is used to parse packets and dispatch data, while the event abstractor is used to abstract predefined real-time events for the link-state protocol. The extended finite state machine (FSM) layer is used to express the real-time behaviour of the protocol engine and to detect intrusions by using pattern matching. A timed FSM, referred to as *JiNao Finite State Machine* (JFSM) by the authors, extends the

Proceedings of the Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM'05)
0-7695-2342-0/05 $20.00 © 2005 IEEE

conventional FSM model with timed states and time constraints on the state transition process. The results of this research clearly demonstrate that this approach is very effective in identifying real-time intrusions and especially known attacks. In RIDAN we use this work as a basis and apply the developed concepts in the field of ad hoc networking environments and more specifically to the AODV routing protocol.

The *watchdog and pathrater* scheme has suggested two extensions to the DSR ad hoc routing protocol that attempt to detect and mitigate the effects of nodes that do not forward packets although they have agreed to do so [8]. The *watchdog* extension is responsible for monitoring that the next node in the path forwards data packets by listening in promiscuous mode. It identifies as misbehaving nodes the ones that fail to do so. The *pathrater* assesses the results of the watchdog and selects the most reliable path for packet delivery. As the authors of the scheme have identified, the main problem with this approach is its vulnerability to blackmail attacks.

Another similar approach that takes advantage of *fear-based awareness* has been presented in [10]. The proposed system utilises hash chains in the route discovery phase of DSR and promiscuous mode to observe malicious acts of neighbour nodes. The observers of the malicious node report their findings to the source node which calculates a rating for the accused node. Each source node advertises the ratings it has calculated in order to allow other nodes to decide on whether to provide routing services to the attacker. Although this system cannot be classified as a pure intrusion detection system since it uses cryptographic mechanisms to detect attacks, it holds many properties like network auditing to decide whether a node is performing an attack.

A cooperative distributed intrusion detection system (IDS) has been proposed in [14] by Zhang and Lee. According to their scheme every node that participates in a mobile ad hoc network analyses locally available network data for anomalies. Intrusion attempts are detected by employing a distributed cooperative mechanism in which all participating nodes cast votes according to the data they have previously analysed. The authors avoid the reliance on known attack patterns by using an anomaly detection model. However, all such IDS models suffer from performance penalties and high false alarm rates. Furthermore, the authors do not present any performance or detection accuracy analysis of their proposed architecture.

In [5] the authors present an IDS for wireless ad hoc networks based on a mobile agent framework. According to their proposal multiple sensors deployed throughout the network collect and merge audit data implementing a cooperative detection algorithm. Few, but not all, of the participating nodes are chosen to host sensors that monitor the traffic of the network. The selection of these nodes is based on their connectivity index and a distributed voting algorithm. The detection decisions are taken by mobile agents that migrate their execution and state information between the different sensor hosts of the network, and finally return to the originator host with the results. The authors propose two different methods of decision making, collaborative and independent. They argue that independent decision making by mobile agents is susceptible to single point of failure problems and therefore propose the use of the collaborative method. The main advantage of their approach is the restriction of computation-intensive operations of the system to few dynamically elected nodes. However, most available mobile agent frameworks are heavyweight and can often be the targets of attacks themselves [7].

3. AODV Security Problems

In this section we present an overview of the AODV ad hoc routing protocol and the threat model associated with it.

3.1. AODV Overview

AODV being a reactive protocol does not require the maintenance of routes to destinations that are not in active communication; instead it allows mobile nodes to obtain routes quickly to new destinations. Moreover, AODV provides loop-freedom that is accomplished through the use of sequence numbers. Every node maintains its own sequence number that it increases monotonically each time it learns of a change in the topology of its neighbourhood. This sequence number ensures that the most recent route is selected whenever a *route discovery* process is executed [11].

The AODV protocol uses *route request* (RREQ) messages flooded through the network in order to discover the paths required by a source node. An intermediate node that receives a RREQ replies to it using a *route reply* (RREP) message only if it has a route to the destination whose corresponding destination sequence number is greater or equal to the one contained in the RREQ [11]. Otherwise, the intermediate node broadcasts the RREQ packet to its neighbours until it reaches the destination. The destination unicasts a RREP back to the node that initiated the route discovery by transmitting it to the neighbour from which it received the RREQ. As the RREP is propagated back to the source, all intermediate nodes set up forward route entries in their tables. The *route maintenance* process utilises link-layer notifications, which are intercepted by nodes neighbouring the one that caused the error. These nodes generate and forward *route error* (RERR) messages to their neighbours that have been using routes that include the broken link. Following the reception of a RERR message a node initiates a route discovery to replace the failed paths.

3.2. AODV Threat Model

In this section the most important attacks are presented that can be easily be performed by an internal node against AODV [3, 6].

- *Sequence number (or black hole) attack*: As an example of this attack consider the following case. A source node initiates a route discovery process directed to a destination node by sending a RREQ packet. When the attacker receives the RREQ creates a RREP

with a forged sequence number and next hop. In order for the false information to be favoured the malicious node puts a relatively high sequence number to the destination sequence number field. If the RREP from the malicious node is received before the one from the legitimate source node then it manages to put itself in the route and it can intercept the routing packets. Even if the malicious RREP does not reach the source node first it will eventually reach it and because the destination sequence number will be greater the original route will be replaced by the forged one. The strength of this attack is that the forged route will be propagated from the legitimate nodes as well since they will reply to future RREQs with the false entries that exist in their routing tables. Thus, the false routing information will propagate to other nodes without the intervention of the malicious node. The aim of the attacker is to segment the network by dropping all incoming traffic or to perform the first step of a man-in-the-middle attack.

- *Resource consumption*: In this attack the malicious node attempts to consume both the network and node resources by generating and sending frequent unnecessary routing traffic. This routing traffic can only be RREQ and RERR packets since all false RREPs are automatically discarded according to the specification of the AODV protocol. The goal of this attack is to flood the network with false routing packets to consume all the available network bandwidth with irrelevant traffic and to consume energy and processing power from the participating nodes.

- *Dropping routing traffic*: Mobile nodes due to limited battery life and limited processing capabilities may decide not to participate in the routing process in order to conserve energy. Thus, a malicious node upon receiving a routing packet that is not destined for itself or it was not initiated by it deliberately drops it. The node by acting selfishly conserves energy but it may also cause network segmentation. If some of the participating nodes are only connected through the malicious node then they become unreachable and isolated from the rest of the network.

There are several other similar attacks presented in the literature [1, 4] but they exploit more or less the same routing protocol vulnerabilities to achieve their goals. Only one of these attacks (sequence number) is specific to AODV, while the other two can be applied to any routing protocol

4. RIDAN Architecture

The RIDAN architecture utilises timed finite state machines (TFSMs) to formally define attacks against the AODV routing process. Therefore, it follows the *knowledge-based* methodology to detect network intrusions. TFSMs enable the system to detect malicious activity in real-time rather than using statistical analysis of previously captured traffic. RIDAN operates locally in every participating node and depends on the network traffic a node observes. Based on the observed packets more than one TFSM may be triggered. The TFSMs were constructed after researching the internal operations of AODV. In order to recognise the patterns occurring when an attack is performed against the routing fabric, the generated AODV traffic was analysed in both its normal operation and when an attack was in progress. The timers that control the transition between the states of the TFSMs where derived from theoretical research and practical experimentation using the network simulator (ns-2).

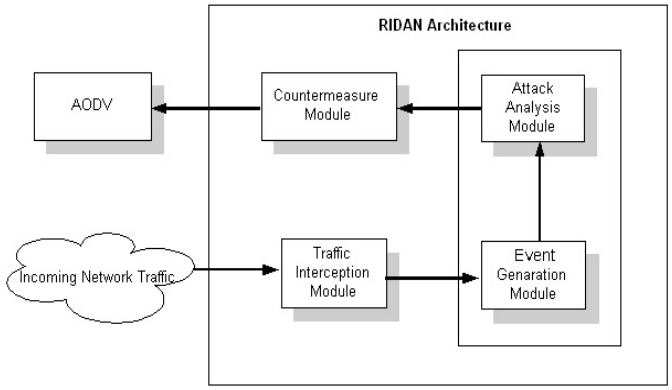


Figure 1. High-level architecture of the RIDAN components.

In Figure 1 the high-level architecture of RIDAN is shown. The *traffic interception module* captures the incoming traffic from the network and selects which of these packets should be further processed. The *event generation module* is responsible for abstracting the essential information required for the *attack analysis module* to determine if there is malicious activity in the network. The event generation and the attack analysis modules are implemented using TFSMs. The final component of the architecture is the *countermeasure module* that is responsible for taking the appropriate actions to keep the network performance within acceptable limits. The RIDAN component operates between the network traffic and the routing protocol. Hence its deployment requires no modifications to the underlying routing protocol.

The assumptions on which our system relies are realistic enough to be implementable in an ad hoc networking environment. They are enumerated below:

- every link between the participating nodes is bidirectional,
- nodes operate in promiscuous mode, meaning that they can listen to their neighbours' transmissions,
- all the participating nodes have RIDAN activated, with the exception of malicious nodes.

4.1. Modelling of the Detection and Countermeasure Components

The design of the TFSMs is crucial for RIDAN since based on their operation a node decides if it should trust another node, or go to an alarm state and take countermeasures against it. The countermeasures that a node takes do not isolate or penalise the

offending node permanently, but for a finite time period in order to avoid the impact of possible false positives. The following paragraphs present the TFSMs that were designed to detect the sequence number, the dropping routing packets and the resource consumption attacks. Also, each section describes the implemented countermeasures.

4.2. Sequence Number Attack

To correctly identify the sequence number attack two different TFSMs are required.

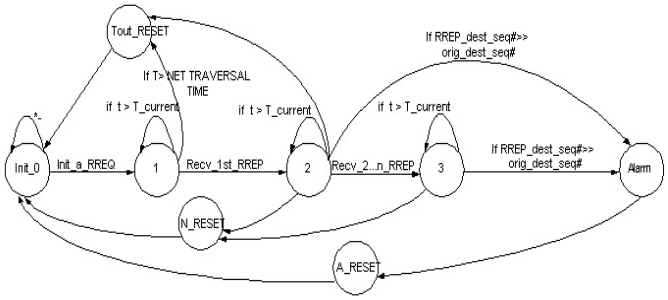


Figure 2. First sequence number attack detection TFSM.

In Figure 2 the TFSM is triggered whenever a node initiates a route discovery process. If a RREP message does not arrive within a predefined time period (*NET_TRAVERSAL_TIME*) the TFSM timeouts (*Tout_RESET*) and resets to its initial state (*init_0*). Upon the reception of the first RREP the TFSM checks if the included destination sequence number (*RREP_dest_seq#*) is much higher than the sequence number included in the RREQ (*orig_dest_seq#*). If it is suspiciously higher it goes directly to the alarm state (*Alarm*). If it is not, it remains in the same state (state *1*) for time *t*. If the timer expires without receiving another RREP it resets normally (*N_RESET*). If within the time limit another RREP(s) arrives, the validity of the destination sequence number is checked again and similarly a decision is taken whether to move to an alarm state. When an alarm occurs the source node knows that the information in the RREP is forged and that it must not update the routing table with the invalid routing information. The following step is to reset (*A_RESET*) the TFSM to its initial state (*init_0*).

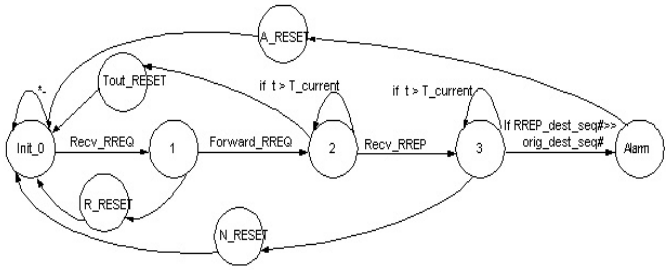


Figure 3. Second sequence number attack detection TFSM.

The second TFSM (Figure 3) for this attack protects the intermediate nodes that receive the RREQ initiated by the source node. When an intermediate node receives a RREQ (state *1*) and has a fresh enough route to the destination it replies (moving the TFSM to the state *R_RESET*). In case the intermediate node does not have the necessary information to reply to this RREQ it forwards the packet downstream and moves to state *2*. The TFSM remains to this state for time *t*. If the timer expires it resets (*Tout_RESET*) and goes back to the initial state (*init_0*). If within a time limit it receives a RREP it moves to state *3* and checks for the validity of the destination sequence number as in the previous TFSM. If the sequence number is within the acceptable limits, the TFSM normally resets (*N_RESET*), otherwise it moves to an alarm state. As a result the forged route is not added to the node's routing table. It should be noted at this point that the intermediate nodes do not drop the RREPs even if they determine that they are forged. Instead the RREPs are unicasted back to the source node which will determine by itself that the packet contained invalid information.

4.3. Dropping Routing Packets Attack

Since all the nodes participating in the network are in promiscuous mode a node can detect whether a neighbouring node has forwarded a routing packet or not. However, the node in question may have not forwarded the routing packet due to traffic overload. In order to prevent false alarms caused by traffic overload, the TFSM moves initially to a pre-alarm state and in this state it unicasts the routing packet to the offending node again.

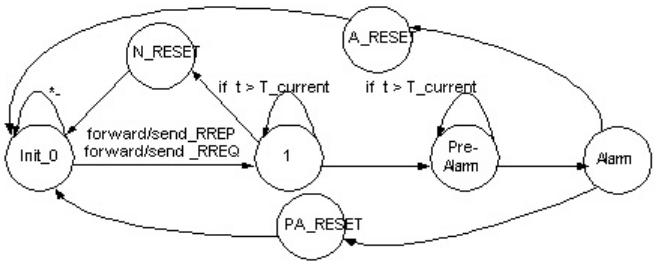


Figure 4. Dropping routing packets attack detection TFSM.

The TFSM (shown in Figure 4) is triggered whenever a node sends or forwards a RREQ or a RREP. It remains in state *1* for time *t* waiting for the node to forward or reply to the routing packet. If the node replies or forwards the packet the TFSM normally resets (*N_RESET*). If the node fails to appropriately respond the TFSM moves to a *Pre-Alarm* state and remains there for time *t*. If the node manages to respond appropriately by forwarding the routing traffic or by replying to a RREQ it is removed from the suspected nodes list and the TFSM normally resets. Otherwise, the TFSM goes to an alarm state and the observing node marks this node as malicious. As a countermeasure the node does not forward any kind of traffic through the identified attacker and also sends a RERR to its upstream neighbours. This action marks the link with the attacking node as broken. When the misbehaving node starts forwarding packets again it will be gradually added to the routing function.

4.4. Resource Consumption Attack

The resource consumption detection TFSM is triggered for every different node that sends a routing packet. The observing

node keeps a list with all the nodes from which it has recently received routing traffic along with a counter that signifies the number of packets from a specific node and a timer.

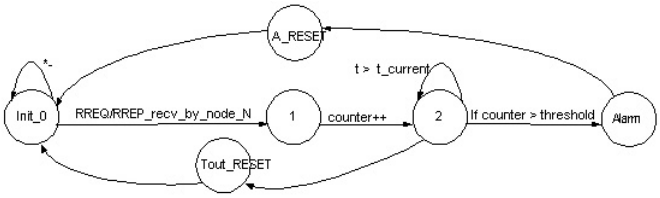


Figure 5. Resource consumption attack detection TFSM.

The TFSM (shown in Figure 5) increments the counter for every new routing packet received from a specific node. It remains in state *2* for time *t*. If the counter reaches the threshold value it means that it has detected abnormal traffic generation and the TFSM moves to the alarm state. Upon an alarm the node drops all the incoming routing traffic from the offending node for a finite time interval so that it does not consume network and node resources. If the timer of state *2* expires the TFSM resets to its initial state indicating that the generated traffic from the monitored node was normal.

5. Evaluation

The experiments for the evaluation of RIDAN were carried out using the network simulator (ns-2). We have evaluated AODV without any modifications, AODV with one malicious node present, and AODV with the RIDAN component enabled having in the network a malicious node. The scenarios developed to carry out the tests use as parameters the mobility of the nodes and the number of active connections in the network. The choices of the simulator parameters that are presented in Table I consider both the accuracy and the efficiency of the simulation.

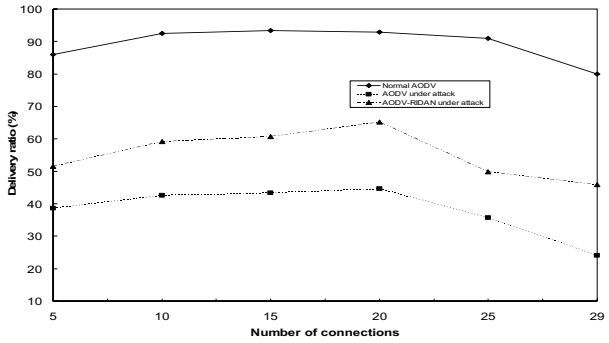
Parameter	Value
Simulation duration	1000 seconds
Simulation area	1000*1000 m
Number of mobile hosts	30
Transmission range	250 m
Movement model	Random waypoint
Maximum speed	5 – 20 m / sec
Traffic type	CBR (UDP)
Data payload	512 bytes
Packet rate	2 pkt / sec
Number of malicious nodes	1
Host pause time	10 seconds

Table 1. Simulation Parameters

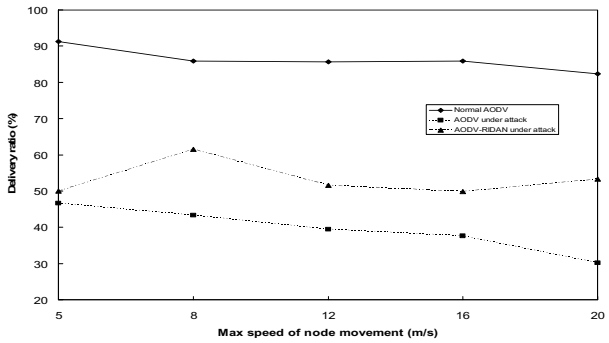
The following metrics were chosen to evaluate the impact of the implemented attacks: (1) packet delivery ratio, (2) false routing packets sent by the attacker, (3) additional routing overhead introduced by the attacker and (4) routing packet dropped ratio. These metrics were used to measure the severity of each attack and the improvement that RIDAN manages to achieve during active attacks. Every point in the produced graphs is an average value of data collected from repeating the same experiment ten times in order to achieve more realistic measurements.

5.2. Sequence Number Attack

The two metrics that were used in the evaluation of the sequence number attack detection and countermeasure mechanism are the delivery ratio and the number of false routing packets sent by the attacker. The former was plotted against the number of active connections (Figure 6(a)) and against the node mobility (Figure 6(b)).



(a)



(b)

Figure 6. Delivery ratio versus number of connections and node mobility in the sequence number attack.

A general observation shows that AODV achieves maximum delivery ratio between 10 to 25 active connections and when this number is increased to 29 there is a slight decrease to 80%. The sequence number attack has a very big impact in the delivery ratio decreasing it to lower than the half compared to normal AODV. RIDAN manages to keep the delivery ratio higher, at around 60%, having a significant improvement. In Figure 6(b) we observe that AODV performs better in low node mobility rates while as the mobility rate increases the delivery ratio slightly drops. The performance of the network is also significantly reduced when AODV is under the sequence number attack and the node mobility increases. However this behaviour is normal because as the node mobility increases the network topology changes making route requests more frequent, and therefore a malicious node has the opportunity to send more false RREP packets. AODV under attack exhibits a decrease of delivery ratio to 37.7%. When

RIDAN is enabled the delivery ratio is increased to 54.3% having an average improvement of 16.6%.

The second metric that was used in the evaluation of this attack was the number of false packets sent by the attacking node versus the number of active connections and the node mobility. This metric was used to examine the overhead of the sequence number attack and we considered only the extra cost on communication imposed by the attack. We have observed that the average number of false RREPs sent by the malicious node in all the evaluated experiments was 2056 and the number of nodes that inserted the false route into their routing table was 22 out of 30.

5.3. Dropping Routing Packets Attack

To evaluate RIDAN during the dropping routing packets attack the metrics of delivery ratio and routing overhead ratio were utilised. The delivery ratio is plotted against the number of active connections (Figure 7(a)) and against node mobility (Figure 7(b)).

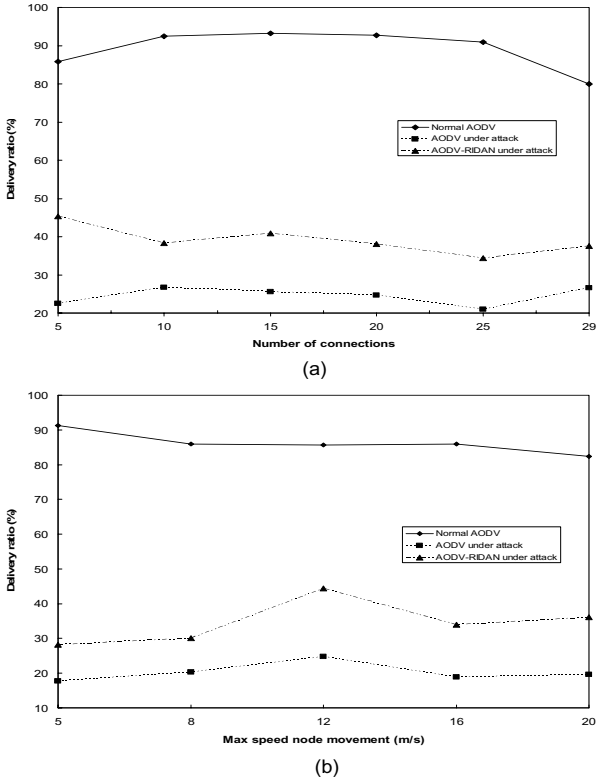


Figure 7. Delivery ratio versus number of connections and node mobility in the dropping routing packets attack.

The dropping routing packets attack has a major impact in network connectivity decreasing the delivery ratio by 64.7%. Even though RIDAN improves the delivery ratio by informing all the other nodes about the attacker, the improvement is not very high (around 15%) since the system cannot force the malicious node to forward routing traffic. In Figure 7(b) we observe an even greater decrease in the delivery ratio (66.2%) when normal AODV is under attack. As the node mobility increases the participating nodes initiate route discovery processes more frequently and the malicious node can drop more routing packets. AODV with the RIDAN system enabled improves the delivery ratio to 34.5% having an improvement of 14.5%.

The second metric used in the evaluation of RIDAN for this attack was the routing overhead ratio. The additional routing overhead introduced by the attacking node reaches 78% when the network size is small and decreases as the number of connections increases. This behaviour is normal since when there are only five active connections in the network the routes to a destination node are limited, and therefore it is essential that all nodes participate in the routing process. Our results show that the routing overhead introduced by the attack reaches 50.3% while normal AODV has only 38.7%. AODV-RIDAN decreases it to 45.1% having an average improvement of 6.1%. As we have already pointed out, the improvement is not very high since RIDAN does not introduce any mechanism that forces a misbehaving node to participate in the routing process. However, RIDAN manages to reduce the routing overhead ratio to approximately the levels that normal AODV demonstrates.

5.4. Resource Consumption Attack

To evaluate RIDAN during the resource consumption attack the delivery ratio and the dropping rate of routing packets were used as metrics. The former is plotted against the number of connections (Figure 8(a)) and node mobility (Figure 8(b)). The number of additional routing packets that were generated by the attacking node was randomly chosen between 1 and 10 for every legitimate packet that was sent in order to demonstrate a realistic attempt to consume resources.

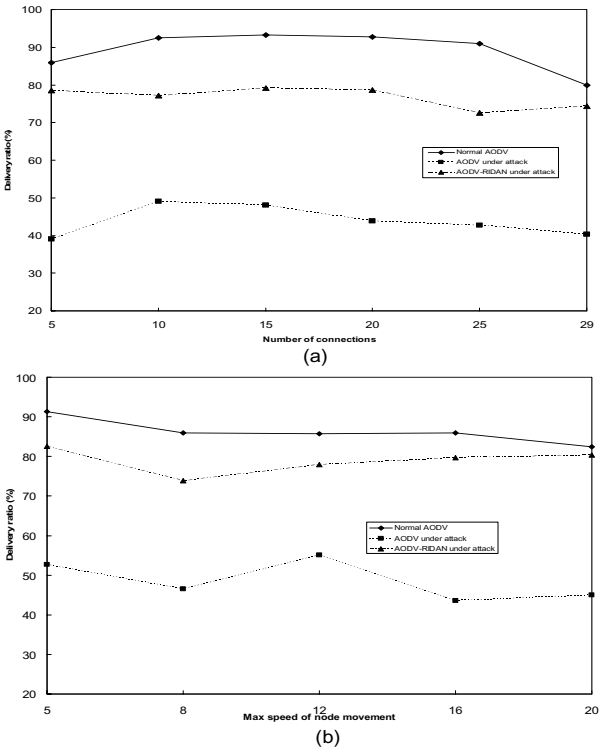


Figure 8. Delivery ratio versus number of connections and node mobility in the resource consumption attack.

The delivery ratio is visibly reduced when AODV is under the resource consumption attack. However, when the RIDAN system is enabled the delivery ratio stays within acceptable performance limits. When plotted against the number of connections (Figure 8(a)) the delivery ratio is 13% lower than what it should normally be, while in the case of node mobility (Figure 8(b)) the decrease is only 6%. The countermeasure mechanism is very effective since it enables the observing node to drop the irrelevant traffic preventing it from flooding the network.

The second evaluation metric for this attack is the routing packets dropped ratio. RIDAN increases the dropping routing packet ratio to 34% while in normal AODV and in AODV under the resource consumption attack is approximately 1.5%. AODV by default drops very few routing packets; however the resource consumption attack takes advantages of this behaviour to flood the network with unnecessary routing traffic. The fact that the dropping rate is high has a positive impact in the overall performance of the network when a flooding attack is in process.

6. Discussion and Conclusions

All intrusion detection systems suffer from false alarms that occur whenever the system incorrectly concludes in an alarm but there is no malicious behaviour present in the network. The knowledge-based methodology that RIDAN employs to detect malicious activity is less error prone than other detection techniques, like for example the behaviour-based. However, the traffic patterns that denote an active attack can be matched when AODV operates normally due to high application traffic and high node mobility. RIDAN was tested in terms of detection accuracy and the percentages of successful detection for the three attacks are the following:

- sequence number attack detection accuracy: 81.2%,
- dropping routing packets attack detection accuracy: 71.5%,
- resource consumption attack detection accuracy: 74.8%.

The detection accuracy of RIDAN in all the three attacks can be considered high compared to the results of other similar projects [2, 9, 13].

We must stress that RIDAN is not a complete security solution for ad hoc networks. For example our system is not able to detect attacks that involve impersonation since we do not employ cryptographic mechanisms for address authentication. However, our evaluation has shown that the developed system manages to detect the specified attacks with high accuracy and by taking countermeasures to keep the network performance within acceptable limits during active malicious behaviour. Our primary direction for future work is the specification of more attack patterns for both reactive and proactive protocols. Furthermore, we plan to continue the evaluation of RIDAN in real ad hoc networking environments in order to confirm our positive results.

REFERENCES

[1] P. Albers, O. Camp, J.M. Parcher, B. Jouga, L. Me and R. Puttini, "Security in Ad hoc Networks: a General Intrusion Detection Architecture Enhancing Trust Based Approaches", In *Proc. Int'l. Workshop on Wireless Information Systems,* 2002.

[2] H.–Y. Chang, S.F. Wu and Y.F. Jou, "Real-Time Protocol Analysis for Detecting Link-State Routing Protocol Attacks", *ACM Tran. Inf. Sys. Sec.*, vol 1, pp. 1-36, 2001.

[3] B. Dahill, B.N. Levine, E. Royer and C. Shields, "A Secure Routing Protocol for Ad hoc Networks", *Technical report, UM-CS-2001-037, University of Massachusetts*, 2001.

[4] A. Habib, M.H. Hafeeda and B. Bhargava, "Detecting Service Violation and DoS Attacks", In *Proc. Network and Distributed System Security Symposium*, 2003.

[5] O. Kachirski and R. Guha, "Effective Intrusion Detection using Multiple Sensors in Wireless Ad hoc Networks", In *Proc. 36th Annual Hawaii Int'l. Conf. on System Sciences (HICSS'03)*, pp.57.1, 2003.

[6] J. Lundberg, "Routing Security in Ad hoc Networks", http://citeseer.nj.nec.com/400961.html.

[7] M.C. Man and V.K. Wei, "A Taxonomy for Attacks on Mobile Agent", In *Proc. Int'l. Conf. on Trends in Communications*, vol 2, pp. 385-388, 2001.

[8] S. Marti, T.J. Giuli, K. Lai and M. Baker, "Mitigating Routing Misbehaviour in Mobile Ad hoc Networks", In *Proc. ACM/IEEE Int'l. Conf. on Mobile Computing and Networking*, pp. 255-265, 2000.

[9] Y. Okazaki and I. Sato, "A New Intrusion Detection Method based on Process Profiling", In *Proc. Symposium on Applications and the Internet*, pp. 82-90, 2002.

[10] K. Paul and D. Westhoff, "Context Aware Detection of Selfish Nodes in DSR based Ad hoc Networks", In *Proc. IEEE Vehicular Technology Conf.*, 2002.

[11] C. Perkins, E. Royer and S. Das, "Ad hoc On-demand Distance Vector (AODV)", *RFC 3561*, 2003.

[12] W. Wang, Y. Lu and B.K. Bhargava, "On Vulnerability and Protection of Ad hoc On-demand Distance Vector Protocol", In *Proc. Int'l. Conf. on Telecommunications,* 2003.

[13] N. Ye, S.M. Emran, X. Li and Q. Chen, "Statistical Process for Computer Intrusion Detection", In *Proc. DARPA Inf. Survivability Conf.*, pp 3-14, 2001.

[14] Y. Zhang and W. Lee, "Intrusion Detection in Wireless Ad hoc Networks", In *Proc. ACM/IEEE Int'l. Conf. on Mobile Computing and Networking*, pp 275-283, 2000.