International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2021/209992 Volume 8, Issue 5, September – October (2021)

RESEARCH ARTICLE

Securing Service Discovery from Denial of Service Attack in Mobile Ad Hoc Network (MANET)

Smitha Kurian
Department of Computer Science and Engineering, HKBK College of Engineering, Research Centre, VTU, Karnataka, India
smitha.nikku@gmail.com

Loganathan Ramasamy
Department of Computer Science and Engineering, HKBK College of Engineering, Research Centre, VTU, Karnataka, India
drloganatahanr@gmail.com

Received: 23 August 2021 / Revised: 08 October 2021 / Accepted: 13 October 2021 / Published: 27 October 2021

Abstract – Mobile Ad Hoc networks (MANET) are resource constrained and operate on the basis of mutual cooperation. As a result, service discovery is one of the essential services of MANET. Service discovery was integrated onto Ad Hoc on Demand Distance Vector (AODV) Routing protocol, since service discovery was best performed at the network layer with minimal control messages. But this integration echoes the security threats of AODV protocol onto the service discovery process. The security of AODV protocol has drawn ample attention and various studies and methodologies are proposed. But most of the proposed techniques either address the flooding attack or the black hole attack but addressing both these issues simultaneously has been a challenge. Since the nodes in the network are resource constrained achieving the security objective with minimal overhead is also a target that needs to be achieved. We propose a trust based methodology at the level of individual node, that avoids the denial of service attack by controlling both the packet dropping attack and the flooding attack of the service discovery extended AODV protocol. This scheme assists in the selection of a safe path between the consumer and the server by ensuring that a cooperative node with high trust is selected at every hop. The trust value of the non-cooperative or flooding nodes is decreased and is thus avoided from safe paths. With simulated experiments it is demonstrated that the proposed system has 4% lesser control message overhead, the service discovery ratio improved by 13% and the service discovery latency was also considerably reduced.

Index Terms – Service Discovery, AODV, Flooding Attack, Packet Dropping Attack, Denial of Service, Sleep Deprivation.

1. INTRODUCTION

The usage of mobile devices like laptops, PDA etc. are on the high as a result of which the popularity of mobile Ad Hoc networks is on the rise. They are widely applied in defense related operations, disaster rescue operations, collaborations at conferences etc. MANETS are self-configurable, independent, and self-deployable and does not require the support of a central control. The devices in MANETS are mobile and therefore the network is completely dynamic.

MANET is resource constrained and therefore they operate on the basis of mutual cooperation, resource sharing and trust. To discover and share the resources service discovery is inevitable in such a network. Service discovery is an operation specific to application layer but is established in article [1] that it is best performed at network layer therefore service discovery was integrated to the network layer. AODV protocol was a good choice as it was a reactive routing protocol and is efficient.

The design of the routing protocols is based on the assumption that the nodes operate based on mutual cooperation and trust. The nodes communicate on hop by hop basis with the cooperation of the neighboring nodes. When the nodes in the network fail to cooperate with each other and misbehaves, the normal operation of the network is disrupted. According to article [2] Misbehaviors occur when the nodes are faulty, or are selfish therefore do not cooperate to save its resources like battery power or are malicious node who deliberately disturb the network operations. The malicious nodes are the biggest threat to the smooth functioning of the network.

A. Nadeem and M. P. Howarth in [3] propose that basically there are two types of malicious node attacks at the network layer; the passive and the active attacks. The passive attacks are the eavesdropping, location disclosure and traffic analysis. In most cases the passive attacks are not severe except in scenarios like military operations. The active attacks are mainly routing and packet dropping attacks. Sometimes these attacks are individual and at times the attack is collaborative. The active attacks are severe and can cause significant performance degradation. Therefore in this paper we are addressing the active attacks that are vulnerable to the AODV routing protocol.

1.1. Problem Statement

The most common routing attack that can affect service discovery is the denial of service attack (DoS), an attack that hinders the discovery and accessibility of service. A DOS attack can be launched as a sleep deprivation attack, flooding attack, black hole attack and gray hole attack. In [4] sleep deprivation attack where the attacker interacts with the victim node to deprive it of its power conserving sleep mode. In flooding attack the network is flooded with control messages so that the network is congested and cannot operate normally in a black hole attack [5] the intruder captures a route by responding to a Route Request (RREQ) message quickly without referring the routing table with a Route Reply (RREP) message that has a higher sequence number. Once it captures the node it drops all the packets that it receives. The gray hole attack [6] is similar to black hole attack where in the packets are dropped selectively.

So the AODV extended to include service discovery has to be enhanced with techniques that

- Controls the flooding of request message that can bring down individual servers or the network as a whole
- Control the packet dropping attack which affects service accessibility.
- Limit the process overhead considering the nodes are resource constrained.

1.2. Open Issues with the AODV Extended Service Discovery

From the study of the related work in section 2, it can be observed that security is generally overlooked during service discovery. But the proposal that secures the AODV protocol can secure the service discovery process as well. So the study proceeded to the various literatures that secure the AODV protocol. Extensive study is underway and various techniques are proposed to mitigate packet dropping attack (Black Hole, Gray Hole attacks) and the flooding attack. The challenges to be addressed are:

- A common technique that controls both, flooding and packet dropping attack.
- Identifying safe path including cooperating nodes
- Avoid computation intensive task as the resources are constrained.

1.3. Objective and Research Contribution

The main objective of this proposal is to secure the service discovery process from denial of service attack by limiting the packet dropping and flooding attack considering the constraints of the network.

To achieve the objective a trust based scheme is devised that helps in identifying the cooperating nodes from among the set of all one hop nodes. Each node monitors the behavior of its one hop neighbor based on the messages received from them as a part of the regular communication process. The trust value is increased as a reward for cooperation and the trust values are decreased for misbehaviors like packet drop and flooding. This process is continuously performed as part of its communication without additional overhead so that each node is aware of the set of reliable one hop nodes at all time. During the selection of a path form source to destination a safe path is selected by selecting reliable nodes at each hop till the destination is reached. Any node whose trust value is low is added to the blacklist and silently ignored from further communication.

The reason for the misbehavior could be the fault in a node or selfish behavior to save resources or a deliberate misbehavior. We do not investigate the reason for noncooperation. Any node that is not cooperating is added to “blacklist” set for a “BLACKLIST_TIMEOUT” period. In the next cycle if the nodes co-operate the nodes will be reconsidered.

The paper is further organized with related work being discussed in section 2. The proposed methodology is outlined in section 3. Section 4 implements the proposal, summarizes the results and observation. Section 5 concludes the findings.

2. RELATED WORK

This section carries out the survey on the various techniques employed to mitigate the routing attacks especially on the AODV protocol. L. Tamilselvan and V. Sankaranarayanan in [7] propose to avoid black hole attack by collecting all replies from its neighbour until the expiration of its timer. It stores sequence number and arrival time of replies. From which it selects a path with repeated next hop node and avoid path without repeated next hop node assuming it to be path with malicious node. This method calls for additional storage.

Payal N. Raj et al. In [8] detect black hole nodes by comparing sequence number against threshold value which are dynamically computed based on timers. They also alert the neighbouring nodes about the malicious node. With this method there is a possibility of false alarms and involves the overhead of dynamic threshold value computation.

V. Mohite and L. Ragha in [9] Proposes a method to detect black hole and cooperative black hole attack by securely transmitting the history records of the packet and analyzing the packet delivery record. This method has to bear the overhead of encrypting the history records, sharing of keys and maintaining Data Routing information table.

K. S. Chavda and A. V. Nimavat in [10] have proposed to prevent black hole attack by comparing the RREP messages from different nodes. The reply with the unusually high destination sequence number is treated as malicious and the neighbouring nodes are alerted. This method may have the problem of false alarms and also the latency will increase since it has to collect all the RREP messages before the comparison can be made.

T. Varshney et al. in [11] propose a watchdog AODV where the nodes monitor its downstream nodes by overhearing the messages. If the watchdog detects a malicious node such nodes are avoided from the path where the packets are forwarded. Here nodes are dedicated as watch dog nodes to monitor its neighbour in promiscuous mode. But this method depends on the sufficient allocation of watchdog nodes across the network.

T. Shu and M. Krunz in [12] have studied a case where the attacker selectively drops few packets and have proposed a technique to verify the packet loss information reported by nodes based on public auditing architecture. But this method is only applicable to static or quasi static Ad Hoc network and is dependent on third party centralized architecture.

Vimal Kumar et al. proposes a methodology in [13] to detect black hole attack by comparing the difference of the sequence number of an RREQ packet and the corresponding RREP packet with a predefined threshold value to detect black hole nodes. This method detects black hole attack launched by modifying sequence number. The computation of threshold values is not clear.S. Jain et al. in [14] have proposed a method where a base node is introduced that sends dummy RREQ messages. They suggest that normal nodes will not respond and only black hole nodes respond and all the nodes that responded are marked as malicious nodes and other nodes are notified about it. The glitch here is that if the black hole node is an internal node it would not respond to it. If the RREQ messages addressed to a non existing node is forwarded in the network, the network will be flooded and may cause a sleep deprivation attack as suggested in [3].

Adwan Yasin et al. in [15] detects black hole attack by dedicating nodes to generate bait with a fake id and broadcast at random interval. The TTL value is set to 1 to avoid flooding of network. All the nodes that responded to the bait are marked as black hole nodes and avoided from communication process. This method is similar to technique proposed in [14] with the contribution that when the TTL is set to one the network is not flooded. This method can detect an external blackhole attack and this method also prevents the network from getting flooded with the bait messages. But if the black hole attack is launched by an internal node it would not answer to the bait message and will go undetected. Multiple nodes will have to generate bait messages to detect black hole nodes at each hop.

Musale S.S et al. proposes a technique in [16] to mitigate gray hole and cooperative attack by sending request messages to one-hop and two hop neighbours and the node that replies first in both these cases is marked as malicious. This technique also detects black hole attack but may land up avoiding legitimate nodes who promptly replies and selects node with higher latency.

Ali H. Ahmed et al. in [17] provides secured service discovery in IoT devices. They propose a secured and broker based service discovery. Consumer submits encrypted query to service broker who then assigns trust values to objects and matches consumer to most appropriate service providers. They suggest that the main requirement for efficient and effective service discovery as security light weight and trust. The main disadvantage with this method is the need for a centralized broker. This technique also involves encryption demanding public key infrastructure.

Mahmoud Abu Zant et al. in [18] have proposed a method to identify flooding attack by the use of a counter that counts the number of request generated by the node. If the request generated by the node is above a threshold the node is added to the suspicious list and the request is dropped so that the network is not flooded. Further they also maintain a request destination identifier table that records the source and the destination of the request. If a node is trying to establish connection to multiple destination nodes simultaneously, and if such a node is there in the suspicious list it is now added to the black list. This technique successfully avoids flooding attack but has to incur the overhead of additional storage.

Hwanseok Yang in [19] proposes a cluster based structure where nodes designated as trust nodes measures the reliability of nodes based on the quality and the number of packets forwarded. The trust information is digitally signed and exchanged between neighbouring trust nodes. Based on the trust values a reliable path is selected for communication. This method may not be suitable in a pure Ad Hoc environment. In Ad Hoc networks the nodes are mobile and the neighbours are dynamic so selection of trust node for each cluster and then computing trust may involve considerable amount of overhead.

Tripathy et al. in [20] proposes an adaptive routing protocol that dynamically configures the routing function as per requirement parameters contextual parameters such as mobility of nodes, trust values, resource constraints etc. The method though did not target specific security issue but trying to assure overall security. Dynamic configuration and the maintenance of the various parameters make this method computationally highly demanding.

Ran eta al. in [21] proposes a multipath secured routing algorithm based on the new improved AODV protocol using the Block chain technology. Here two optimized paths are chosen avoiding abnormal nodes. This improves the security of the routing process. However this process may not be suitable in an environment where the nodes are highly mobile. The process of path selection involves highly computation demanding task and so if the nodes are static this method would be more effective. If the nodes are highly mobile the path discovered may not be active for a long time.

A. M. El-Semary and H. Diab proposes a BP AODV [22] that protects the network from cooperative black hole attacks by establishing trusted routes using a challenge response and confirm pattern. One of the advantages is that it allows multiple paths. The main disadvantage is high latency involved in the process.

Y. Fu et al. Proposes a technique [23] to identify malicious nodes by comparing the difference in sequence number of the initial state and reply messages as a threshold. This method is similar to the methods already proposed. Authors have contributed a new algorithm for threshold computation. The sequence numbers are not static so the process of threshold identification has to be very dynamic and contribute to the time complexity.

Muruganantham Ponnusamy et al. Proposes a scheme called SNRRM[24] that Selects nodes with reputation and high residual energy on a reliable path. The reputation of a node is calculated based on the communication ratio which is the ratio of the number of request that got response to the number of request generated and based on residual energy of a node. The nodes who reply to the messages are treated as reputed nodes. This method has to bear the overhead to keep a count of the number of request generated and response obtained for each node.

Alaa Althalji et al. in [25] proposed a defensive AODV model based on immune system algorithm called the V-Detector algorithm. The have built the defense mechanism that identifies fake reply message comparing the lifetime, hop count and destination sequence number and thus identifies malicious nodes and ignores them. Time complexity of Immune system algorithms are generally high. The V-detector algorithm has a time complexity that is comparatively low, but it is still high to be implemented in Mobile Ad Hoc network environment.

Ankit Kumar & Madhavi Sinha in [26] have through compared the working of the AODV protocol in the presence of Black hole attack and flooding attack and have demonstrated that more delay is introduced when there is a black hole attack when compared to flooding attack. Flooding attack consumes higher bandwidth. But both the attacks degrade the performance of the network. The main disadvantage here is that the paper does not discuss how the proposed changes can control the malicious node attacks.

Md Ibrahim et al. in [27] has proposed a technique to protect the AODV protocol against black hole attack by using digital signatures. Each node is issued a digital signature. When a reply is obtained the requesting node verifies the digital signature of all nodes in path and accepts a node only if its signature is valid. The method has high computation overhead for digital signature verification. Needs a centralized control to issue signatures. The signatures of all valid nodes has to be stored by all nodes for verification by comparison else a centralized authority will be required for the same. A comparison of the various literature surveyed is summarized in Table 1.

Reference	Technique used	Advantage	Disadvantage
. L. Tamilselvan et al.- [7]	Collect replies and chose path with repeated nodes	Detects and avoids Black hole attack during route discovery	Requires additional storage. Time consuming to find paths with repeated nodes.
Payal N. Raj et al.-[8]	Comparing Sequence number against threshold values.	Detects and avoids Black hole attack during route discovery	False alarms could avoid cooperative nodes also
V. Mohite and L. Ragha- [9]	Analyzing haring encrypted history records to study communication pattern	Detect and avoids Black hole Attack	High computation and need of Public Key Infrastructure.
K. S. Chavda and A. V. Nimavat in -[10]	Comparing sequence number from replies and rejecting replies with high sequence number. Inform neighbours about malicious node.	Detects and avoids Black hole attack during route discovery	False alarms could avoid cooperative nodes also
T. Varshney et	Watchdog monitoring	Detect Misbehaviours	Nodes have to be dedicated as

al. -[11]	neighboring nodes		Watchdog.
T. Shu and M. Krunz - [12]	Analyzing packet loss information.	Detect Gray hole Attacks	Requires public auditing architecture, Not suitable for pure Ad Hoc network
Vimal Kumar et al. -[13]	Comparing the difference in sequence number against threshold value	Detects and avoids Black hole attack during route discovery	False alarms Computation of threshold value not clear
S. Jain et al. -[14]	Sending Fake RREQ messages	Detect Black hole Attack	Network is flooded with fake messages. Internal malicious node cannot be detected.
Adwan Yasin et al. -[15]	Sending Fake RREQ messages up to TTL=1. Similar to [14]	Detect Black hole Attack	Internal malicious node cannot be detected. Multiple fake messages have to be generated to identify malicious nodes at multiple hop.
Musale S.S et al. -[16]	Sending Request message to one hop and two hop nodes. Node with first reply is marked malicious	Detects Gray hole attack	Sometimes even legitimate nodes in close proximity that replies first will be avoided.
Ali H. Ahmed et al -[17]	Consumer and servers are mapped by centralized broker	Secured Service Discovery in IoT devices	Requires centralized broker not suitable for AdHoc environment. Requires public key Infrastructure
Mahmoud Abu Zant et al. -[18] AIF_AODV	Counting the number of request generated	Detects Flooding Attack	Computation of Threshold value not defined
Hwanseok Yang[19]	Cluster heads called trust nodes computes trust values of cluster nodes.	Secures Routing protocol	Overhead is high. Not suitable in pure Ad Hoc environment.
Tripathy et al. -[20]	Dynamic routing configuration considering contextual parameters	Securing Routing Protocol	Computationally demanding
Ran eta al. -[21]	Multipath technology with Block chain technology	Securing Routing Protocol	Not suitable if nodes are highly mobile.
A. M. El-Semary and H. Diab -[22]	Establishing trusted routes using challenge, response and confirm pattern	Detects cooperative Black hole attack	Time complexity is high
Y. Fu et al.- [23]	Compares Sequence number of reply message against threshold. Similar to [10]	Proposed algorithm for threshold computation	The sequence numbers are increasing always. The threshold computation has to be highly dynamic.
Muruganantham Ponnusamy et	Finds reliable path by selecting nodes that are reliable and high	Can identify packet dropping attack	Additional computation is involved at every hop.

al.-[24] SNRRM	residual energy.		Overhead of maintaining transmission information
Alaa Althalji et al. - [25]	Identifies Malicious node by using V- Detector immune system algorithm considering, the life time, hop count and destination sequence number	Identifies malicious node	High Time complexity
Ankit Kumar et al. - [26]	Compares and studies the effect of Black Hole and Flooding attack on AODV protocol	Identifies Black Hole attack introduces more delay	It is not clear how proposed technique avoids malicious nodes
Md Ibrahim et al. - [27]	Uses digital signatures to identify malicious nodes	Detects and avoids malicious nodes	High time and space complexity. Requires central control for key distribution and maintenance.

Table 1 Summary of Existing System

2.1. Knowledge Gaps Identified

Various methodologies are employed in the surveyed articles to mitigate the security tasks and they have achieved the task of controlling the targeted attacks. If we analyze the techniques it is evident that the techniques that provide high level security in terms of authentication and integrity employ encryption and cryptographic techniques which involves high time complexity and demands infrastructure and a centralized control for key maintenance and to arrive at common process, which is a challenge for Ad Hoc networks and should be avoided if possible at the network layer and to be taken care by the presentation or application layer at which is generally done. Most techniques have targeted either a packet dropping attack or flooding attack. A technique that can control both the attacks together is needed to avoid Denial of service attack.

In the next section we propose a trust based mechanism that controls both the flooding attack and the packet dropping attack and provide secured service discovery as part of its routine operation with minimal overhead and without the need of a central control.

3. PORPOSED MODELLING

The network under Consideration is an Ad Hoc network set up for a specific purpose where the numbers of services available are predefined and fixed. These services are shared based on mutual cooperation. AODV protocol is modified as per [28] to perform service discovery at the network layer The RREQ and RREP messages of AODV protocol are extended and now called the SREQ and SREP messages respectively. Two-hop service information is added as suggested in [29].The routing table is extended to store the service information as suggested in [30]. So we now have a service discovery at the network layer with no security measures.

Service discovery is integrated to AODV protocol assuming that the nodes cooperate with each other. But in practice such ideal condition does not exist and the threats of the AODV protocol will be echoed on to the service discovery process as well. So it is of utmost importance to secure the service discovery process of Ad Hoc network. Various techniques that can secure the AODV protocol is studied in section 2 and the gaps identified are also discussed in section 2.1.

An attempt is made to propose a trust based technique that can limit the packet dropping and the flooding attack with minimal overhead. The process of avoiding malicious node is performed continuously as long as the routes are active as part of its routing operations so that there are minimal additional overheads. Careful observations reveal that AODV protocol host multiple features within the protocol which if enforced can secure the protocol without much overhead and deviation from its basic operating methods. In the protocol most of these features are enforced at the sender end which can be violated by malicious node. Therefore our technique focuses on enforcing these features at the receiver end and securing the protocol as a part of its normal operation.

We propose to secure the process of service discovery by limiting the packet dropping and flooding attack at one hop neighbor level by a trust based scheme where the misbehaving nodes are isolated. Efforts are not invested to differentiate intentional misbehaviours from accidental misbehaviours or misbehaviours to save resources. In either of the cases the performance is affected and so any misbehaving node has to be isolated from the safe path and should be given a chance to be included and considered back in the network when they participate in mutual cooperation.

The basic idea of the methodology is that every node receives messages from its neighbors and reads the messages to check if it is destined to it. As these messages are processed the communication pattern of the node is also analyzed simultaneously to identify misbehaviors. Based on its observation every node maintains a trust value for its one hop neighbors. The trust values are increased for cooperation and decreased for misbehaviors. As the topology is dynamic this list is periodically refreshed. Thus every node is aware of the list of one hop nodes with its trust value and when a choice has to be made to select a node to be a part of a safe path, always trust worthy nodes are preferred by selecting nodes with high trust value in reverse route. As soon as a node in an existing path misbehaves the existing error handing routines are used to inform the precursors and recover. So the implementation of this scheme harnesses the existing routines and hence limits its overhead. We call this extension of AODV that provides secured service discovery as AODV-SSD.

The architecture of the secured service discovery system (AODV-SSD) is outline in Figure 1.

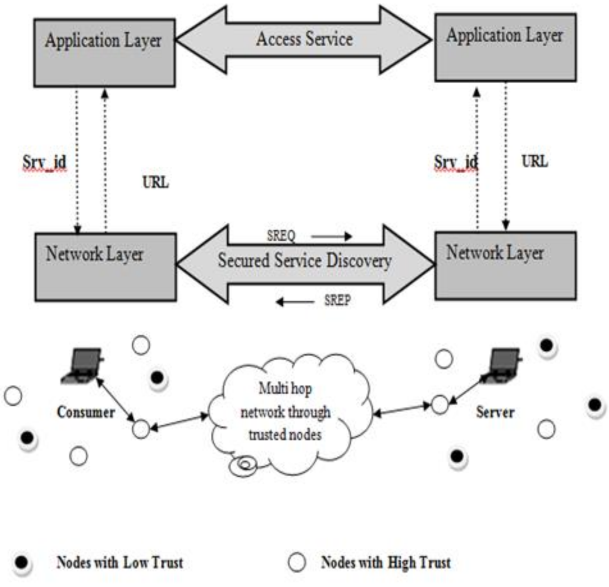


Figure 1 Architecture of Secured Service discovery AODV-SSD

From Figure 1 it can be observed that the services are hosted at the application layer. The application layer maps each service to an unique service identifier (Srv_id). A request to discover the service is sent to the network layer by sending the service identifier. At the network layer the services are discovered along with a route to it. The discovered service information is communicated back to the application layer where the service is accessed.

The process of service discovery which is integrated with the AODV protocol is secured by the proposed system by establishing a safe path by selecting trust worthy nodes at each hop. In figure 1 the nodes that are shaded indicate nodes with low trust and the nodes without the shade indicate nodes with high trust. Therefore the safe path is shown through nodes with high trust. The process of trust computation and the selection of safe path is outline in the next subsection.

3.1. Trust Based Security Scheme

On receiving a packet, a node either consumes the packet or forwards the packet or drops the packet as shown in Figure 2. If the packet is destined to the same node it processes the packet and consumes it. If the packet is destined to another node the packet is forwarded or broadcasted. The packet may be dropped if it received the packet from a malicious node. All these action can be marked as valid. But if a node drops a legitimate packet the action is misbehaviour and has to be isolated. Generation and broadcast of packet above the allowed limit is also treated as misbehaviour.

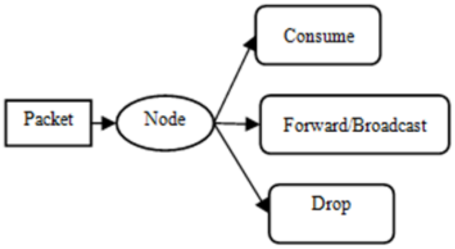


Figure 2 Packet Processing

As a part of the normal operation every node is processing the control packets received from its neighbours and based on the type of messages appropriate actions are taken. The main three control packets received are the request message (SREQ), the Reply message (SREP), Hello message and the RERR message. Based on the control packets received the trust based scheme monitors it's one hop neighbour list and assigns trust values in accordance with their behaviours.

To monitor and assign trust values, two new fields are introduced for each node in the neighbour cache. The first one is the "Trust" field that defines the level of trust and the second field is called "SREQ_Count" that keeps a count of the number of SREQ messages originated by a 1- hop node. The initial value of Trust=1 and the initial value of SREQ_Count=0.

The Trust value is defined at four levels [0-3] which indicate no trust, low trust, moderate trust and high trust respectively. The computation of trust, monitoring of nodes and avoidance of malicious nodes is guided by the technique outlined in the Algorithm 1 *Detecting and Avoiding Misbehaving nodes*.

The Hello messages are periodically broadcasted by all 1- hop neighbours at regular interval. These Hello message also include the service advertisements. When a Hello message is received from a neighbour initially it is assumed that the nodes is trustworthy but with a low level of trust i.e. Whenever a new neighbour is added to the list the Trust value is initialized as one. Every time the node cooperates by forwarding its neighbours packet the trust value is incremented by one up to a maximum value of 3 which defines the highest level of trust.

If a node does not forward any packets its Trust value continues to remain constantly at one. When the Trust value of a node is one, other nodes will cooperate with the node by forwarding the packets, but that particular node is never included as a precursor in the path between the server and the consumer. A node is selected as a precursor and is added in a reverse route so that it becomes a part of the final route only if its trust value is greater than one that is either two or three.

If a node is observed as flooding the network the Trust value is reduced to zero irrespective of the level it currently belongs to. And if the Trust value of a node is zero the node is termed misbehaving node, added to “blacklist” set for PATH_DISCOVERY_TIME and is isolated from the network by dropping all the packets received from the node. [31] PATH_DISCOVERY_TIME is a default value defined in the AODV protocol which is equal to twice the NET_TRAVERSAL_TIME.

The Trust value of a node is increased only if it is one or two. If the Trust value is zero it is not incremented even if it cooperates in forwarding. The neighbour cache is cleared periodically when all the values are reset and the nodes get a chance to behave responsibly by forwarding packets of its neighbour nodes and become trusted nodes. The information stored at the neighbour cache is summarized in Table 2.

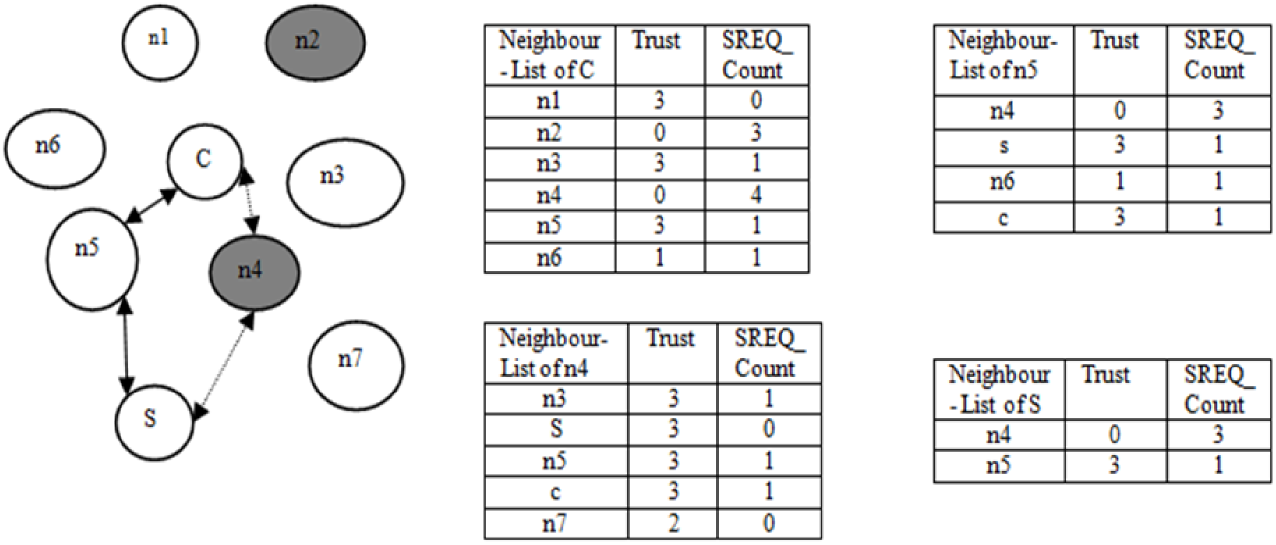
Destination IP	IP address of the sender node
Destination Sequence number	The sender node’s latest sequence number
Hop count	0
Lifetime	ALLOWED_HELLO_LOSS*HELLO INTERVAL
Srv_id	Service provided by the node
Trust	1(Initially then dynamically updated) [0-4]
SREQ_Count	0(Initially then dynamically updated)

Table 2 Fields of Neighbour Cache

```
Trust is a variable that defines the trust level of 1-hop node
Node A is under consideration and is receiving packets
Let N1 symbolize the set of all 1- hop nodes
SREQ_Count is a variable that keeps the count of the number of SREQ packets generated by the set N1
    1.  Initialize Trust= 1 for all 1-hop neighbours. // Initially assuming all nodes are having a low  trust
    2.  Initialize SREQ_Count=0 for all 1- hop neighbours
    3.  If node A receives SREQ message from a node n in N1
        a.  If Trust=0 for node n, discard packet
        b.  If SREQ message originated by a node n  and if SREQ_Count_ > RREQ_RATELIMIT
              Set Trust of node n to zero
              Delete Node n from 1- hop list
              Send RERR messages to nodes in      precursor list
              Exit
        c.  If message forwarded by a node n is the SREQ message originated by node A
                 If Trust <3 and Trust> 0      then Trust++ for node n    Discard packet;   // duplicate packet but
                                                                                           //n   is   cooperating
            node
             Exit;
        d.  If destination in SREQ message is A
             If Trust <3 and Trust> 0   then Trust++ for node n
```

```
                    Set reverse route if trust
                    Send SREP through reverse route
        e.          If node n  forwards SREQ message of other nodes      // n is cooperating node
                      If Trust <3 and Trust >0 , then Trust++ for n;
                        Set reverse route if trust > 1        // include only trust worthy node in safe path
                      Forward packet
4.      if node A receives SREP message from a node n in N1
        a.          If Destination is node A
                    accept packet
                    If Trust <3 and Trust >0     then Trust++ for n;   // n is cooperating node
        b.          if SREP message received is forwarded by A
                   If Trust <3and Trust >0   then Trust ++;
                   Discard packet;          // duplicate packet shows that n is cooperative
                     Exit
        c.          else if trust > 0 forward packet        // forward data of only cooperating nodes gives opportunity to
                                                            //nodes who is a new neighbour
        Else discard packet
5.      If node receives HELLO message update 1- hop neighbour list N1.
        Set Trust=1for all nodes in N1
6.      Clear  cache periodically
```

Algorithm 1 Detecting and Avoiding Misbehaving Nodes



Neighbour -List of C	Trust	SREQ_ Count
n1	3	0
n2	0	3
n3	3	1
n4	0	4
n5	3	1
n6	1	1

Neighbour- List of n5	Trust	SREQ_ Count
n4	0	3
s	3	1
n6	1	1
c	3	1

Neighbour- List of n4	Trust	SREQ_ Count
n3	3	1
S	3	0
n5	3	1
c	3	1
n7	2	0

Neighbour -List of S	Trust	SREQ_ Count
n4	0	3
n5	3	1

Figure 3 Illustration of Safe Path Selection between Nodes C (Consumer) and S (Server)

3.1.1. Illustration of Algorithm

Consider the set of nodes as shown in Figure 3. We assume bidirectional communication exist between all these nodes. Each of these nodes computes trust value for each other based on the Algorithm Detecting and Avoiding Misbehaving nodes. For example Node S will maintain the trust values of all its six one-hop nodes (n1-n6). Initially the trust values of all these six nodes will be one defining a low level of trust.

For illustration the "Trust "value of node n4 is shown as zero since the SREQ_Count=3. If SREQ_Count is greater than the RREQ_RATELIMIT the trust is reduced to zero The Trust value is not gradually reduced here since this misbehaviour is considered as severe. The node n6 has generated only one SREQ message which is within the limit but its trust value is still at 1 probably because node n6 has not forwarded packets and so cannot be a part of safe path.

Suppose Node C and Node S is consumer and server node respectively. Suppose Node C wishes to access the service from Node S. Node C does not have a valid route and so want to discover a safe path. Therefore Node C initiates a SREQ message and broadcast it to its one hop neighbour. The one hop neighbour of node C broadcast it to the next hop. Now the SREQ message is received by node S from node n5 and n4. Node S which is the destination node would send a SREP message through n5 as the trust value of node n4 is 0 and the node will be ignored during the current PATH_DISCOVERY. Thus the safe path is shown in solid line from C-> n5-> S in Figure 3.

3.1.2. Limiting Packer Dropping Attack

Step 3 and 4 of the Algorithm Detecting and Avoiding Misbehaving nodes is monitoring and rewarding the one hop neighbour who is forwarding the packets. So when a route is discovered a safe path will be selected by including nodes from the one- hop list that have highest trust, i.e. a node is included in the reverse route only if trust >1. This process is repeated by every node in the path as shown in Figure 3. Thus a safe path is established between the source and destination. When a SREQ arrives through multiple nodes a node which has the highest trust is chosen at each next hop in the path.

If the intermediate nodes replies to request it may so happen that the destination node is not aware of the routes to the source and may initiate route discovery again. Or we should include mechanism to inform the destination node about the reply sent. This also gives an opportunity to the malicious nodes to capture the routes by sending reply faster with higher sequence numbers.

Therefore if a restriction is imposed such that only the destination will send reply, then the intermediate nodes can be restricted from sending spurious replies. Also for the process of service discovery the metrics of the server assist in the selection of optimal servers, so it is best to retrieve this information from the destination nodes itself. This might induce some delay in the reception of the reply message but the path established will be more secure and avoid route discovery process towards the source by destination node.

3.1.3. Limiting Flooding Attack

Flooding attack is launched by a node by generating repeated SREQ messages to the server with the intention to bring down the server and degrade the performance of the network. After an SREQ message is generated the node waits for the NET_TRAVERSAL_TIME milliseconds before the second request is generated with an updated RREQ_ID. [24]The number of Retries allowed is given by RREQ_RETRIES whose default value is 2. In the protocol this condition is enforced at the sender before the generation of new SREQ messages. But a misbehaving node will violate the said condition and generate more than the allowed number of SREQ messages to degrade the performance of the network. Therefore this condition has to be verified at the receiver end also to identify misbehaving nodes. In the Algorithm 1 Detecting and Avoiding Misbehaving nodes step 3.b monitors the neighbouring node and limits the node from flooding the network with SREQ messages. If the number of SREQ messages generated are greater than RREQ_RATELIMIT the trust of the node is reduced to zero. Any node whose trust is zero is silently ignored. The proposed system is implemented with the help of a simulator to evaluate its efficacy.

4. RESULTS AND PERFORMANCE EVALUATION

The network simulator ns2.35 is chosen to simulate the proposal. Most of the existing system is implemented with this proposal and they suggest it to be effective. The AODV protocol available within the simulator is modified to integrate service discovery and the proposed Algorithm 1.

4.1. Integrating Secured Service Discovery in AODV

The proposed secured service discovery is called AODV-SSD. The request and reply messages are extended to include service discovery information. Routing table is extended to include service information. In the neighbour cache for each node the fields Trust and SREQ_Count is added. The receive routines present in the protocol was modified to implement the *Algorithm 1 Detecting and Avoiding Misbehaving node*s. New routines were added to the neighbour management to update Trust values and SREQ_Count. After having made the said changes to the AODV protocol in the back end we now discuss the simulation environment.

4.2. Simulation Environment

Three simulation environments with 10 nodes, 30 nodes and 50 nodes were built. The simulation parameters are summarized in Table 3. Initially the node were arranged close to each other as a grid and later the nodes were made mobile to move away from each other.

The attacker nodes were marked and the routines were modified to implement attacker to drop and flood the packets. At the front end initially TCL scripts were written with 10 nodes 3 attacker nodes and one request was introduced. The simulation was run and the trace file thus obtained was analysed to design the awk scripts that could extract the desired information for analysis. Further To evaluate the implementation Tcl scripts were built with 30 nodes and 50 nodes.

In the simulation scenario with 30 nodes, 5 attacker nodes were introduced, out of which 3 nodes implemented dropping attack and two nodes implemented flooding attack. Four requests were generated in two slots with a total of eight requests

In the second scenario with 50 nodes, 8 attacker nodes were introduced out of which 5 nodes implemented dropping attack and 3 nodes implemented flooding attack. 3 slots were identified with 6 service request generated in 3 slots, thus a total of 18 requests were generated.

To evaluate the proposed system we had to compare and analyse packet dropping attack and flooding attack. The modifications of the AODV in the existing system does not control both packet dropping and flooding attack simultaneously, therefore two existing system are chosen one which prevents flooding attack [18] known as AIF_AODV and the other that prevents packet dropping attack [24] known as SNRRM . The existing system chosen are recent closet to the proposal and has minimal overhead.

Parameters	Values
Channel	Wireless Channel
Mac	802_11
Propagation Model	Two Way Ground
Queue	DropTail/PriQueue
Antenna	OmniAntenna
Routing Protocol	AODV
Number of Nodes	30/50
Grid x,y	1000X1000
Simulation time	30s
Packet size	100
Application	ftp

Table 3 Simulation Parameters

The simulation environment was run repeatedly with the proposed modification to AODV protocol and the AODV protocol with the methodology specified in the existing system. The trace files were obtained from all the scenarios and data was extracted with awk scripts. The results obtained are presented, compared and analysed in the next section.

4.3. Evaluation Parameters

To evaluate the implementation four parameters were considered the control message overhead, the storage overhead, the service discovery latency (SDL) and service discovery ratio (SDR) [30]. Control message overhead kept track of the number of control messages generated per slot in each of the methodologies evaluated. The storage overhead evaluated the additional storage incurred in implementing the secured methodology. The SDL is the time required to receive the response after the request is generated. The service discovery Ratio is the ratio of the number of response obtained to the number of request generated in percentile as shown in equation 1 and 2 respectively per slot.

SDL= Response time- Request time --------- (1)

$$\text{SDR}=\frac{\text{Total number of response recieved per slot}}{\text{Total number of requests generated per slot}} \text{ ---- (2)}$$

Equation (2) is represented in percentage

4.4. Results

For ease of representation we use the abbreviation of the different system under consideration. The existing service discovery system which is operating in the ideal environment without malicious node is called AODV-SD this system serves as the upper bench mark. The same service discovery system which is operating in the presence of malicious nodes without applying any protection mechanism is called AODV-MSD. This system serves as the lower benchmark. The proposed Secured service discovery is called AODV- SSD. Two existing system under consideration is the AIF_AODV [18] which proposes a technique to avoid flooding attack and SNRRM[24] proposes a technique to avoid Blackhole or packet dropping attack. So we extract data for all these five techniques, compare and analyze them.

4.4.1. Performance Evaluation of Control Message overhead

The control messages generated specific to the AODV protocol for the process of service discovery was extracted for all the five proposals and plotted as a graph in figure 4 and figure 5 for 30 and 50 node environments respectively.

Analysis: It can be observed from figure 4 and figure 5 that the control messages generated in the proposed and the existing system lies in-between the upper and lower benchmark. It can also be observed that the proposed system has generated fewer control messages as compared to the existing system. When the existing systems are compared the

AIF_AODV has generated lesser control messages as it controls flooding of the SREQ messages. But the control messages generated are higher than the proposed system due to the fact that it does not control the black hole attack which hinders the service discovery and the service request messages are regenerated. The proposed AODV-SSD generated 4% less control message when compared to AIF_AODV and 16% less control message when compared to SNRRM.

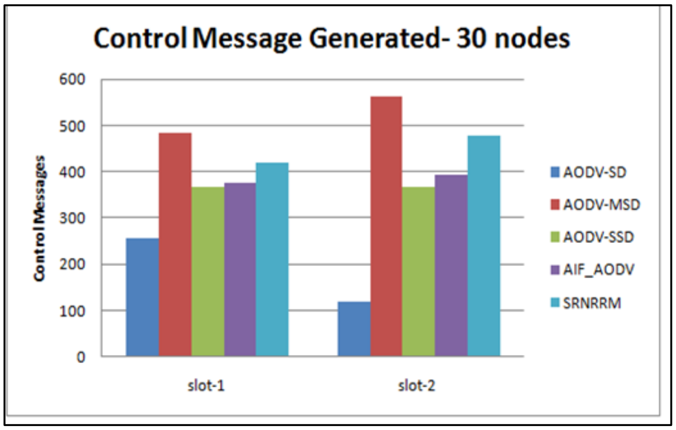


Figure 4 Control Message Generated for 30 Nodes

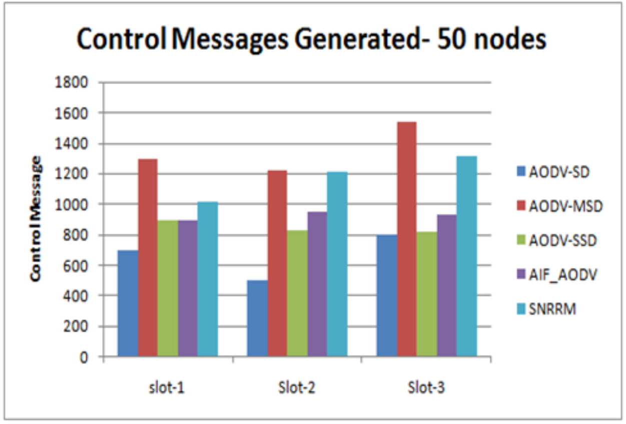


Figure 5 Control Message Generated for 50 Nodes

4.4.2. Performance Evaluation of Storage Overhead

The storage overhead required to implement the protection mechanism against malicious nodes is compared here. In the proposed system, two new variables are introduced one to track the trust and the other to count the request message generated. If we consider a minima 1 byte per variable we require 2 byte for every one hop node. If we roughly consider the total number of one hop node= n. We use the word roughly since the number of one hop node is dynamic.

Storage overhead (AODV_SSD) = 2*n

The AIF_AODV also uses a variable to count the number of request therefore

Storage overhead (AIF_AODV) = n

The SNRRM technique keeps track of the number of packets sent, the number of packets received to compute communication reliability in a trust variable. It also considers the residual energy so again considering one byte per variable

Storage overhead (SNRRM) = 4*n

Therefore it can be observed that the storage overhead of the proposed AODV-SSD is in between that of the two compared existing system. It has a lower storage overhead than the SNRRM technique but higher storage overhead than AIF_AODV. But considering the advantage of preventing the packet dropping and the flooding attack the storage overhead can be neglected..

4.4.3. Performance Evaluation of Service Discovery Latency

The response time data extracted from the 30 node simulation environment and the corresponding service discovery latency value is computed and is tabulated in Table 4. The data corresponding to 50 node simulation environment is plotted in Table 5. The cell marked with – indicate that the responses were not obtained for the corresponding request. The average service discovery latency per slot for 30 node environment and 50 node environments for all five techniques are plotted in figure 6 and figure 7 respectively.

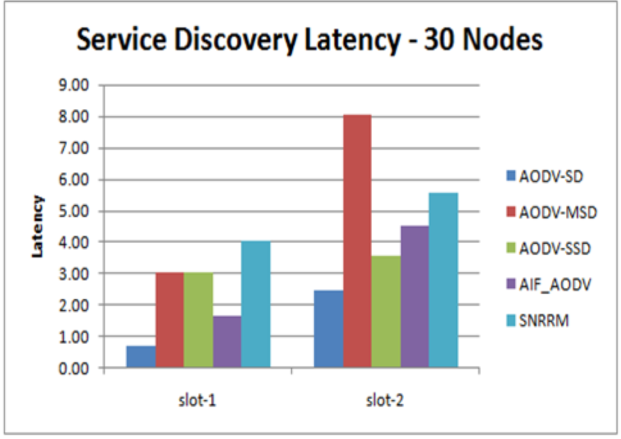


Figure 6 Service Discovery Latency 30 Nodes

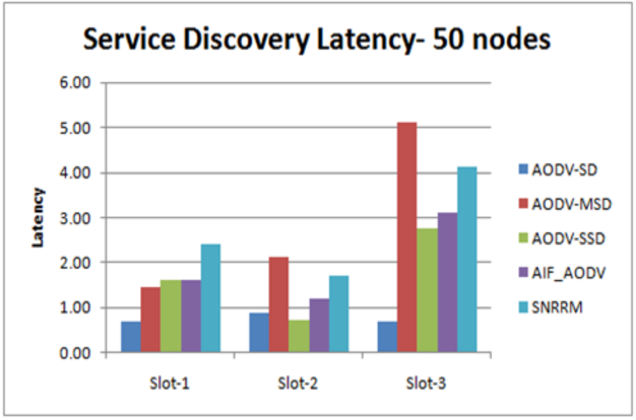


Figure 7 Service Discovery Latency 50 Nodes

Request #	Start Time	AODV-SD		AODV-MSD		AODV-SSD		AIF_AODV		SNRRM	
		End Time	Latency	End Time	Latency	End Time	Latency	End Time	Latency	End Time	Latency
1.	1	1.10	0.10	-	-	-	-	-	-		
2.	1	3.07	2.07	7.05	6.05	7.05	6.05	4.33	3.33	8.07	7.07
3.	1	1.05	0.05	1.04	0.04	1.04	0.04	1.04	0.04	2.05	1.05
4.	1	1.54	0.54	-	-	-	-	-	-	-	-
5.	15	20.52	5.52	21.06	6.06	21.06	6.06	21.01	6.01	22.32	7.32
6.	15	17.10	2.10	25.04	10.04	15.05	0.05	-	-	17.33	2.33
7.	15	17.06	2.06	-	-	17.06	2.06	18.05	3.05	-	-
8.	15	15.14	0.14	-	-	21.05	6.05	-	-	22.06	7.06
		Average Latency	1.57	Average Latency	5.55	Average Latency	3.39	-	3.11	-	4.97

Table 4 Response Time and Service Discovery Latency 30 Nodes

Request #	Start Time	AODV-SD		AODV-MSD		AODV-SSD		AIF_AODV		SNRRM	
		End Time	Latency	End Time	End Time	Latency	Latency	End Time	Latency	End Time	Latency
1.	1	1.21	0.21	3.09	2.09	3.09	2.09	3.09	2.09	4.22	3.22
2.	1	1.25	0.25	3.15	2.15	3.15	2.15	3.15	2.15	3.73	2.73
3.	1	1.25	0.25	1.14	0.14	1.14	0.14	1.14	0.14	2.23	1.23
4.	1	3.13	2.13	-	-	3.13	2.13	3.13	2.13	3.45	2.45
5.	1	-	-	-	-	-	-				
6.	1	-	-	-	-	-	-				
7.	10	10.20	0.20	-	-	12.64	2.64	12.64	2.64	13.02	3.02
8.	10	10.68	0.68	-	-	10.15	0.15			11.07	1.07
9.	10	10.73	0.73	-	-	10.24	0.23	10.24	0.24	11.33	1.33
10.	10	10.58	0.58	-	-	10.46	0.46	10.77	0.77	12.01	2.01
11.	10	11.07	1.07	-	-	No Res					
12.	10	12.14	2.14	12.14	2.14	10.19	0.19			11.07	1.07
13.	20	20.23	0.23	-	-	21.10	1.10	21.1	1.1		
14.	20	-	-	28.11	8.11	26.11	6.11	26.11	6.11		
15.	20	20.17	0.17	-	-	-	-				
16.	20	21.68	1.68	-	-	21.68	1.68				
17.	20	20.70	0.70	22.13	2.13	22.13	2.13	22.13	2.13		
18.	20	20.66	0.66	-	-	-	-				
		Average Latency	0.78	Average Latency	2.79	Average Latency	1.63	Average Latency	1.95	Average Latency	2.41

Table 5 Response Time and Service Discovery Latency 50 Nodes

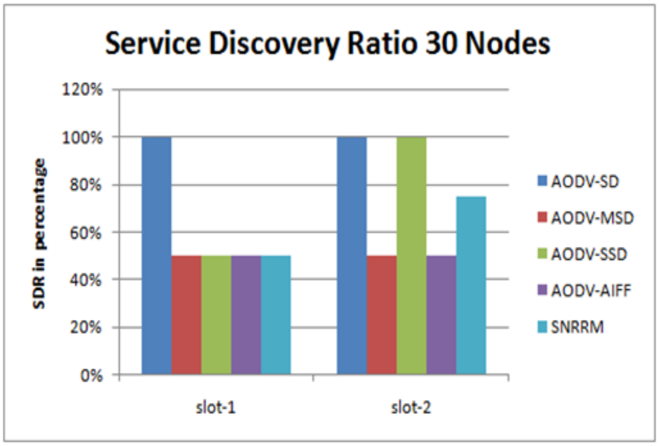


Figure 8 Service Discovery Ratio 30 Nodes

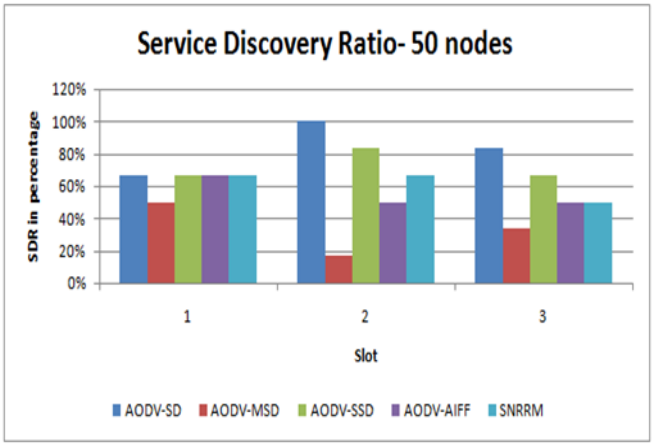


Figure 9 Service Discovery Ratio 50 Nodes

Analysis and evaluation: If the graphs in figure 6 and figure 7 is analyzed it can be observed that the proposed system and the existing system has a service discover latency lesser than the lower bench mark and more than the upper bench mark. When the proposed system is compared with the existing system, in slot-1 the service discovery latency of the existing and proposed system is closer to the lower benchmark since the protection techniques need a warm up time to collect data and identify malicious node. As the system matures and progresses towards slot-2 and slot 3 the service discovery latency of the proposed system is closer to the upper benchmark and lesser than both the existing system as it is able to control the threats.

The latency of the proposed AODV-SSD is higher than the existing system AIF_AODV in slot-1. This could be because of the fact that the proposed AODV-SSD restricts the sending of SREP message only by destination node so that a safe path is established from end to end. As the system progresses the latency of proposed system is lesser since the overhead due to threats is reduced and this overcomes the delay incurred in restricting the destination to reply. When the two existing systems are compared AODV_AIF has a lesser latency than SNRRM as it has lower hop by hop computation overhead. In the SNRRM technique it identifies reliable node by observing the communication pattern and residual energy which involves computation. Thus it can be observed that the proposed AODV-SSD has a lower latency when compared to existing system as it is able to control both the attacks.

4.4.4. Performance Evaluation of Service Discovery Ratio

Service discovery ratio computed as per equation (2) for simulation scenario with 30 and 50 nodes for the proposed and existing system are plotted as a graph in figure 8 and figure 9 respectively. Analysis and Evaluation: Similar to service discovery latency the performance of the existing and proposed system is closer to the lower benchmark in slot-1. As time progress towards slot 2 in 30 node scenarios and slot 2 and 3 in 50 node scenario the performance of the proposed system is approaching closer to the upper benchmark as it is able to control the activities of the malicious node. SDR ratio is higher than both the existing system. The SDR ratio of the proposed AODV-SSD system is 25% higher than the AIF_AODV and 13% higher than SNRRM.

5. CONCLUSION

The main objective of this paper is to secure the process of service discovery from Denial of Service attack such as packet dropping attack and flooding attack, when implemented at the network layer. The AODV protocol was chosen and modified such that every node in the network will always be aware of the list of trust worthy 1- hop nodes at all times and communicates with only such nodes. We believe that since each node is interacting with its trusted 1- Hop neighbour, and all the misbehaving nodes are ignored at the first hop, the network as a whole is able to show a better performance in terms control message overhead, service discovery latency and service discovery ratio. As a future work the experiment can be implemented in a real time environment and also work can be carried out to integrate the control of more threats.

REFERENCES

[1] Alex Varshavsky, Bradley Reid, Eyal de Lara walex,brad A Cross-Layer Approach to Service Discovery and Selection in MANETs, Department of Computer Science University of Toronto 2005 IEEE.

[2] P.-W. Yau and C. J. Mitchell, “Security vulnerabilities in ad hoc networks,” in Proceedings of the 7th International Symposium on Communication Theory and Applications, 2003, pp. 99–104

[3] A. Nadeem and M. P. Howarth, "A Survey of MANET Intrusion Detection & Prevention Approaches for Network Layer Attacks," in IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2027-2045, Fourth Quarter 2013, doi: 10.1109/SURV.2013.030713.00201.

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

[5] S. Kurosawa and A. Jamalipour, “Detecting Blackhole Attack on AODVbased Mobile Ad Hoc Networks by Dynamic Learning method”, International Journal of Network Security, Vol.5, No.3, pp 338-345, November2007.

[6] J.Sen, M.Chandra, S.G. Harihara, H.Reddy and P.Balamuralidhar, “A Mechanism for Detection of Gray Hole Attacks in Mobile Ad Hoc Networks”, Proc. IEEE International Conference on Information Communication and Signal Processing ICICS, Singapore, Dec. 2007

[7] L. Tamilselvan and V. Sankaranarayanan, "Prevention of Blackhole Attack in MANET," The 2nd International Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless 2007), Sydney, NSW, Australia, 2007, pp. 21-21, doi: 10.1109/AUSWIRELESS.2007.61.

[8] Payal N. Raj and Prashant B. Swadas, “DPRAODV: A DYANAMIC LEARNING SYSTEM AGAINST BLACKHOLE ATTACK IN AODV BASED MANET “,International Journal of Computer Science Issues, Vol. 2, 2009,PP 54-59

[9] V. Mohite and L. Ragha, "Cooperative Security Agents for MANET," 2012 World Congress on Information and Communication Technologies, Trivandrum, 2012, pp. 549-554, doi: 10.1109/WICT.2012.6409138.

[10] K. S. Chavda and A. V. Nimavat, "Removal of black hole attack in AODV routing protocol of MANET," 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 2013, pp. 1-5, doi: 10.1109/ICCCNT.2013.6726832.

[11] T. Varshney, T. Sharma and P. Sharma, "Implementation of Watchdog Protocol with AODV in Mobile Ad Hoc Network," 2014 Fourth International Conference on Communication Systems and Network Technologies, Bhopal, India, 2014, pp. 217-221, doi: 10.1109/CSNT.2014.50.

[12] T. Shu and M. Krunz, "Privacy-Preserving and Truthful Detection of Packet Dropping Attacks in Wireless Ad Hoc Networks," in IEEE Transactions on Mobile Computing, vol. 14, no. 4, pp. 813-828, 1 April 2015, doi: 10.1109/TMC.2014.2330818.

[13] Vimal Kumar, Rakesh Kumar, An Adaptive Approach for Detection of Blackhole Attack in Mobile Ad hoc Network, Procedia Computer Science, Volume 48, 2015, Pages 472-479,ISSN 1877-0509, https://doi.org/10.1016/j.procs.2015.04.122.

[14] S. Jain and A. Khuteta, "Detecting and overcoming blackhole attack in mobile Adhoc Network," 2015 International Conference on Green

Computing and Internet of Things (ICGCIoT), Noida, 2015, pp. 225-229, doi: 10.1109/ICGCIoT.2015.7380462.

[15] Adwan Yasin, Mahmoud Abu Zant, "Detecting and Isolating Black-Hole Attacks in MANET Using Timer Based Baited Technique", Wireless Communications and Mobile Computing, vol. 2018, Article ID 9812135, 10 pages, 2018. https://doi.org/10.1155/2018/9812135

[16] Musale S.S., Dhende S.L., Shirbahadurkar S.D., Najan A.S. (2019) Gray Hole and Cooperative Attack Prevention Protocol for MANETs. In: Abraham A., Dutta P., Mandal J., Bhattacharya A., Dutta S. (eds) Emerging Technologies in Data Mining and Information Security. Advances in Intelligent Systems and Computing, vol 814. Springer, Singapore. https://doi.org/10.1007/978-981-13-1501-5_49

[17] Ali H. Ahmed, Nagwa M. Omar, and Hosny M. Ibrahim, "Secured Service Discovery Technique in IoT," Journal of Communications, vol. 14, no. 1, pp. 40-46, 2019. Doi: 10.12720/jcm.14.1.40-46.

[18] Mahmoud Abu Zant, Adwan Yasin, "Avoiding and Isolating Flooding Attack by Enhancing AODV MANET Protocol (AIF_AODV)", Security and Communication Networks, vol. 2019, Article ID 8249108, 12 pages, 2019. https://doi.org/10.1155/2019/8249108.

[19] Hwanseok Yang, "A Study on Improving Secure Routing Performance Using Trust Model in MANET", Mobile Information Systems, vol. 2020, Article ID 8819587, 17 pages, 2020. https:// doi.org/10.1155/2020/8819587.

[20] Tripathy, B.K., Jena, S.K., Bera, P. et al. An Adaptive Secure and Efficient Routing Protocol for Mobile Ad Hoc Networks. Wireless Pers Commun 114, 1339–1370 (2020). https://doi.org/10.1007/s11277-020-07423-x

[21] Ran, C., Yan, S., Huang, L. et al. An improved AODV routing security algorithm based on blockchain technology in ad hoc network. J Wireless Com Network 2021, 52 (2021). https://doi.org/10.1186/s13638-021-01938-y

[22] A. M. El-Semary and H. Diab, "BP-AODV: Blackhole Protected AODV Routing Protocol for MANETs Based on Chaotic Map," in IEEE Access, vol. 7, pp. 95197-95211, 2019, doi: 10.1109/ACCESS.2019.2928804.

[23] Y. Fu, G. Li, A. Mohammed, Z. Yan, J. Cao and H. Li, "A Study and Enhancement to the Security of MANET AODV Protocol Against Black Hole Attacks," 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2019, pp. 1431-1436, doi: 10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00259.

[24] Muruganantham Ponnusamy, Dr. A. Senthilkumar, and Dr.R.Manikandan, “Detection of Selfish Nodes Through Reputation Model In Mobile Adhoc Network MANET’ Turkish Journal of Computer and Mathematics Education, Vol.12 No.9 (2021), 2404–2410.

[25] Alaa Althalji, Souheil Khawatmi, Mohamed Khatib Improving the Security of AODV Protocol using V-Detector Algorithm, International Journal of Computer Applications (0975 – 8887) Volume 177 – No. 17, November 2019

[26] Ankit Kumar & Madhavi Sinha (2019) Design and analysis of an improved AODV protocol for black hole and flooding attack in vehicular ad-hoc network (VANET), Journal of Discrete Mathematical Sciences and Cryptography, 22:4, 453-463, DOI: 10.1080/09720529.2019.1637151

[27] Md Ibrahim Talukdar, Rosilah Hassan, Md Sharif Hossen, Khaleel Ahmad, Faizan Qamar, Amjed Sid Ahmed, "Performance Improvements of AODV by Black Hole Attack Detection Using IDS and Digital Signature", Wireless Communications and Mobile Computing, vol. 2021, 13 pages, 2021. https://doi.org/10.1155/2021/6 693316.

[28] N. Islam and Z. A. Shaikh, "A Novel Approach to Service Discovery in Mobile Adhoc Network," 2008 IEEE International Networking and Communications Conference, 2008, pp. 58-62, doi: 10.1109/INCC.2008.4562692.

[29] Smitha Kurian, Loganathan R,International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-4, November 2019

[30] Kurian, S., Ramasamy, L. Novel AODV based service discovery protocol for MANETS. Wireless Netw 27, 2497–2508 (2021). https://doi.org/10.1007/s11276-021-02596-0.

[31] https://tools.ietf.org/html/rfc3561 accessed last on August 2021.

Authors

Smitha Kurian: Received B.E. in CSE from SJMIT, Kuvempu University, M.Tech from VTU and is a Research Scholar at HKBK College of Engineering – CSE Research Centre affiliated to VTU. Worked as Assistant Professor in HKBKCE, Department of CSE for 15 years. Published various journal in conferences and journals in the field of Wireless Ad Hoc networks, Artificial Intelligence and Machine Learning.

Dr. Loganathan R. Received B.E in CSE from Bharathiar University, Coimbatore, M.TECH IN CSE from Visvesvaraya Technological University, Karnataka. PhD in CSE from from Sathyabama University, Chennai. He is now working as Professor and Head of Department of CSE, HKBK College of Engineering, Bangalore. He has 20 years of Academic experience. He published his research work in several reputed journals and conferences in the field of Image Processing, Wireless Networks and machine Learning.

How to cite this article:

Smitha Kurian, Loganathan Ramasamy, “Securing Service Discovery from Denial of Service Attack in Mobile Ad Hoc Network (MANET)”, International Journal of Computer Networks and Applications (IJCNA), 8(5), PP: 619-633, 2021, DOI: 10.22247/ijcna/2021/209992.