International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Design and Development of Anonymous Zone Based Partitioning and Routing Protocol in MANETS (AZPR)

Ayeesha Siddiqha[1], Arshad Khan [2]

[1]Assistant Professor, Malnad College of Engineering, Hassan, India

[2]Mtech (Computer Science & Engineering), BIT, Banglore, India

Abstract: ***Mobile Ad Hoc Networks (MANETs) feature of self-organizing and independent infrastructures which makes mobile ad hoc networks to be used for information sharing and communication. MANET use anonymous routing protocols that hide node identities and/or routes from outside observers in order to provide anonymity protection. Anonymous routing protocols relying on either hop-by-hop encryption or redundant traffic either generates high cost or cannot provide full anonymity protection to source, destination, and route. The high cost introduces the inherent resource constraint problem in MANETs especially in multimedia wireless applications. To offer high anonymity protection at a low cost; this paper proposes An Anonymous Zone-Based Partitioning and Routing Protocol in MANET (AZPR). AZPR dynamically partitions the network field into zones and randomly chooses nodes in zones as relay nodes, which form an anonymous route which is untraceable. AZPR hides the data initiator/receiver among many initiators/receivers to strengthen source and destination anonymity protection. Hence it offers anonymity protection to source, destination, and route. It also provides some strategies to effectively prevent intersection and timing attacks.***

Keywords: Anonymity, Mobile ad hoc networks, Routing protocol, Geographical routing.

1. Introduction

A mobile ad hoc network (MANET) is a self-configuring infrastructure less network of mobile devices connected by wireless. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router. The primary challenge in building a MANET is equipping each device to continuously maintain the information required to properly route traffic. MANETs are a kind of Wireless and ad hoc network that usually has a routable networking environment on top of a Link Layer ad hoc network. Nodes in MANETs are vulnerable to malicious entities that aim to tamper and analyze data and traffic analysis by communication eavesdropping or attacking routing protocols.

Anonymity is critical in military applications (e.g., soldier communication). Consider a MANET deployed in battlefield. Through traffic analysis, enemies may intercept transmitted packets, track our soldiers (i.e., nodes), attack the commander nodes, and block the data transmission by comprising relay nodes (RN), thus putting us at a tactical disadvantage. Anonymous routing protocols are crucial in MANETs to provide secure communications by hiding node identities and preventing traffic analysis attacks from outside observers.

Anonymity in MANETs includes identity and location anonymity of data sources (i.e., senders) and destinations (i.e., recipients), as well as route anonymity. “Identity and location anonymity of sources and destinations” means it is hard if possible for other nodes to obtain the real identities and exact locations of the sources and destinations. For route anonymity, adversaries, either en route or out of the route, cannot trace a packet flow back to its source or destination, and no node has information about the real identities and locations of intermediate nodes en route. Also, in order to dissociate the relationship between source and destination (i.e., relationship un-observability); it is important to form an anonymous path between the two endpoints and ensure that nodes en route do not know where the endpoints are, especially in MANETs where location devices may be equipped.

Existing anonymity routing protocols in MANETs can be mainly classified into two categories: hop-by-hop encryption [2], [3], [4], [1], [2] and redundant traffic [3], [4], [5], [10], [6], [12], [7]. Most of the current approaches are limited by focusing on enforcing anonymity at a heavy cost to precious resources because public-key-based encryption and high traffic generate significantly high cost. In addition, many approaches cannot provide all of the aforementioned anonymity protections. For example, ALARM [1] cannot protect the location anonymity of source and destination, SDDR [8] cannot provide route anonymity, and ZAP [7] only focuses on destination anonymity.

In order to provide high anonymity protection (for sources, destination, and route) with low cost, we propose An Anonymous Zone-Based Partitioning and Routing Protocol (AZPR). AZPR dynamically partitions a network field into zones and randomly chooses nodes in zones as intermediate relay nodes, which form a non traceable anonymous route. Specifically, in each routing step, a data sender or forwarder partitions the network field in order to separate itself and the destination into two zones. It then randomly chooses a node in the other zone as the next relay node and uses the GPSR algorithm to send the data to the relay node. In the last step,

the data is broadcasted to k nodes in the destination zone, providing k-anonymity to the destination. In addition, AZPR has a strategy to hide the data initiator among a number of initiators to strengthen the anonymity protection of the source. AZPR is also resilient to intersection attacks and timing attacks. In summary, the contribution of this work includes:

[1] To strengthen the anonymity protection of source and destination by hiding the data initiator/receiver among a number of data initiators/receiver.
[2] To provide route anonymity and identity.
[3] To provide resiliency to intersection attacks and timing attacks.
[4] To reduce the cost of providing anonymity

2. An Anonymous Zone- Based Partitioning and Routing Protocol in MANET

2.1 Network Model and Assumptions

AZPR can be applied to different network models with various node movement patterns such as random way point model and group mobility model. Consider a MANET deployed in a large field where geographic routing is used for node communication in order to reduce the communication latency. The location of a message's sender may be revealed by merely exposing the transmission direction. Therefore, an anonymous communication protocol that can provide untraceability is needed to strictly ensure the anonymity of the sender when the sender communicates with the other side of the field. Moreover, a malicious observer may try to block the data packets by compromising a number of nodes, intercept the packets on a number of nodes, or even trace back to the sender by detecting the data transmission direction. Therefore, the route should also be undetectable. A malicious observer may also try to detect destination nodes through traffic analysis by launching an intersection attack. Therefore, the destination node also needs the protection of anonymity. In this work, the attackers can be battery powered nodes that passively receive network packets and detect activities in their vicinity. They can also be powerful nodes that pretend to be legitimate nodes and inject packets to the network according to the analytical results from their eavesdropped packets. The assumptions below apply to both inside and outside attackers.

1. Capabilities- By eavesdropping, the adversary nodes can analyze any routing protocol and obtain information about the communication packets in their vicinity and positions of other nodes in the network. They can also monitor data transmission when a node is communicating with other nodes and record the historical communication of nodes. They can intrude on some specific vulnerable nodes to control their behavior.
2. Incapabilities- The attackers do not issue strong active attacks such as black hole. They can only perform intrusion to a proportion of all nodes. Their computing resources are not unlimited; thus, both symmetric and public/private key cannot be brutally decrypted within a reasonable time period. Therefore, encrypted data are secure to certain degree when the key is not known to the attackers.

2.2 Dynamic Pseudonym and Location Service

In one interaction of node communication, a source node S sends a request to a destination node D and the destination responds with data. A transmission session is the time period that S and D interact with each other continuously until they stop. In AZPR, each node uses a dynamic pseudonym as its node identifier rather than using its real MAC address, which can be used to trace nodes' existence in the network. To avoid pseudonym collision, we use a collision resistant hash function, such as SHA-1, to hash a node's MAC address and current time stamp. To prevent an attacker from recomputing the pseudonym, the time stamp should be precise enough (e.g., nanoseconds). Considering the network delay, the attacker needs to compute, e.g., 105, times for one packet per node. There may also be many nodes for an attacker to listen, so the computing overhead is not acceptable, and the success rate is low. A node's pseudonym expires after a specific time period in order to prevent adversaries from associating the pseudonyms with nodes. If pseudonyms are changed too frequently, the routing may get perturbed; and if pseudonyms are changed too infrequently, the adversaries may associate pseudonyms with nodes across pseudonym changes. Therefore, the pseudonym change frequently should be appropriately determined. Each node periodically piggybacks its updated position and pseudonym to "hello" messages, and sends the messages to its neighbors. Also, every node maintains a routing table that keeps its neighbors' pseudonyms associated with their locations.

As previous works [10] [7], assumption is made that the public key and location of the destination of a data transmission can be known by others, but its real identity requires protection. We can utilize a secure location service [9] to provide the information of each node's location and public key. Such a location service enables a source node, which is aware of the identity of the destination node, to securely obtain the location and public key of the destination node. The public key is used to enable two nodes to securely establish symmetric key Ks for secure communication. The destination location enables a node to determine the next hop in geographic routing. Specifically, trusted normal nodes or dedicated service provider nodes are used to provide location service. Each node has a location server. When a node A wants to know the location and public key of another node B, it will sign the request containing B's identity using its own identity. Then, the location server of A will return an encrypted position of B and its public key, which can be decrypted by A using the pre distributed shared key between A and its location server. When node A moves, it will also periodically update its position to its location server.

3. The AZPR Routing Algorithm

To be easy for illustration, we assume the entire network area is generally a rectangle in which nodes are randomly disseminated. The information of the bottom-right and upper left boundary of the network area is configured into each node when it joins in the system. This information enables a

node to locate the positions of nodes in the entire area for zone partitions in AZPR.

AZPR features a dynamic and unpredictable routing path, which consists of a number of dynamically determined intermediate relay nodes. As shown in the upper part of Fig. 1, given an area, we horizontally partition it into two zones A_1 and A_2. We then vertically partition zone A_1 to B_1 and B_2. After that, we horizontally partition zone B_2 into two zones. Such zone partitioning consecutively splits the smallest zone in an alternating horizontal and vertical manner. We call this partition process hierarchical zone partition. AZPR uses the hierarchical zone partition and randomly chooses a node in the partitioned zone in each step as an intermediate relay node (i.e., data forwarder), thus dynamically generating an unpredictable routing path for a message.

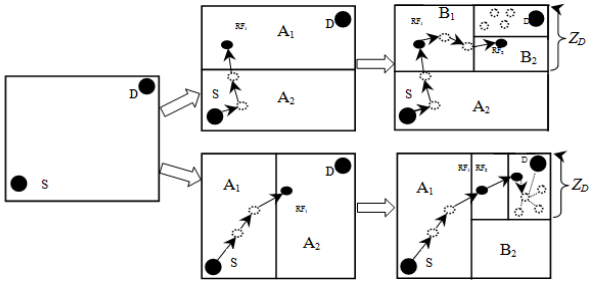

Figure 1: Examples of Horizontal and Vertical partitions

Example of routing in AZPR is as shows in Fig. 2. We call the zone having k nodes where D resides the destination zone, denoted as Z_D. k is used to control the degree of anonymity protection for the destination. The shaded zone in Fig. 2 is the destination zone. Specifically, in the AZPR routing, each data source or forwarder executes the hierarchical zone partition. It first checks whether itself and destination are in the same zone. If so, it divides the zone alternatively in the horizontal and vertical directions. The node repeats this process until it and Z_D are not in the same zone. It then randomly chooses a position in the other zone called temporary destination (*TD*), and uses the GPSR routing algorithm to send the data to the node closest to *TD*. This node is defined as a random forwarder (*RF*). Fig. 3 shows an example where node N_3 is the closest to TD, so it is selected as a *RF*. AZPR aims at achieving k-anonymity [25] for destination node D, where k is a predefined integer. Thus, in the last step, the data are broadcasted to k nodes in Z_D, providing k-anonymity to the destination. Fig. 1 shows two possible routing paths for a packet *pkt* issued by sender S targeting destination D in AZPR. There are also many other possible paths. In the upper routing flow, data source S first horizontally divides the area into two equal-size zones, A_1 and A_2, in order to separate S and Z_D. S then randomly selects the first temporary destination TD_1 in zone A_1 where Z_D resides. Then, S relies on GPSR to send *pkt* to TD_1. The *pkt* is forwarded by several relays until reaching a node that cannot find a neighbor closer to TD_1. This node is considered to be the first random-forwarder RF_1. After RF_1 receives *pkt*, it vertically divides the region A_1 into regions B_1 and B_2 so that Z_D and it are separated in two different zones. Then, RF_1 randomly selects the next temporary destination TD_2 and uses GPSR to send *pkt* to TD_2. This process is repeated until a packet receiver finds itself residing in Z_D, i.e., a partitioned zone is Z_D having k nodes. Then, the node broadcasts the *pkt* to the k nodes. The lower part of Fig. 1 shows another routing path based on a different partition pattern. After S vertically partitions the whole area to separate itself from Z_D, it randomly chooses TD_1 and sends *pkt* to RF_1. RF_1 partitions zone A_2 into B_1 and B_2 horizontally and then partitions B_1 into C_1 and C_2 vertically, so that itself and Z_D are separated.

Therefore, AZHPR sets the partition in the alternative horizontal and vertical manner in order to ensure that a *pkt* approaches D in each step. As GPSR, we assume that the destination node will not move far away from its position during the data transmission, so it can successfully receive the data.

3.1 Destination Zone Partition

The reason we use ZD rather than D is to avoid exposure of D. Zone position refers to the upper left and bottom-right coordinates of a zone. One problem is how to find the position of Z_D, which is needed by each packet forwarder to

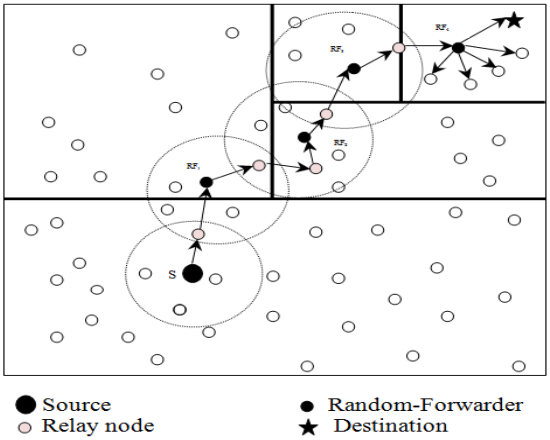

Figure 2: Routing among partitions in AZPR

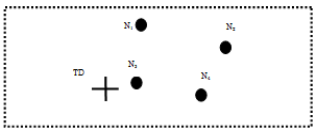

Figure 3: Selecting a RF according to TD

Check whether it is separated from the destination after a partition and whether it resides in Z_D. Let H denote the total number of partitions in order to produce Z_D. Using the number of nodes in Z_D (i.e., k), and node density ρ, H is calculated by

$$H = \log_2\left(\frac{\rho \cdot G}{k}\right)$$

where G is the size of the entire network area. Using the calculated H, the size G, the positions (0, 0) and (x_G, y_G) of the entire network area, and the position of D, the source S can calculate the zone position of ZD. Assume AZPR partitions zone vertically first. After the first vertical partition, the positions of the two generated zones are (0, 0), $(0.5x_G, y_1G)$ and $(0.5x_G, 0), (x_1(G,) y_1G)$. S then finds the zone where Z_D is located and divides that zone horizontally. This recursive process continues until H

partitions are completed. The final generated zone is the desired destination zone, and its position can be retrieved accordingly. Therefore, the size of the destination zone is $\frac{G}{2^H}$ 2H. For example, for a network with size G=8 and position represented by (0,0) and (4,2), if H= 3 and the destination position is (0.5, 0.8), the resulting destination zone's position is (0,0) and (1,1) with size of $\frac{8}{2^3}$.=1.

3.2 Parameters

The following metrics are used to evaluation the routing Performance in terms of effectiveness on anonymity protection and efficiency:

[1] *The number of actual participating nodes.* These nodes include RFs and relay nodes that actually participate in routing. This metric demonstrates the ability of ALERT's randomized routing to avoid routing pattern detection.

[2] *The number of random forwarders.* This is the number of actual RFs in a S-D routing path. It shows routing anonymity and efficiency.

[3] *The number of remaining nodes in a destination zone.* This is the number of original nodes remaining in a destination zone after a time period. A larger number provides higher anonymity protection to a destination and to counter the intersection attack.

[4] *The number of hops per packet.* This is measured as the accumulated routing hop counts divided by the number of packets sent, which shows the efficiency of routing algorithms.

[5] *Latency per packet.* This is the average time elapsed after a packet is sent and before it is received. It includes the time cost for routing and cryptography. This metric reflects the latency and efficiency of routing algorithms.

[6] *Delivery rate.* This is measured by the fraction of packets that are successfully delivered to a destination. It shows the robustness of routing algorithms to adapt to mobile network environment.

3.3 Packet Format of AZPR

For successful communication between S and D, S and each packet forwarder embeds the following information into the transmitted packet.

[1] The destination zone position Z_D, i.e., the H^{th} partitioned zone.

[2] The encrypted zone position of the H^{th} partitioned zone of S using D's public key.

[3] The current randomly selected temporary destination (TD) for routing.

[4] A bit (i.e., 0/1), which is flipped by each RF, indicating the partition direction (horizontal or vertical) of the next RF.

In order to save computing resources, we let the source node calculate the information of (1) and (2) and forward it along the route rather than letting each packet forwarder calculates the values. In order to hide the packet content from adversaries, AZPR employs cryptography. The work in [26] experimentally proved that generally symmetric key cryptography costs hundreds of times less overhead than public key cryptography while achieving the same degree of security protection. Thus, instead of using public key cryptography, AZPR uses symmetric key encryption for transmitted data. Recall that S can get D's public key from the secure location service. In an S-D communication, S first embeds a symmetric key $K^S{}_s$, encrypted using D's public key, into a packet. Later, D sends S its requested contents, encrypted with $K^S{}_s$, decrypted by its own public key. Therefore, the packets communicated between S and D can be efficiently and securely protected using $K^S{}_s$.

<table>
<tr><td colspan="2">RREQ/RREP/ NAK</td><td>P_S</td><td>P_D</td><td>L_{Zs}</td><td>L_{Z_D}</td><td>L_{RF}</td></tr>
<tr><td>h</td><td>H</td><td>$K^S{}_{pub}$</td><td>(TTL) $K^{RN}{}_{pub}$</td><td colspan="2">(Bitmap) $K^D{}_{pub}$</td><td>Data (null in NAK)</td></tr>
</table>

Figure 4: Packet format of AZPR

Fig. 4 shows the packet format of AZPR, which omits the MAC header. Because of the randomized routing nature in AZPR, we have a universal format for RREQ/RREP/NAK. Anode use NAK to acknowledge the loss of packets. The data field of RREQ/RREP is left blank in NAK packets. Flooding based anonymity routing usually uses ACKs, while NAKs are often adopted in geographic routing-based approaches [7] to reduce traffic cost. For the same purpose, we choose to use NAKs. In the packet, P_S is the pseudonym of a source; P_D is the pseudonym of the destination; L_{Z_S} and L_{Z_D} are the positions of the H^{th} partitioned source zone and destination zone, respectively; L_{TD} is the currently selected TD's coordinate; h is the number of divisions so far, H is the maximum allowed number of divisions; and $K^S{}_s$ denotes the symmetric key of a source. Particularly,(TTL)K^{RN}_{pub} pub is used for the protection of source anonymity and ,(Bitmap)K^{RN}_{pub} pub is used for solving intersection attack. When node A wants to know the location and public key of another node B, it will contact its location server, thus there is no need to exchange shared keys between nodes.

3.4 Source Anonymity

AZPR contributes to the achievement of anonymity by restricting a node's view only to its neighbors and constructing the same initial and forwarded messages. This makes it difficult for an intruder to tell if a node is a source or a forwarding node. To strengthen the anonymity protection of the source nodes, we further propose a lightweight mechanism called "notify and go." Its basic idea is to let a number of nodes send out packets at the same time as S in order to hide the source packet among many other packets.

"Notify and go" has two phases: "notify" and "go." In the first "notify" phase, S piggybacks its data transmission notification with periodical update packets to notify its neighbors that it will send out a packet. The packet includes two random back-off time periods, t and t_0. In the "go" phase, S and its neighbors wait for a certain period of randomly chosen time ϵ $[t, t+t_0]$ before sending out

messages. *S's* neighbors generate only several bytes of random data just in order to cover the traffic of the source. *T* should be a small value that does not affect the transmission latency. A long t_0 may lead to a long transmission delay while a short t_0 may result in interference due to many packets being sent out simultaneously. Thus, t_0 should be long enough to minimize interference and balance out the delay between *S* and *S's* farthest neighbor in order to prevent any intruder from discriminating *S*. Notify and go provides difficult for an attacker to analyze traffic to discover *S* even if it receives the first notification.

4. Anonymity Protection and Strategies against Attack

This section discusses the performance of AZPR in providing anonymity protection and its performance and strategies to deal with some attacks.

4.1 Anonymity Protection

AZPR offers identity and location anonymity of the source and destination, as well as route anonymity. AZPR makes the route between *S-D* pair difficult to discover by randomly and dynamically selecting the relay nodes. The resultant different routes for transmissions between a given *S-D* pair make it difficult for an intruder to observe a statistical pattern of transmission. This is because the *RF* set changes due to the random selection of *RFs* during the transmission of each packet. Even if an adversary detects all the nodes along a route once, this detection does not help it in finding the routes for subsequent transmissions between the same *S-D* pair. Also, the anonymous path between *S* and *D* ensures that nodes on the path do not know where the endpoints are. AZPR incorporates the "notify and go" mechanism to prevent an intruder from identifying which node within the source neighborhood has initiated packets. AZPR also provides *k*-anonymity to destinations by hiding *D* among *k* receivers in Z_D. Thus, an eavesdropper can only obtain information on Z_D, rather than the destination position, from the packets and nodes en route. The route anonymity due to random relay node selection in AZPR prevents an intruder from intercepting packets or compromising vulnerable nodes en route to issue DoS attacks. In AZPR, the routes between two communicating nodes are constantly changing, so it is difficult for adversaries to predict the route of the next packet for packet interception.

4.2 Resilience to Timing Attacks

In timing attacks [16], through packet departure and arrival times, an intruder can identify the packets transmitted between *S* and *D*, from which it can finally detect *S* and *D*. For example, two nodes *A* and *B* communicate with each other at an interval of 5 seconds. After a long observation time, the intruder finds that *A's* packet sending time and *B's* packet receiving time have a fixed five second difference such as (19:00:55, 19:01:00) and (20:01:33, 20:01:38). Then, the intruder would suspect that *A* and *B* are communicating with each other.

Avoiding the exhibition of interaction between communication nodes is a way to counter timing attacks. In AZPR, the "notify and go" mechanism and the broadcasting in Z_D both put the interaction between *S-D* into two sets of nodes to obfuscate intruders. More importantly, the routing path between a given S-D and the communication delay changes constantly, which again keeps an intruder from identifying the *S* and *D*.

4.3. Strategy to Counter Intersection Attacks

In an intersection attack, an attacker with information about active users at a given time can determine the sources and destinations that communicate with each other through repeated observations. Intersection attacks are a well known problem and have not been well resolved [16]. Though AZPR offers *k*-anonymity to *D*, an intersection attacker can still identify *D* from repeated observations of node movement and communication if *D* always stays in Z_D during a transmission session. This is because as long as *D* is conducting communication, the attacker can monitor the change of the members in the destination zone containing *D*. As time elapses and nodes move, all other members may move out of the destination zone except *D*. As a result, *D* is identified as the destination because it always appears in the destination zone. Fig. 5a is the status of a Z_D

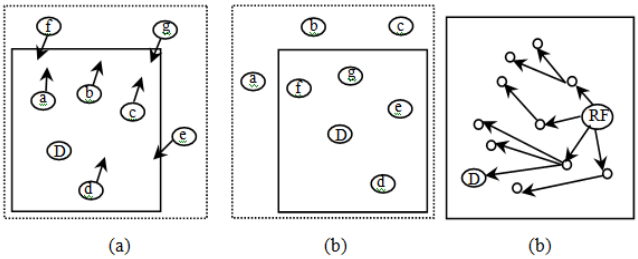

Figure 5: Intersection Attack and Solutions

After a packet is broadcasted to the zone. The arrows show the moving directions of nodes. We can see that nodes a, b, c, d, and *D* are in Z_D. Fig. 5b is the subsequent status of the zone the next time a packet is transmitted between the same *S-D* pair. This time, nodes d, e, f, g, and *D* are in Z_D. Since the intersection of the in-zone nodes in both figures includes d and *D*, *D* could be identified by the attacker. Therefore, the longer an attacker watches the process, the easier it is to identify the destination node. We propose another strategy to resolve this problem. Note that the attacker can be puzzled and lose the cumulated observation by making it occasionally fail to observe *D's* reception of packets. Since packets are delivered to Z_D constantly in long-duration sessions rather than using direct local broadcasting in the zone, the last *RF* multicasts packet *pkt1* to a partial set of nodes, say m nodes out of the total *k* nodes in the zone. The *m* nodes hold the packets until the arrival of the next packet *pkt2*. Upon the arrival of the next packet, the *m* nodes conduct one-hop broadcasting to enable other nodes in the zone to also receive the packet in order to hide *D*. Fig. 5c shows the two-step process with the first step in solid arrows and the second step in dashed arrows. We can see that the first step reaches a number of nodes in the destination zone, but the destination is reached in the second step. Because the deliveries of *pkt1* and *pkt2* are mixed, an attacker observes that *D* is not in the recipient set of *pkt1* though *D* receives

pkt1 in the delivery time of *pkt2*. Therefore, the attacker would think that *D* is not the recipient of every packet in Z_D in the transmission session, thus foiling the intersection attack.

5. Simulation Result

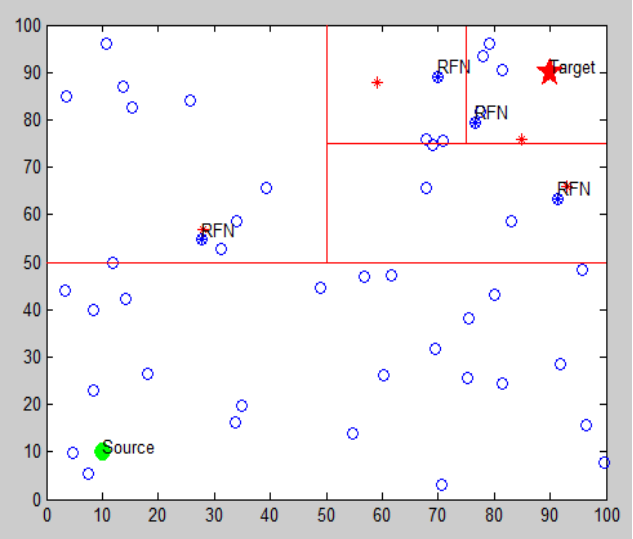

Figure 6: Result demonstrating a network partitions along with Random Forwarders

The tests were carried out on MATLAB simulator using MAC protocol with a standard wireless transmission range of 250 m and UDP/CBR traffic with a packet size of 512 bytes. The test field in our experiment was set to a 100m * 100m area with 50 nodes moving at a speed of 2 m/s, unless otherwise specified. The density was set to 5 to 10 nodes per square meters. The duration of each simulation was set to 100 s unless otherwise indicated. The number of pairs of S-D communication nodes was set to 10 and S-D pairs are randomly generated. S sends a packet to D at an interval of 2m/s. The final results are the average of results of 30 runs.

6. Conclusion

The anonymous routing protocols in MANET relying on either hop-by-hop encryption or redundant traffic generate high cost and cannot provide complete anonymity. To provide anonymity protection for source, destination and route at low cost, an algorithm AZPR is introduced in this paper. This algorithm partition the network into zones and selection of random forwarder provides route anonymity and consumes less resource. Although the randomized routing provides anonymity for the source and destination, the Notify and Go mechanism that we have introduced strengthen the source anonymity. The two tier broadcasting mechanism used to forward the packets to all the nodes in the destination zone (D_Z), strengthen the destination anonymity. These two mechanisms provide resiliency to intersection and timing attacks. Like other anonymity routing algorithms, AZPR is not completely resistant to all attacks. Future work lies in reinforcing AZPR in an attempt to thwart stronger, active attackers and demonstrating comprehensive theoretical and simulation results.

References

[1] K.E. Defrawy and G. Tsudik, "ALARM: Anonymous Location-Aided Routing in Suspicious MANETs," *Proc. IEEE Int'l Conf. Network Protocols (ICNP),* 2007.
[2] K.E. Defrawy and G. Tsudik, "PRISM: Privacy-Friendly Routing in Suspicious MANETs (and VANETs)," *Proc. IEEE Int'l Conf. etwork Protocols (ICNP)*, 2008.
[3] Y.-C. Hu, A. Perrig, and D.B. Johnson, "Ariadne: A Secure On- Demand Routing Protocol for Ad Hoc Networks," *Wireless Networks,* vol. 11, pp. 21-38, 2005.
[4] Aad, C. Castelluccia, and J. Hubaux, "Packet Coding for Strong Anonymity in Ad Hoc Networks," *Proc. Securecomm and Workshops*, 2006.
[5] C.-C. Chou, D.S.L. Wei, C.-C. Jay Kuo, and K. Naik, "An Efficient Anonymous Communication Protocol for Peer-to-Peer Applications over Mobile Ad-Hoc Networks," *IEEE J. Selected Areas in Comm.,* vol. 25, no. 1, pp. 192-203, Jan. 2007.
[6] B. Zhu, Z. Wan, M.S. Kankanhalli, F. Bao, and R.H. Deng, "Anonymous Secure Routing in Mobile Ad-Hoc Networks," *Proc. IEEE 29th Ann. Int'l Conf. Local Computer Networks (LCN),* 2004.
[7] X. Wu, J. Liu, X. Hong, and E. Bertino, "Anonymous Geo-Forwarding in MANETs through Location Cloaking," *IEEE Trans. Parallel and Distributed Systems*, vol. 19, no. 10, pp. 1297-1309, Oct. 2008.
[8] K. El-Khatib, L. Korba, R. Song, and G. Yee, "Anonymous Secure Routing in Mobile Ad-Hoc Networks," *Proc. Int'l Conf. Parallel Processing Workshops (ICPPW)*, 2003.
[9] J. Li, J. Jannotti, D.S.J. De, C. David, R. Karger, and R. Morris, "A Scalable Location Service for Geographic Ad Hoc Routing," *Proc. ACM MobiCom,* 2000.
[10] Y. Zhang, W. Liu, and W. Luo, "Anonymous Communications in Mobile Ad Hoc Networks," *Proc. IEEE INFOCOM,* 2005.
[11] J. Kong, X. Hong, and M. Gerla, "ANODR: Anonymous on Demand Routing Protocol with Untraceable Routes for Mobile Ad-Hoc Networks," *Proc. ACM MobiHoc,* pp. 291-302, 2003.
[12] Y.-C. Hu, D.B. Johnson, and A. Perrig, "SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks," *Proc. IEEE Workshop Mobile Computing Systems and Applications (WMCSA),* 2002.
[13] L. Zhao and H. Shen, "ALERT: An Anonymous Location-Based Efficient Routing Protocol in MANETs," *Proc. Int'l Conf. Parallel Processing (ICPP),* 2011.

Author Profile

Ayeesha Siddiqha received the B.E degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum in 2011 and M.Tech degree Computer Science and Engineering from Visvesvaraya Technological University, Belgaum in 2013. At Present she is working as an Assistant Professor in Computer Science Department in Malnad College of Engineering, Hassan

Arshad Khan received the B.E degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum in 2011 and persuing M.Tech degree Computer Science and Engineering from Visvesvaraya Technological University, Belgaum.