Chapter II

RELATED WORKS

2.1 Key Management Approaches for MANET
2.2 Routing Protocols for MANET
2.3 Mobility models for MANET
2.4 Chapter Summary
II. RELATED WORKS

Group confidentiality[20] is the most important service for several MANET applications. In group communication, the group members share a common secret key, which need to be distributed to the members of the group in a secure and reliable manner. Group confidentiality function authenticates the valid intended users to decrypt the data.

Key management becomes a central part of any secure communication and is the weakest point of system security and the protocol design. Most of these security services rely generally on encryption using Traffic Encryption Keys (TEKs) and re-encryption using Key Encryption Keys (KEKs) [43]. The Key management includes creating, distributing and updating the keys which constitutes the basic block for secure multicast communication applications[40].

In a secure multicast communication, each member holds a key to encrypt and decrypt the multicast data. When a member joins and leaves a group, the key has to be updated and distributed to all group members in order to meet the multicast key management requirements.

A key is a piece of input information for cryptographic algorithms. If the key is released, the encrypted information will be disclosed. To break the cycle, some non-cryptographic approaches need to be used, eg. smart card, or biometric identity, such as fingerprint, etc.

2.1 Key Management Approaches for MANET

The key management approaches can be classified into three classes, namely centralized, distributed and decentralized. Figure 2.1. illustrates this classification.
2.1.1 Centralized Approaches

In centralized approaches, a designated node (e.g., the group leader or a key server) is responsible for calculation and distribution of the group key to all the participants. The architecture uses only one server. The server is responsible for the generation, distribution and renewal of the group key.

![Key Management Approaches Diagram]

**Figure 2.1. Classification of key management approaches**

This approach is clearly not scalable since it suffers from the “1 affects n” phenomenon. Moreover, the unique key server forms a bottleneck in terms of security and resources. Centralized protocols are classified into three sub-categories namely pair wise key approach, secure broadcast and hierarchy of keys approach.

(a) **Pair wise key approach**: In this approach, the key server shares secret pair wise keys with each participant. These pair wise keys are generally called Key Encryption Keys (KEK). These keys are used to establish secure channels between
the key server and each group member in order to redistribute the current TEK securely when required. In dynamic groups, a new TEK is sent to all the valid group members encrypted with their respective KEKs. This valid group may contain a new member which joins the group to ensure backward secrecy and exclude the existing member which leaves the group to ensure forward secrecy.

For example, in GKMP[33], apart from pairwise keys and the group key, all current group participants know a group Key Encryption Key (gKEK). If a new participant joins the group, the server generates a new group key and a new gKEK. These keys are sent to the new member using the key it shares with key server, and to the old group member using the old gKEK.

Lazos.et.al.[45] proposed a similar scheme to GKMP. In this approach, authentication and authorization functions are delegated to other group members rather than centralized at the same group controller entity. It optimizes energy consumption by using the geographical localization of the group members.

In Chu.et.al. [24], a group leader shares a separate secret KEK with each group member. To send a secret multicast message ‘m’, the sender encrypts ‘m’ with a random key ‘k’. Then, the sender encrypts ‘k’ with the secret KEK which is shared with the group leader, and sends it to the group along with the encrypted message. On receiving the message, receivers cannot decrypt it since they do not know the random key ‘k’. When the leader receives the message, it decrypts ‘k’ using the key that it shares with the source and constructs a validation message which contains k encrypted with each KEK and then shares it with a valid group member (excluding the departing members). On receiving the validation message, each receiver decrypts ‘k’ using its KEK and decrypts ‘m’ which is encrypted using ‘k’.

(b) **Secure Broadcast**: In this approach, the rekeying of the group is based on broadcast messages instead of peer to peer secret transmission. Chiou and Chen
[22] proposed Secure Lock, a key management protocol where the key server requires only a single broadcast to establish the group key or to re-key the entire group in case of a leave. This protocol minimizes the number of re-key messages. However, it increases the computation at the server due to the Chinese Remainder calculations before sending each message to the group.

In Zheng et al.[82], the authors introduce two centralized group key management protocols based on the Chinese Remainder Theorem[83] (Chinese Remaindering Group Key (CRGK) and Fast Chinese Remaindering Group Key (FCRGK)). By shifting more computing load onto the key server, they optimize the number of broadcast messages to distribute the group key, user-side key computation and number of key storages.

The protocol requires the key servers to compute the solution of the Chinese Remainder Theorem. They broadcast one message for distributing the group key and each group member to compute only one modulo arithmetic and one XOR operation for each key update and store only two keys all the time. With the necessity for key updating broadcasting message number, user key computation, and user key storage, the protocols require more computation power from the key server.

(c) **Hierarchy of Keys Approach**: The most efficient approach to rekeying in the centralized case is the hierarchy of keys approach. Here, the key server shares keys with subgroups of the participants, in addition to the pairwise keys. Thus, the hierarchical approach trades off storage for number of transmitted messages. The aim of this approach is to reduce the required number of TEK update messages induced by re-keying after membership changes.

The key server shares secret keys with sub-groups of the entire secure group in addition to the individual channels. When a member leaves the secure
session, the key server uses the secret sub-group keys, that are unknown to the departing member, to distribute the new TEK.

Logical key hierarchy proposed in [25], is a typical solution to this category. The intermediate keys shared with different combinations of sub-groups form a hierarchy (generally a binary tree) of keys. The number of update messages induced by this protocol is in the order of log (n) with ‘n’ being the number of valid group members. The key server maintains a tree with subgroup keys in the intermediate nodes and the individual keys in the leaves. Apart from the individual keys shared with the key server, each node knows all keys on the path to the root. The group key is stored in the root. For a balanced binary tree, each member stores a logarithmical number of keys, and the number of rekey messages is also logarithmic.

One-way function trees (OFT) [1] enable the group members to calculate the new keys based on the previous keys using a one-way function, which further reduces the number of rekey messages. Efficient Large group Key distribution (ELK) [64] protocol proposed by Perrig et al. are variants of LKH protocol that allow to save some update message transmissions of intermediate keys of the hierarchy by replacing them with one-way function computations. It uses pseudo random functions to generate the new KEKs.

Centralized Flat table Key Management (CFKM) [78] protocol proposed by Waldvogel et al. replaces the key hierarchy with a flat table in order to reduce the number of keys maintained by the Key Server. The table contains a single entry for the TEK and ‘2w’ entries for the KEKs, where ‘w’ is the number of bits in a member identifier. Two keys are associated to the two possible values of each bit in a member id. When a member leaves the group, all the keys held by this departing member should be modified to assure forward secrecy.
Therefore the key server sends a re-key message containing two parts: The first part contains the TEK encrypted with uncompromised KEK from the flat table, and hence all the remaining members would be able to decrypt the new TEK. The second part contains the new KEKs encrypted with both the old KEK and the new TEK. By this way, the leaving member cannot recover the new TEK and the remaining members can update their old KEKs without having access to the KEKs of other members. Table 2.1 shows the comparison results of some of the Centralized key management protocols.

### TABLE 2.1. **COMPARISON OF CENTRALIZED APPROACHES**

<table>
<thead>
<tr>
<th>Category</th>
<th>Protocol</th>
<th>1 affects n</th>
<th>Rekey Overhead</th>
<th>Storage Overheads</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Key Server</td>
<td>Member</td>
</tr>
<tr>
<td>Pair wise Keys</td>
<td>GKMP</td>
<td>Yes</td>
<td>2</td>
<td>n + 2</td>
<td>3</td>
</tr>
<tr>
<td>Secure Broadcast</td>
<td>Secure Lock</td>
<td>No</td>
<td>2</td>
<td>2n</td>
<td>2</td>
</tr>
<tr>
<td>Hierarchy of Keys</td>
<td>LKH</td>
<td>Yes</td>
<td>Log₂(n) - 1</td>
<td>2n - 1</td>
<td>Log₂(n) + 1</td>
</tr>
<tr>
<td></td>
<td>OFT</td>
<td>Yes</td>
<td>Log₂(n) + 1</td>
<td>2n - 1</td>
<td>Log₂(n) + 1</td>
</tr>
</tbody>
</table>

The Pair wise key approach protocols achieve an excellent result for storage of the members. However, this result is achieved by providing no method for re-keying the group after a member has left, except re-creating the entire group which induces a O(n) re-key messages overhead, with ‘n’ being the number of the remaining group members.

Secure Broadcast achieves excellent results for storage and communication overheads on both members and the key server. However, these results are achieved by increasing the computation overhead at the key server due to the
Chinese Remainder calculations. So far, the best solution for centralized group key management appear to be those using a hierarchical tree of KEKs. They achieve overall good results without compromising any aspects of security.

2.1.2 Distributed Key Agreement Approaches

With distributed or contributory key-agreement protocols, the group members cooperate to generate the traffic encryption key, to establish secure communications between them. This improves the reliability of the overall system and reduces the bottlenecks in the network in comparison to the centralized approach, but is less scalable because the traffic encryption key is composed of the contributions of all group members, and needs more processing. The protocols of this category are classified into three sub-categories namely Ring based cooperation, Hierarchical based cooperation and Broadcast based cooperation depending on the virtual topology created by the members for cooperation.

(a) **Ring based Cooperation**: In some protocols, members are organized in a ring. The CLIQUES [28] protocol suite is an example of ring-based cooperation. This protocol arranges group members as \((M_1, M_n)\) and \(M_n\) as controller. It specifies a role of the controller that collects contributions of other group members, adds own contribution, and broadcasts information that allows all members to compute the group key.

The choice of the controller depends on the dynamic event and the current structure. During additions, new members are appended to the end of the list CLIQUES and they do not provide verifiable trust relationship, because no other member can check whether values forwarded by \(M_i\), or the set, broadcasted by the controller are correctly built.

(b) **Hierarchical based Cooperation**: In the hierarchical GKA protocols, the members are organized according to some structure. STR protocol [72] uses
the linear binary tree for cooperation and provides communication efficient protocols with especially efficient join and merges operations. STR defines the role of the sponsor temporarily and it can be assigned to different members on dynamic events depending on the current tree structure. The sponsor reduces the communication overhead as it performs some operations on behalf of the group. The sponsor is not a central authority. STR provides verifiable trust relationship because every broadcasted public key can be verified by at least one other participant.

(c) **Broadcast based Cooperation:** Broadcast based protocols have constant number of rounds. For example, in three-round Burmester-Desmedt (BD) protocol [54], each participant broadcasts intermediate values to all other participants in each round. The communication and computational load is shared equally between all parties. This protocol does not provide verifiable trust relationship, since no other group member can verify the correctness of the broadcasted values. Table 2.2 shows the comparison results of Distributed Key Agreement Approaches.

**TABLE 2.2.**

**COMPARISON OF DISTRIBUTED KEY AGREEMENT APPROACHES**

<table>
<thead>
<tr>
<th>Category</th>
<th>Protocol</th>
<th>Verifiable Trust</th>
<th>Leader Required</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ring based</td>
<td>Cliques</td>
<td>No</td>
<td>No</td>
<td>Suitable to low level security</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>STR</td>
<td>Yes</td>
<td>No</td>
<td>Suitable to high level security</td>
</tr>
<tr>
<td>based based</td>
<td>BD</td>
<td>No</td>
<td>No</td>
<td>Suitable to low level security</td>
</tr>
</tbody>
</table>

40
2.1.3 Decentralized Approaches

The decentralized approach divides the multicast group into a fixed number of sub-groups. Each of them shares a local session key managed by a local controller, thus attenuating the “1 affects n” phenomenon. When a member joins or leaves the group, only the concerned sub-group will renew its local key. The decentralized protocols are subdivided into two categories as static clustering and dynamic clustering.

In static clustering approach, the multicast group is initially divided into several subgroups. Each cluster or sub-group of the multicast group shares and manages a local TEK, requiring several decryption and re-encryption operations of the multicast flow, when it passes from a sub-group to another. This family of protocols is not flexible and not adapted to the dynamics of the multicast group. IOLUS [55] and DEP [28] belong to the categories, which are more scalable than centralized protocol.

Dynamic clustering approach aims to solve the “1 affects n” phenomenon. It uses only one TEK for all clusters in the multicast group. The multicast group divides into sub-groups hierarchically and each of them being managed by a sub-group manager. The managers, which are not members of the multicast group, do not need to decrypt the multicast flow sent by the source. These protocols use a double encryption of the traffic encryption key that requires more KEKs.

This approach starts a multicast session with centralized key management and divides the group dynamically. AKMP [8], SAKM [19] belong to this approach and are dedicated to wired networks. Enhanced BAAL [11] and OMCT [12] proposes dynamic clustering scheme for multicast key distribution in ad hoc networks.
OMCT (Optimized Multicast Cluster Tree) [12,13,14] is a dynamic clustering scheme for multicast key distribution dedicated to Mobile Ad hoc Networks. This scheme optimizes energy consumption and latency for key delivery. The main idea is to elect the local controller of the created clusters. OMCT needs the geographical location information of all group members in the construction of the key distribution tree.

Once the clusters are created within the multicast group, the new LC becomes responsible for the local key management and distribution to their local members. The election of local controllers is done according to the localization and GPS (Global Positioning System) information of the group members, which does not reflect the true connectivity between nodes. Table 2.3 shows the comparison of decentralized approaches.

**TABLE 2.3. COMPARISON OF DECENTRALIZED APPROACHES**

<table>
<thead>
<tr>
<th>Category</th>
<th>Protocol</th>
<th>1 affects n</th>
<th>Local Re-key</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Clustering</strong></td>
<td>IOLUS</td>
<td>Yes</td>
<td>Yes</td>
<td>More Scalable</td>
</tr>
<tr>
<td></td>
<td>DEP</td>
<td>Yes</td>
<td>No</td>
<td>More Scalable</td>
</tr>
<tr>
<td><strong>Dynamic Clustering</strong></td>
<td>AKMP</td>
<td>No</td>
<td>Yes</td>
<td>Suitable to wired networks</td>
</tr>
<tr>
<td></td>
<td>SAKM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhanced BAAL</td>
<td>No</td>
<td>Yes</td>
<td>Suitable to wired networks</td>
</tr>
<tr>
<td></td>
<td>OMCT</td>
<td>No</td>
<td>Yes</td>
<td>Suitable to wireless networks</td>
</tr>
</tbody>
</table>

Based on the literature review, it is obvious that OMCT is the efficient dynamic clustering approach for secure multicast distribution in mobile ad hoc networks. To enhance its efficiency, it is necessary to overcome the criteria, as
OMCT needs geographical location information in the construction of key
distribution tree by reflecting true connectivity between nodes.

To overcome the above limitations another method called Optimized
Multicast Cluster Tree with Multipoint Relays (OMCT with MPR)[13] is
introduced, which uses the information of Optimized Link State Routing Protocol
(OLSR) to elect the LC of the created clusters. OMCT with MPRs assumes that
routing control messages have been exchanged before the key distribution. It does
not acknowledge the transmission and results in retransmission which consumes
more energy and unreliable key distribution due to high packet drop ratio for
mobile ad hoc networks.

2.1.4 Discussions

In centralized protocols, GKMP achieves an excellent result for storage of
the members. However this result is achieved by no method for rekeying the group
after a member has left, except re-creating the entire group which induces O(n)
rekey message overhead where ‘n’ is the number of the remaining group members.
Secure Lock achieves excellent results for storage and communication overheads
on both, the members and the key server. However, these results are achieved by
increasing the computation overhead at the key server due to the Chinese
Remainder calculations.

Distributed key agreement protocols do not rely on a group leader. Without
a leader, all members are treated equally and if one or more members fail to
complete the protocol, it will not affect the whole group. In the protocols with a
group leader, the leader failure is fatal for creating the group key and the operation
has to be restarted from scratch.

The “1 affects n” phenomenon is not considered because in distributed
protocols all the members are contributors in the creation of the group key and
hence all of them should commit to the new key whenever a membership change occurs in the group.

In Decentralized protocols, the static clustering approaches are more scalable than centralized protocol. These protocols are dedicated to operate within wired networks. Dynamic clustering approach aims to solve the “1 affect n” phenomenon. Dynamic clustering scheme is well suited for multicast key distribution in Mobile Ad hoc Networks. OMCT (Optimized Multicast Cluster Tree) is a dynamic clustering scheme for multicast key distribution dedicated to operate in Mobile Ad hoc Networks. This scheme optimizes energy consumption and latency for key delivery.

As the nodes are dynamic in nature, the resulting topology needs to be communicated to all other nodes to maintain the connectivity information. In accommodating the communication needs of the user applications, the limited bandwidth of wireless channels and their generally hostile transmission characteristics, impose additional constraints on how much and how often administrative and control information may be exchanged. Ensuring effective routing is one of the major challenges for mobile ad hoc networking.

2.2 Routing in Mobile Ad hoc Networks

Routing in mobile ad hoc network faces additional problems and challenges when compared to routing in traditional wired networks with fixed infrastructure. There are several well known Protocols [2,70] specifically developed for ad hoc networking environments. In order to facilitate communication within the network, a routing protocol is used to discover routes between nodes. The primary goal of such an ad hoc network routing protocol is to establish efficient route between a pair of nodes, so that messages may be delivered in a timely manner.
Route construction should be done with minimum overhead and bandwidth consumption. An ad hoc routing protocol is a convention or standard that controls the nodes and communication between computing devices in a MANET. In ad hoc networks, nodes do not have a priori knowledge of topology of network around them and they have to discover it. To provide routes in such dynamic environments, many routing protocols have been proposed over the last few years. Routing in ad hoc networks has been an active area of research for some time now.

2.2.1 Broad classification of routing algorithms

The existing routing protocols[16,47 and 80] for Mobile ad hoc networks emphasize different implementation scenarios. However, the basic goals are to devise a routing protocol that minimizes control overhead, packet drop ratio, and energy usage while maximizing the throughput. The routing protocols in mobile ad hoc networks can hence be divided into categories based on their underlying architectural framework as shown in Figure 2.2.

Figure 2.2. Classification of Routing Protocols in Mobile Ad hoc Networks
**Proactive Routing:** In proactive routing algorithms, each node maintains a routing table. It contains the next hop information for every other node, and one route between the source node and the destination node in the network makes the approach proactive.

This type of protocols maintain fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. Example of proactive protocol is Destination Sequenced Distance Vector (DSDV) routing protocol.

**Reactive Routing:** In reactive routing algorithms, a path discovery process determines the path to the destination only. When the node has a packet to forward, it reacts to a request to send data to a host. These types of routing algorithms are also referred to as on-demand routing protocols. This type of protocols find a route on demand by flooding the network with Route Request packets. Two prominent examples are Dynamic Source Routing (DSR) and Ad hoc On demand Distance Vector (AODV) routing algorithm.

**Hybrid Routing:** This type of protocols combine the advantages of proactive and of reactive routing. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding. The choice for one or the other method requires predetermination for typical cases. Example of hybrid routing protocol is Zone Routing Protocol (ZRP).

The advantages and disadvantages of the main routing protocol approaches are cited in Table 2.4.
TABLE 2.4 COMPARISON OF PROACTIVE, REACTIVE AND HYBRID APPROACHES

<table>
<thead>
<tr>
<th></th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Proactive</strong></td>
<td>• Up-to-date routing information</td>
<td>• Slow convergence</td>
</tr>
<tr>
<td></td>
<td>• Quick establishment of routes</td>
<td>• Tendency of creating loops</td>
</tr>
<tr>
<td></td>
<td>• Less delay</td>
<td>• Requires more resources</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Routing information not fully used</td>
</tr>
<tr>
<td><strong>Reactive</strong></td>
<td>• Saving resources</td>
<td>• Not always up-to-date routes</td>
</tr>
<tr>
<td></td>
<td>• Reduces routing load</td>
<td>• More delay</td>
</tr>
<tr>
<td></td>
<td>• Loop free</td>
<td>• Control traffic and Overhead cost</td>
</tr>
<tr>
<td><strong>Hybrid</strong></td>
<td>• Scalability</td>
<td>• Inter-zone delay</td>
</tr>
<tr>
<td></td>
<td>• Limited search cost</td>
<td>• Requires more resources for large size zones</td>
</tr>
<tr>
<td></td>
<td>• Up-to-date routing information within zones</td>
<td></td>
</tr>
</tbody>
</table>

2.2.2 Well known routing algorithms

Based on the desired routing properties in mobile ad hoc networks, some of the specific well known routing algorithms proposed are discussed in the literature.

**Destination Sequenced Distance Vector (DSDV)**

DSDV is a table-driven routing protocol based on Bellman–Ford routing algorithm[50]. It was developed by C.Perkins et al [60]. The main contribution of the algorithm was to solve the route loop problem. Every mobile node maintains a routing table. It contains all of the possible destinations in the network and each
individual hop counts to reach those destinations. Each entry also stores a sequence number that is assigned by the destination. Sequence numbers are used to identify stale entries and avoidance of loops. In order to maintain routing table consistency, routing updates are periodically sent throughout the network.

The two types of updates that can be employed are full dump and incremental. A full dump sends the entire routing table to the neighbors and requires multiple network protocol data units (NPDUs). Incremental updates are smaller updates that must fit in a packet and are used to transmit those entries from the routing table since the last full dump update.

When a network is stable, incremental updates are sent and full dump are usually infrequent. On the other hand, full dumps will be more frequent in a fast moving network. The mobile nodes maintain another routing table to have the information sent in the incremental routing packets. In addition to the routing table information, each route update packet contains a distinct sequence number that is assigned by the transmitter. The route labeled with the most recent (highest number) sequence number is used. If the two routes have the same sequence number then the shortest route is chosen.

DSDV is a hop-by-hop distance vector routing protocol. The key advantage of DSDV over traditional distance vector protocols is that it guarantees loop-freedom by using the concept of sequence numbers. Each DSDV node maintains a routing table listing the “next hop” for each reachable destination. Each node in the network advertises an increasing even sequence number for itself. Figure 2.3 shows the DSDV routing protocol workflow mechanism.

When a route from a node to a destination is broken, it advertises the route to destination with an infinite metric and increments the sequence number by one. This causes any node routing packets through source S to incorporate the
infinite-metric route into its routing table until the origin node hears a route to destination D with a higher sequence number.

![Figure 2.3 DSDV Routing Protocol](image)

**Figure 2.3 DSDV Routing Protocol**

DSDV uses triggered route updates when the topology changes. The transmission of updates is delayed to introduce a softening effect when the topology is changing quickly. This gives an additional adaptation of DSDV to ad hoc networks.

**Dynamic Source Routing (DSR)**

DSR[26] is a reactive protocol that uses source routing rather than hop-by-hop routing. Each routed packet carries the complete ordered list of nodes through which the packet must pass. The key advantage of source routing is that intermediate nodes do not need to maintain up-to-date routing information in order to route the packets they forward, since the packets themselves already contain all the routing decisions. For better understanding, Figure. 2.4 shows an example scenario of the DSR routing protocol workflow mechanism.
The DSR protocol consists of two mechanisms: Route Discovery and Route Maintenance. Route Discovery is the mechanism by which a node A needs to send a packet to a destination E and obtains a path. To perform a Route Discovery, the source node A broadcasts a Route Request packet flooded through the network in a controlled manner. It is answered by a Route Reply packet from either the destination node or another node that knows a route to the destination.

![DSR Routing Protocol](image)

**Figure 2.4 DSR Routing Protocol**

In order to reduce the cost of Route Discovery, each node maintains and actively uses a cache of source routes which it has learned or overheard. Thus the frequency and propagation of Route Requests are limited. Route Maintenance is the mechanism by which the sender A detects if the network topology has been changed then it can no longer use its route to the destination E because two nodes listed in the route have moved out of range of each other.

When Route Maintenance indicates a source route is broken, A is notified with a Route Error packet. The sender A can attempt to use any other route to destination node E already in its cache or can invoke Route Discovery again to
find a new path. The nodes 1, 2, 3...7 in the network is away from the task that is, not processing the packet after looking into the destination address and next possible neighbor node. A DSR node is able to learn routes by overhearing packets not addressed to it. However this feature requires an active receiver in the nodes, which may be rather power consuming and apparently does not improve performance.

**Ad Hoc On-Demand Distance Vector (AODV)**

The Ad hoc On-Demand Distance Vector routing protocol (AODV)[61,63] is an improvement of the Destination-Sequenced Distance Vector routing protocol (DSDV). It creates the routes in on-demand basis, as opposed to maintain a complete list of routes for each destination. Therefore, the literature on AODV, classifies it as a pure on-demand route acquisition system. The usage of the AODV protocol for mobile ad hoc networking applications provided consistent results for large scale scenarios. For better understanding Figure 2.5. shows the AODV routing protocol workflow mechanism.

![AODV Routing Protocol](image)

**Figure 2.5. AODV Routing Protocol**
AODV is essentially a combination of both DSR and DSDV. It borrows the basic on-demand mechanism of Route Discovery and Route Maintenance from DSR, plus the use of hop-by-hop routing, sequence numbers, and periodic beacons from DSDV. When a node S needs a route to some destination D, it broadcasts a Route Request message to its neighbors, including the last known sequence number for that destination. The route request is flooded until it reaches a node that knows a route to the destination D. Each node that forwards the Route Request creates a reverse route for itself back to node S.

When the Route Request reaches a node with a route to D, that node generates a Route Reply that contains the number of hops necessary to reach D and the sequence number for D most recently seen by the node generating the Route Reply. Every node that participates in forwarding this Route Reply towards the originator of the Route Request (node S), will create a route to D. The state created in each node along the path from S to D is hop-by-hop state; that is, each node remembers only the next hop and not the entire route, as would be done in source routing [29].

In order to maintain routes, AODV normally requires that each node periodically transmit a HELLO message, with a default rate of once per second. Failure to receive three consecutive HELLO messages from a neighbor is taken as an indication that the link to the neighbor is down.

Alternatively, the AODV specification briefly suggests that a node may use physical layer or link layer methods to detect link breakages to nodes that it considers neighbors. When a link goes down, any upstream node that has recently forwarded packets to a destination using that link is notified via an Unsolicited Route Reply containing an infinite metric for that destination [10]. Upon receipt of such a Route Reply, a node must acquire a new route to the destination using Route Discovery.
Zone Routing Protocol (ZRP)

The Zone Routing Protocol (ZRP) [34] is also called a hybrid ad hoc routing protocol as it combines proactive and reactive schemes. The ZRP maintains routing information for a local zone. It establishes routes, on demand for destinations beyond its logical neighborhood. It limits the scope of the local zone by defining a maximum hop number for the local zone (e.g. 3 hops). Using ZRP with a maximum hop count of zero for the local neighborhood creates a reactive routing algorithm, and using it with maximum hop count as infinity creates a pure proactive routing algorithm. A route to a destination within the local zone can be established from the proactively cached tables of the source node.

The routing algorithm used in the local zone can be based on any table driven routing algorithm, but it has to be extended in a way, that packets contain the “time to live” (TTL) information, which describes the maximum hop count of the local zone. For routes beyond the local zone, route discovery happens reactively. The source node sends a route request to its border nodes, containing its own address and a unique sequence number.

Border nodes are nodes, which are exactly the maximum number of hops to the defined local zone away from the source. The border nodes check their local zone for the destination. If the requested node is not a member of this local zone, the node adds its own address to the route request packet and forwards the packet to its border nodes. If the destination is a member of the local zone of the node, it sends a route reply on the reverse path back to the source.

The source node uses the path saved in the route reply packet to send data packets to the destination. The main advantage of the ZRP is a reduced number of required RREQ messages and further on the possibility to establish new routes without the necessity to completely flood the network.
2.2.3 Comparison of Routing Protocols

This section describes the classification of the routing algorithms presented so far, followed by detailed discussion on well known routing protocols. Further Table 2.5 summarizes the similarities and differences among some of the well-known routing protocols.

**TABLE 2.5 COMPARISON OF SOME AD HOC ROUTING PROTOCOLS**

<table>
<thead>
<tr>
<th>Protocol Property</th>
<th>DSDV</th>
<th>DSR</th>
<th>AODV</th>
<th>ZRP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multicasting</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Loop Free</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Multiple Routes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Distributed Operation</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Reactive</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Partially</td>
</tr>
<tr>
<td>Unidirectional Link Support</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Power Conservation</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Security</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>QoS Aware</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

DSDV is the only proactive protocol taken for comparison. It is also the protocol that has most common features with the traditional routing protocols in wired networks. The sequence numbers are added to ensure loop-free routes. DSDV will probably be good enough in networks, which allow the protocol to converge in reasonable time. This means that the mobility cannot be high.

The authors of DSDV came to the same conclusions and designed AODV, which is a reactive version of DSDV. They also added multicast capabilities,
which will enhance the performance significantly when one node communicates with several nodes. The reactive approach in AODV has many similarities with the reactive approach of DSR. They both have a route discovery mode that uses request messages to find new routes.

The difference is that DSR is based on source routing and will learn more routes than AODV. DSR also has the advantages that it supports unidirectional links. DSR has one major drawback as it is the source route that must be carried in each packet. This can be quite costly, especially when QoS in terms of issues.

ZRP is a very interesting proposal that divides the network into several zones. This approach is probably a very good solution for large networks. Within zones they have a more proactive scheme and between the zones they have a reactive scheme. They have many similarities with the operation of AODV and DSR. For instance the route discovery phase in them sends request through the network. In ZRP all zones are overlapping.

None of the presented protocols are adaptive, that is the protocols do not take any smart routing decisions when the traffic load in the network is taken into consideration. None of the protocols support power conservation or quality of service or security issues. So the existing protocols are to be enhanced to meet the above considerations.

Many researchers have been working on the protocol optimizations to reduce the overhead at different levels in order to achieve efficiency. DSDV is a well-known routing algorithm proposed for ad hoc network routing, but it has many problems as mentioned in previous sections. Currently, there are no standard specifications and commercial implementations available for DSDV, yet one DSDV simulator has been implemented with C++ at UC Berkeley [90]. Many improved protocols based on DSDV have been developed. Therefore DSDV is a
protocol under study. Thus, it will be substituted with its improvements in real time applications.

### 2.2.4 Previous Enhancements on DSDV

There are numerous enhancements in DSDV to improve the performance of DSDV. The review of some of the enhanced DSDV are discussed.

**Improvement of DSDV (I-DSDV)**

To improve the packet delivery ratio of DSDV routing protocol in mobile ad hoc networks with high mobility, a novel message exchange scheme for its invalid route reconstruction is proposed as improvement of DSDV (I-DSDV) [49, 66]. When route becomes invalid due to link breakage, the node that detects the link breakage tries to create a new loop-free route through message exchange with its neighbors. When route reconstruction in one-hop area is not accomplished, the area of message exchange for invalid route reconstruction is enlarged gradually on demand. It reduces the number of dropped data packets with little increased overhead at higher rates of node mobility.

**Multichannel version of DSDV (DSDV-mc)**

The basic idea proposed in [76] is to use multiple channels so that multiple useful transmissions can occur simultaneously, thus increasing network capacity. This requires minor changes to an existing proactive ad-hoc routing protocol and no modifications to the current IEEE 802.11 medium access control (MAC) protocol. To avoid inefficiencies due to periodic updates in the proactive routing protocol, this scheme divides the network layer into control and data planes.

Nodes sent routing updates using the control channel and user packets using the data channel. Thus the scheme extend the Destination-Sequenced Distance-Vector (DSDV) routing protocol to a multi-channel version, DSDV-MC.
Simulation results indicate that DSDV-MC exploits multiple channels to improve network capacity. Throughput increases in proportion to the number of available channels as the number of nodes and network load increase in both single-hop and multiple-hop networks.

**An Efficient DSDV (Eff-DSDV)**

An efficient DSDV (Eff-DSDV) [41, 42] protocol is proposed for ad hoc networks. Eff-DSDV overcomes the problem of stale routes, and thereby improves the performance than regular DSDV. The main objective is to provide a full duplex connectivity between the Ad hoc hosts and the hosts on the infrastructure network, using the Efficient Destination Sequenced Distance Vector routing protocol (Eff-DSDV).

One of the Ad hoc hosts is used as Mobile Internet Gateway (MIG) acting as a bridge between the MANET and the infrastructure network. The Full Duplex communication between MANET nodes and the Internet host is through this MIG. The MIG accesses the Switch in the wired network through an Access Point. The MIG is placed in close proximity to the Access Point. The MIG runs the Eff–DSDV protocol and takes care of the addressing mechanisms to ensure the transfer of packets between the hybrid networks.

**Randomized version of DSDV (RDSDV)**

Message routing requires mobiles to act as routers, by means of store and forward mechanisms. However, limitations on capabilities of mobiles require a control on node congestion due to message forwarding. A randomized version of the Destination-Sequenced Distance Vector (DSDV) [5] routing protocol, referred to as R-DSDV address the feasibility of congestion control using a congestion-related routing metric. The effectiveness of this protocol is evaluated through a probabilistic model.
**Multipath Destination Sequenced Distance Vector (MDSDV)**

The new protocol, called Multipath Destination Sequenced Distance Vector (MDSDV) [30] is compared with two known protocols DSDV and AODV. MDSDV finds disjoint paths which do not have any common nodes between a source and destination.

**Secure Destination Sequenced Distance Vector (S-DSDV)**

Secure routing protocol based on DSDV, namely S-DSDV[74], in which, a well-behaved node can successfully detect a malicious routing update with any sequence number fraud (larger or smaller) and any distance fraud (shorter, same, or longer) provided no two nodes are in collusion. It compares security properties and efficiency of S-DSDV with superSEAD.

The efficiency analysis shows that S-DSDV generates high network overhead, however, it can be reduced by configurable parameters. Thus the S-DSDV overhead is justified by the enhanced security. Further Table 2.6 summarizes the similarities and differences among some of the DSDV routing protocols.

Research work on improvement of DSDV is still active. It can be further enhanced to have the multicast capabilities in order to have secure communication among the nodes. In a secure communication in mobile ad hoc networks, the nodes mobility is significant, as a new node may join or an existing may leave the network. In order to achieve efficient routing information of each node, DSDV can also improved to be as mobility aware routing protocol. It can further enhanced to have better performance based on QoS metrics. Thus an effort is made in the same direction to achieve efficiency on DSDV protocol by overcoming the overheads.
2.3. Mobility Models for MANET

All the research in ad hoc networks till tries to increase the throughput of the network with minimum possible overheads. For efficient operation of the routing protocols[27], it is necessary that the correct network topology information is available to the nodes in the network so that packets could be forwarded correctly between a sender and a receiver. In order for the nodes to have a correct
topological view of the network, it is required that the nodes have reasonably correct location information of the nodes in their neighborhood.

If the nodes are stationary, then their location information remains unchanged which leads to a correct topological view of the network and routing should work well as it does in the wire-line networks. However in ad hoc networks, the nodes move to different location, thus creating a different network topology. Almost all well-known routing protocols are shown to perform poorly for a network where the topology is changing at random.

With the realization of importance of node mobility in the routing process of ad hoc networks, quite a considerable level of work has been done in mobility characterization of the mobile nodes. This mobility characterization research is attempting to quantify the randomness in the mobility of the nodes [52, 53]. Most of the research in this area of mobility characterization has however been towards mobility characterization of individual nodes [48].

Considering individual node mobility is of minor significance in ad hoc networks as the nodes in the ad hoc network show cohesive properties. This observation is directly related to the every existence of ad hoc networks to support group collaboration and/or group activities. The effect of movement of neighboring nodes on a movement of an individual node has been recently reported in [75]. In this paper authors argue about the importance of treating the mobility of mobile nodes as a group or the network as a whole, over the mobility model followed by an individual. However the paper does not throw any light on how the group mobility information can be used in improving the performance of routing algorithms.

Most of the routing algorithms proposed so far, merely have been an attempt to mitigate the effects arising due to mobility in two ways. As one by letting the nodes in the network constantly update each other of the latest
neighborhood information and other by flooding. In both ways, it leads to increased overhead. Most of the algorithms so far have overlooked the main cause, which leads to this instability in the topology that is the mobility of the nodes.

Thus the mobility of the nodes render the network topology in a state of flux, resulting in the degradation in performance of the ad hoc networks. Hence, in this research, an attempt is made to resolve the issue of mobility by trying to define mobility into various models. There by attempting to characterize the mobility of the nodes into some pre-identified mobility models where the movements can be approximated to simple movement pattern and then can use this information to bring in predictability to motion.

2.3.1 Mobility Models

A mobility model [89] is used to capture the movement of objects in simulation. In MANET, a mobility model is used to define the movement of a mobile wireless node. There are two types of MANET mobility models: single-entity and group as shown in Figure 2.6.

In single entity models, each mobile node moves independently of all the other nodes within the network area. In group mobility models, nodes are assumed to be organized in groups and the mobility of a node is often reflective of the movement pattern of the entire group.

For simplicity, most of the mobility models are defined for a rectangular network area enclosed by (0, 0), (0, ymax), (xmax, ymax), and (xmax, 0). A characteristic feature of every mobility model is to ensure that a mobile node will not travel outside the network area. In this section, three mobility models are used which are designed to capture a wide range of mobility patterns for ad hoc applications. Mobility models are chosen for simulation based on their different classes of motion as random based and group based movements.
Random based Mobility Models

In random based mobility models, the mobile nodes move randomly and freely without restrictions. The destination, speed and direction are all chosen randomly and independently of other nodes. The different types are discussed below:

(a) **Random Waypoint Mobility model:** The random way point[81] mobility model is simple and is widely used to evaluate the performance of MANETs. The random way point mobility model contains pause time between changes in direction and/or speed. Once a Mobile Node (MN) begins to move, it stays in one location for certain period of time called pause time. After the specified pause time is elapsed, the mobile node randomly selects the next destination in the simulation area.

Figure 2.7 shows the traveling pattern of mobile nodes using random way point mobility model. The mobile node chooses a speed uniformly distributed between the minimum speed and maximum speed and travels with a speed ‘v’
whose value is uniformly chosen in the interval $(0, V_{\text{max}})$. $V_{\text{max}}$ is some parameter that can be set to reflect the degree of mobility.

![Traveling pattern of mobile nodes using Random way point mobility model](image)

**Figure 2.7 Traveling pattern of mobile nodes using Random way point mobility model**

Then, the MN continues its journey toward the newly selected destination at the chosen speed. As soon as the MN arrives at the destination, it stays again for the indicated pause time before repeating the process. This model is used in mobility management schemes for mobile communication systems.

**b) Random Walk Mobility Model**: This model is similar to the random waypoint, but at a trip transition instant, a node picks direction, trip duration and numeric speed. The node moves in the given direction with the given numeric speed for the given trip duration. Each movement in the Random Walk mobility model occurs in either a constant time interval ‘$t$’ or a constant distance traveled ‘$d$’, at the end of which a new direction and speed are calculated.

If a mobile node moving according to this model reaches a boundary area, it bounces off the boundary border with an angle determined by the incoming direction. The mobile node then continues along this new path. The Random Walk mobility model is a memory less mobility pattern because it does not retain
knowledge concerning its past locations and speed values. Figure 2.8 shows the traveling pattern of mobile nodes using random walk mobility model in the application area of node movement.

![Figure 2.8 Traveling patterns of mobile nodes using Random walk mobility model](image)

**Figure 2.8 Traveling patterns of mobile nodes using Random walk mobility model**

**Group Mobility Model**

Group mobility model[35] represents multiple mobile nodes whose actions are completely independent of each other. For example, a group of soldiers in a military scenario may be assigned the task of searching a particular plot of land in order to destroy land mines. In order to model such situations, a group mobility model is needed to simulate this kind of characteristic.

Each group has a logical centre (group leader) that determines the group’s motion behavior. Initially each member of the group is uniformly distributed in the neighborhood of the group leader. Subsequently, at each instant, every node has speed and direction that is derived by randomly deviating from that of the group leader. Each node deviates from its velocity (both speed and direction) randomly from that of the leader.
This section also describes a sampling of the mobility models that have been designed specifically for ad hoc networks. Classification and survey of existing mobility models are given in [4]. Since tactical network consist of mobile devices, the mobility models used has a decisive impact. A recent study shows that the average speed of a node using Random Waypoint decreases with time, and hence the results obtained using this model becomes unreliable as the simulation advances [87]. In [31], the effect of mobility models on the performance of mobile ad hoc network using unicast routing protocol is discussed.

The IMPORTANT framework [7] characterizes movement based on spatial dependence, relative speed, and other factors and illustrates how these metrics impact unicast routing performance. In [38] the authors have shown that the mobility model used can significantly impact the performance of ad hoc routing protocols, including the packet delivery ratio, the control overhead and the data packet delay.

The performance of two multicast routing protocols ODMRP and ADMR for mobile ad hoc networks with different mobility models are compared in [51]. The difference in the performance is analyzed widely across the different mobility models. The performance of AODV with effect of random mobility models patterns are compared in [32]. Here the performance is analyzed using varying network load, random based mobility model and network size.

In [58], the simulation done to evaluate the performance of multicast tree algorithms for MANET. In [59], the simulation done to evaluate the performance of multipath routing protocols for MANET. Both include performance metric as lifetime per multicast tree and multipath set respectively. Mobility framework called Dispersion mobility model [67] organizes mobile nodes as group of clusters and evaluates performance under different mobility patterns and for different implementations.
2.4. Chapter Summary

Secure multicast communication is a significant requirement in emerging applications in ad hoc environments like military or public emergency network applications. Membership dynamism is a major challenge in providing complete security in such networks. Key management plays a vital role in secure communication. The summary of different key management approaches are presented briefly with examples.

As the nodes are dynamic in nature, ensuring effective routing is one of the major challenges for MANET. Some of the well known routing protocols are discussed indicating the differences, advantages and disadvantages based on the desired routing properties of MANET. The previous enhancement of DSDV routing protocol is also discussed. The need for mobility pattern for dynamic network topology and its different mobility models for MANET are presented. These directions paved the way to propose a novel methodology for efficient secure multicast communication in MANET.

Publication of Related Works :

- International Journal