Secure Addressing Protocols for Mobile Ad Hoc Networks

Chapter · March 2019

DOI: 10.1201/9781003010463-11

CITATIONS
2

READS
114

4 authors, including:

Uttam Ghosh
Meharry Medical College
176 PUBLICATIONS **1,541** CITATIONS

SEE PROFILE

Pushpita Chatterjee
SRM Institute of Science and Technology
41 PUBLICATIONS **319** CITATIONS

SEE PROFILE

Danda B Rawat
Howard University
400 PUBLICATIONS **6,491** CITATIONS

SEE PROFILE

Chapter 8

Secure Addressing Protocols for Mobile Ad hoc Networks

Uttam Ghosh
Vanderbilt University

Pushpita Chatterjee
Old Dominion University

Raja Datta
IIT Kharagpur

Al-Sakib Khan Pathan
Southeast University

Danda B. Rawat
Howard University

Contents

8.1 Introduction

Internet of Things (IoT) is a heterogeneous network of physical devices and other objects embedded with sensors and actuators. It enables the objects to connect and communicate a large amount of data to offer a new class of advanced services available at anytime, anywhere, and for anyone. IoT consists of various types of wireless networks such as wireless sensor and ad hoc networks (WiFi, ZigBee, and RFID) to make the physical infrastructures such as buildings (homes, schools, offices, factories, etc.), utility networks (electricity, gas, water, etc.), transportation networks (roads, railways, airports, harbors, etc.), transportation vehicles (cars, rails, planes, etc.), healthcare systems, and information technology networks smarter, secure, reliable, and fully automated. It collects, stores, and communicates a large volume of heterogeneous data from various types of networks and provides critical services in manufacturing, healthcare, utility, and transportation networks.

A *Mobile Ad hoc Network* (MANET) is a collection of devices equipped with wireless communications and networking capability [1]. These nodes can be arbitrarily located and are free to move randomly at any given time. Node mobility can vary from almost stationary to constantly moving nodes. Network topology and interconnections between the nodes can change rapidly and unpredictably. As MANET is infrastructure-less and highly dynamic in nature, unique node identification is a very important requirement in such setting. This is required for a node to participate in unicast communications and routing, and also to retain its identity when a network gets partitioned and/or merged due to its dynamicity.

Originally, MANET was conceived as a small isolated ad hoc network that does not have any connection to the outer world. However, over the years, MANET has evolved into an important network having applications in various fields that requires communication with other infrastructure networks. This has necessitated MANET's connectivity with the Internet, albeit in a restricted manner. This means that usually, MANET will work in isolation but occasionally may connect to an infrastructure network (e.g., Internet or IoT) if need arises. This brings us to an important issue of node identification in accordance with the usual IP network. Moreover, IP address facilitates multi-hop routing in the network and also when the network is connected with IoT [2].

Dynamic Host Configuration Protocol (DHCP) [3] provides static or dynamic address allocation to the network nodes that can be manual or automatic. Manual or static address configuration in most cases is inapplicable as the nodes in MANET are highly mobile leading to partitioning/merging of networks. Further, the centralized DHCP is not a suitable solution, as it has to maintain configuration information of all the nodes in the network.

The address allocation protocol needs to consider several challenges and dynamic scenarios as MANET is a distributed and dynamic network. In the simplest scenario, a node can join and leave the network at any time as shown in Figure 8.1. A new node *N* needs an IP address whenever it joins the network. An allocated IP address can be reused when the node departs from the network (the node either switches off itself or leaves the network due to node mobility).

In MANET, the mobile nodes are free to move arbitrarily and one or more configured nodes go out of others' transmission ranges for a while. As a result of this, the network may get partitioned as shown in Figure 8.2a. These partitions may grow independently and continue communications with their IP addresses. Due to node mobility, these partitions may merge later. If a new node *N* joins a partition and obtains an IP address belonging to the other partition, then conflict occurs when these two partitions merge as presented in Figure 8.2b.

Figure 8.3 shows another scenario where two separately configured MANETs merge. There may be two different cases. In the first case (for instance), there is no address conflict as both MANET 1 and MANET 2 have different network identifiers. In the second case, there are some duplicate addresses as the address allocation

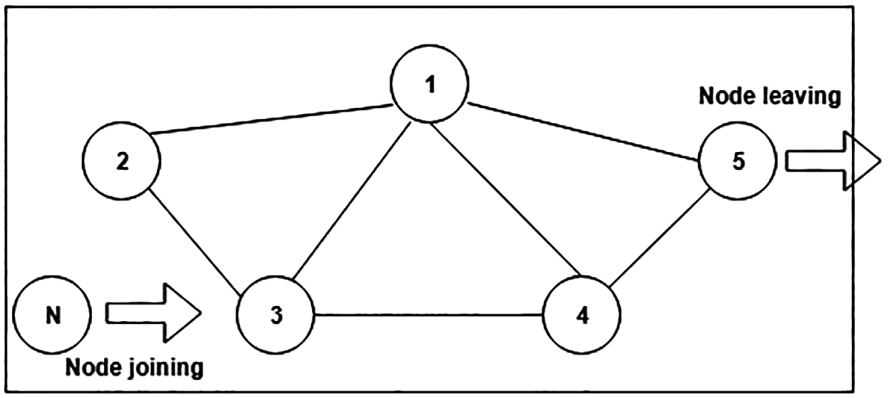

Figure 8.1 Node joining and leaving in MANET.

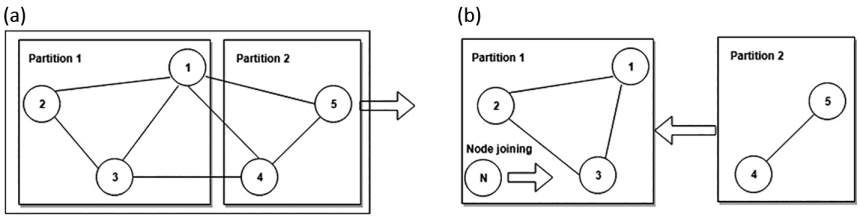

Figure 8.2 (a) Network partitions; (b) network merging.

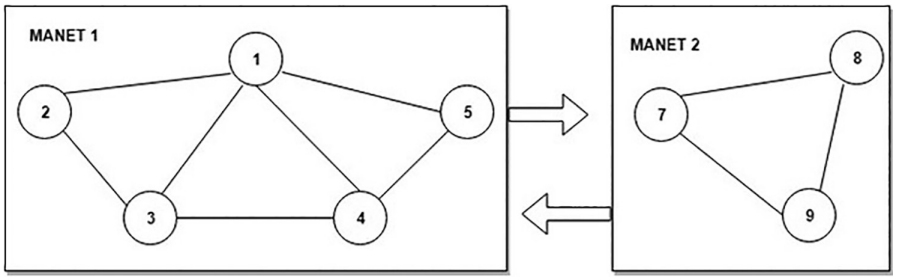

Figure 8.3 Network merges of two independent MANETs.

in MANET 1 is independent of MANET 2. As a result of this, some (or all) nodes in one MANET may need to change their addresses.

It can be seen from the above discussion that there is a need to design a secure distributed addressing protocol for MANETs that can provide compatible connectivity with IoT. Thus, for assigning addresses in MANETs, a standard addressing protocol should have the following objectives [2,4]:

- **Distributed Dynamic Address Configuration**: As MANET is infrastructure-less, and its nodes are mobile in nature, the addressing protocol for providing dynamic IP addresses to the new nodes should be a distributed one (rather than centralized). If the addressing protocol is centralized, then it has to rely heavily on duplicate address detection (DAD) mechanism for address resolution in the network. This causes broadcast storm problem.
- **Uniqueness**: The protocol should assign unique IP addresses to the nodes of the network for correct routing and for point-to-point communication.
- **Robustness**: The chances of address conflicts among the nodes due to network partitioning and network merging should be as low as possible.
- **Scalability**: As the network grows, the time taken to obtain an IP address (i.e., *addressing latency*) and the number of message exchanges (i.e., *addressing overhead*) during address allocation should be minimum.
- **Security**: The protocol should be able to withstand attacks while allocating addresses to the nodes of a MANET. The major security threats associated with dynamic IP address configuration in MANET are as follows:
  - **Address spoofing attack**: In this attack, an IP address of a node can be spoofed by a malicious host to hijack the network traffic.
  - **Address exhaustion attack**: Here, an attacker may claim as many IP addresses as possible for exhausting all the valid IP addresses so as to prevent a newly arrived node from getting an IP address.
  - **False address conflict attack**: In this attack, an attacker may transmit address conflict messages falsely so that a victim node may give up its current address and seek for a new one.

  - **False deny message attack**: Here, an attacker may continuously transmit false deny messages to prevent a newly arrived node from getting an IP address.
  - **Reply attack**: In this type of attack, an attacker injects previously captured packets (from an authorized node) into the network. This attack can also be seen at other layers of the protocol stacks.
- **Organization of the chapter**: This chapter presents the main challenges related to addressing protocols in dynamic and ad hoc environment like MANET. It further categorizes the existing addressing protocols and provides a detailed overview of them in Section 8.2. In Section 8.3, the chapter also evaluates and compares the performance of addressing protocols of MANET through a set of performance metrics. Finally, it concludes with future research direction related to addressing in MANET.

8.2 Address Allocation Protocols for MANET

This section presents a brief review on recently proposed dynamic address allocation protocols [2,5,6,7–29] for MANET to enable proper communication in the network. In order to adapt to the dynamic environment of a MANET, these protocols bear many similarities to each other, such as self-organizing, self-healing behavior. However, these approaches also differ in a wide range of aspects, such as address format, address allocation information, usage of centralized servers or full decentralization, hierarchical structure or flat network organization, and explicit or implicit DAD mechanism. According to [30], all the existing IP address allocation schemes for ad hoc networks can be classified into *stateless allocation* and *stateful allocation* approaches.

8.2.1 Stateless Allocation Approaches

In *stateless* allocation approaches, nodes in a network do not store any address allocation information. These approaches are mainly based on self-configuration. Each node chooses its address randomly and then performs DAD to ensure uniqueness of the chosen address within the network. The major disadvantages of stateless approaches are high addressing overhead and latency.

Most of the existing address allocation algorithms for a MANET use DAD mechanism [5] to resolve address conflict in the network. Figure 8.4 shows the DAD process. DAD is required whenever a new node joins a MANET or independent networks merge. A new node picks up a tentative IP address and uses DAD process to determine whether this address is available or not. All the nodes having a valid IP address participate in DAD mechanism to protect their IP addresses being used by a new node. In order to detect the uniqueness of the address, the node sends a *Duplicate Address Probe* (DAP) message and expects an

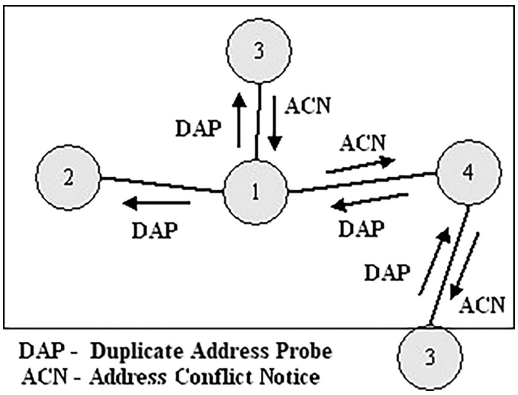

Figure 8.4 DAD mechanism.

Address Conflict Notice (ACN) message back in a certain timeout period. If no ACN message is received, the node may assume that the address is not in use.

In [5], the authors propose a mechanism called weak DAD. The basic idea of weak DAD is that the duplicate addresses may be tolerated as long as packets reach the destination node intended by the sender even if the destination node's address is also being used by another node. Here, each node has a unique key which is included in the routing packets and in the routing table entries. Even if the IP address conflicts exist, they can be identified by their unique keys. The main drawback of weak DAD is its dependency on the routing protocol and also traffic overhead caused by the integration of the key value in routing packets.

In *passive DAD* [6], nodes use periodic link state routing information to notify other nodes about their neighbors. This is usually very costly and will result in serious redundancy, contention, and collision, which is called *broadcast storm* problem [31].

The *ad hoc address autoconfiguration* (AAA) scheme [7] under the stateless allocation approaches uses randomly selected addresses from the address range of 169.254/16. Thereafter, to guarantee the uniqueness of the selected address, DAD is performed by each node of the network. During this process, an *Address Request* message is flooded in the network to query the usage of its tentative address. If the address is already in use, an *Address Reply* message is unicast back to the requesting node. The requesting node then re-initiates the address allocation process. However, this approach does not consider complex scenarios such as network partitions and merger.

Fazio, Villari, and Puliafito have proposed an *automatic IP address configuration* (AIPAC) scheme [8] for MANET. In the AIPAC address allocation scheme, there are two phases. In the initialization phase, two distinct nodes enter each other's range by using the Host Identifier (HID). The AIPAC then locates the nodes without a valid IP address using the HID. The node that has the higher HID then selects the configuration parameters for both the interacting nodes. When a new

node (Requester) wants to join the network, it receives the IP address through one of the configured nodes (Initiator). The Initiator negotiates for the Requester's valid IP address in the allocation phase, corrects the configuration, and then offers it to the Requester.

In [9], Wang, Reeves, and Ning have proposed a secure auto-configuration scheme that uses self-authentication technique. By using one-way hash function, it binds a node's address with a public key. Address owner can use the corresponding public key to unilaterally authenticate itself. In this scheme, whenever a node, P, wants to join the network, it first randomly generates a public/private key pair and a secret key. Thereafter, node P calculates a 32-bit (in IPv4) or a 128-bit (in IPv6) hash value of the public key, that is, $IP = H(public\ key)$, where H is a secure one-way hash function. Next, node P temporarily uses this IP address, initiates a timer, and broadcasts a DAP message throughout the network to verify the uniqueness of the IP address. If a configured node (call it Q) finds that the IP address in a received DAP message is same as its own, it verifies the authenticity of this DAP message. If the verification result is positive, Q then broadcasts an ACN message to inform the corresponding node P of the address conflict. Otherwise, node Q simply discards the received DAP message. The scheme handles network partitioning/merging by employing the concept of passive DAD mechanism.

Passive autoconfiguration for mobile ad hoc networks (PACMAN) [10] has been proposed by Weniger where a new node selects an address using a probabilistic algorithm. In order to verify the uniqueness of the addresses, it makes use of passive DAD in conjunction with a distributed maintenance of a common allocation table. When it detects that two nodes are using the same IP address, it reports the problem to one of them using a unicast message to change its address. Moreover, it takes into account the problem of changing an address that has some ongoing communication. To fix this, while changing an address, a node notifies other nodes (with whom communication is going on) of its new IP address, so that they can make an encapsulation of the messages properly.

8.2.2 Stateful Allocation Approaches

In *stateful* allocation approaches, the nodes in the network keep track of assigned and free addresses for address assignment as well as network management. Each node in *MANETconf* scheme [11], presented by Nesargi and Prakash under the stateful allocation approach, maintains a list of IP addresses which are in use in the network. A new node X obtains an address through an existing node Y in the MANET; thereafter, the node Y performs an address query throughout the MANET. This address allocation requires a modified DAD for checking address duplication. Here, a positive acknowledgment (ACK) is required from all the known nodes indicating that the IP address is available for use. This may result in an ACK explosion. Network partitions and mergers are detected throughout the modified DAD.

In [12], Zhou, Ni, and Mutka propose a scheme called Prophet for MANET. The scheme proposes a function $f(n)$ to generate a series of random numbers for address allocation. The desired properties of function $f(n)$ are as follows:

i. A sequence satisfies the extremely long interval between two occurrences of the same number.
ii. The probability that the function returns the same number for two different state values is very low.

The protocol works as follows: The first node M in the MANET generates a random number and sets its IP address. It also uses a random state value as a seed for its function $f(n)$. Another node N can get its IP address from node M along with a state value as a seed for its $f(n)$. Whenever a node joins the network, same process continues for address allocation of new nodes.

Figure 8.5 shows an example of Prophet address allocation where $f(n)$ = (address*state*11) mod 17 and the effective address range is [1, 16]. The first node M randomly generates 5 as its address and the seed. When N enters, node M changes its state of $f(n)$ to 3 (=5*5*11 mod 17) and assigns it to N as address and seed. Node O and node P get addresses 12 (=3*5*11 mod 17) and 14 (=3*3*11 mod 17), respectively. The main advantages of this protocol are its low addressing latency and low communication overhead. However, its major drawback is that even with a large address space, address conflicts may exist in the network. To resolve these address conflicts, it requires mechanisms such as passive DAD or weak DAD.

In [13], a *Dynamic Address Configuration Protocol* named DACP has been presented for MANET. In DACP, an elected address authority maintains the state information of MANET. In the initialization phase, a node chooses two addresses: *temporary* and *tentative* addresses. The temporary address is used to verify the uniqueness of the tentative address using DAD. The tentative address is made

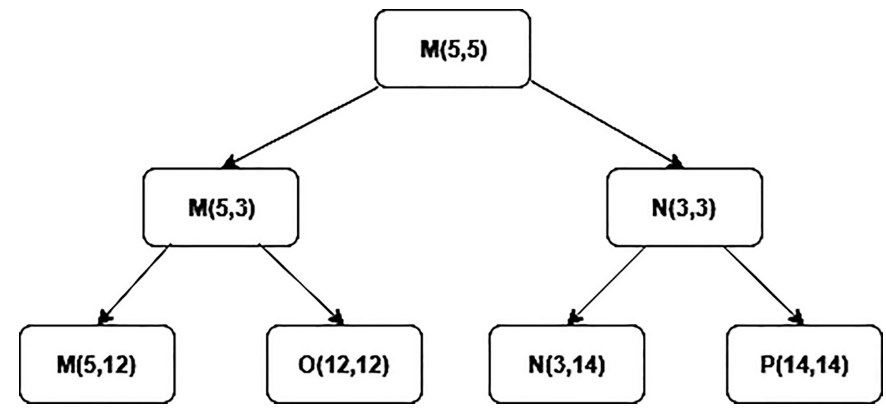

Figure 8.5 An example of Prophet address allocation.

permanent after it is found that the same is not a duplicate address. After successful address configuration is made, a new node registers its tentative address with the address authority to make it permanent. Here, the new node waits for an advertisement from the address authority for a certain period of time. After receiving the said advertisement, the new node sends a registration request and waits for the registration confirmation message from the address authority. The concerned new node can use this address only after this confirmation message is received. The overhead due to duplication detection mechanism and the high periodic flooding of the messages due to registration with address authority are the major drawbacks of this protocol.

Another protocol used for dynamic address allocation is the optimized DACP [14], in which, instead of discovering the server, the server itself periodically broadcasts the address request to reduce the overhead of broadcasting. However, the latency for the hosts to obtain the addresses in this scheme is much higher.

Taghiloo et al. have presented an address auto-configuration based on *virtual address space mapping* (VASM) scheme [15]. According to this protocol, when a new node wants to join the network, it sends a single-hop message in order to find an *Initiator*. If no reply message is received for this packet, it assumes that it is the only node in that network and starts the network setup process. If the node receives more than one response, it selects the sender of the first arrived packet as *Initiator* and sends it an address request packet. The main task of Initiator is to obtain a new IP address from its *Allocator* and assign it to the requesting node (*Requester*). In order to balance the overhead of protocol traffic and minimize the addressing latency, an *Allocator* can create another *Allocator* in the network for generating unique addresses for the new nodes. For an efficient management of network events (such as merging and partitioning), each *Allocator* holds a list of all *Allocators* in the network. As the number of *Allocators* in each network is limited, the size of *Allocators*' list will be very small. Tajamolian, Taghiloo, and Tajamolian have proposed a lightweight secure address configuration scheme [16]. For address allocation, VASM address configuration scheme is used [15]. To secure the address allocation, the scheme uses a zero knowledge approach.

In [17], X. Chu et al. proposed a quadratic residue-based address allocation scheme for mobile ad hoc networks. In this scheme, when a new node wants to join a MANET, it sends a DISCOVER message to obtain an IP address. If no reply is received, it assumes that it is the first node in the network. The node configures itself with an IP address and also generates an address block. Thereafter, a new node can obtain an address and also an address block from a configured node. In this way, network grows up from one to many in the network. To handle network merging and partitioning, this scheme uses DAD to remove duplicate addresses from the network.

In the *Buddy* system allocation scheme [18], each node maintains a block of free addresses. A configured node which receives an *Address Request* from a new node assigns the requesting node an IP address from its block of free addresses. It also

divides its block of free addresses into two equal parts and gives one half to the requesting node and the other half it keeps with itself for future use. However, it is always difficult for the individual nodes to manage such type of address blocks in a MANET and is also complex to implement. Cavalli and Orset presented a secure hosts auto-configuration scheme [19]. The scheme employs the concept of challenge, where a node has to answer a question to prove its identity. Here, a new node X sends a request with its public key and a temporary identifier to its neighbors. The neighbors then calculate a nonce, cipher it with the public key of the node X, and then return it to the node X. The new node X, after having deciphered it with its private key, returns the nonce to the concerned nodes. It uses the *Buddy* system technique to allocate the IP addresses.

Another dynamic address configuration scheme called Prime DHCP [20] is proposed by Hsu and Tseng. In this scheme, address can be allocated to the new host without broadcasting it over the whole of MANET. Prime DHCP makes each host a DHCP proxy and runs a prime numbering address allocation (*PNAA*) algorithm individually to compute unique addresses for address allocation. According to *PNAA* algorithm, the first node that creates the network is called the root node. The *PNAA* algorithm works based on the following two rules:

i. The root node having address 1 can allocate prime numbers in an ascending order.
ii. Other nodes (not the root) can allocate addresses equal to its own address multiplied by the unused prime number, starting from the largest prime factor of its own address.

Figure 8.6 shows an example of *PNAA* address allocation tree. For example, let us take the node having address 9. As the largest prime factor of 9 is 3, it can allocate the sequence of addresses 9 * 3, 9 * 5, 9 * 7, and so on up to the largest address bounded by address space. Therefore, it eliminates the need of DAD mechanism. Prime DHCP [20] uses Destination-Sequenced Distance-Vector (DSDV) [32] routing protocol in order to handle network partitions and mergers.

MMIP [21], ADIP [22], IDDIP [23], IDSDDIP [24], and SD-RAC [2] are the address allocation schemes proposed by Ghosh and Datta, where the nodes of the network act as proxies and are able to allocate addresses independently to the new nodes. None of these schemes needs to run DAD mechanism to verify the uniqueness of addresses in the network. In order to provide authentication, MMIP binds the hardware address with the IP address at the time of address allocation. The authentication for address configuration is done with the help of a trusted third party in case of ADIP scheme, whereas IDDIP scheme uses self-authentication technique. IDSDDIP [24] provides security using a RSA-based cryptography system, and SD-RAC [2] uses a bilinear pairing-based signature scheme.

Figure 8.7 partially shows an example of how unique IPv6 address can be allocated by a node acting as proxy in SD-RAC [2]. For simplicity, they present the

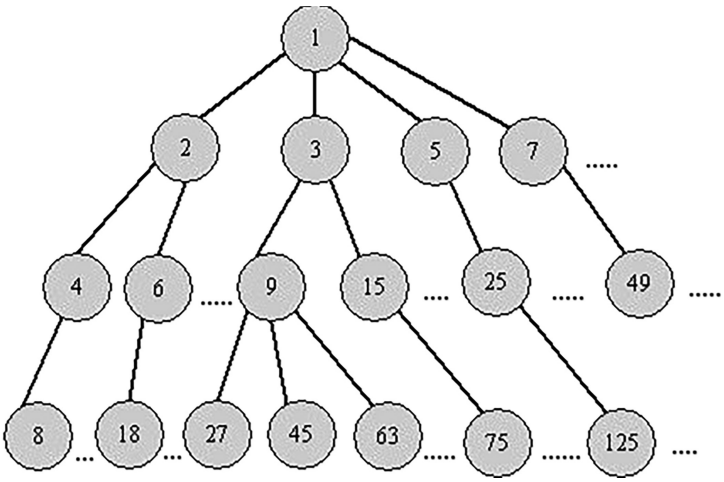

Figure 8.6 An example of *PNAA* address allocation tree.

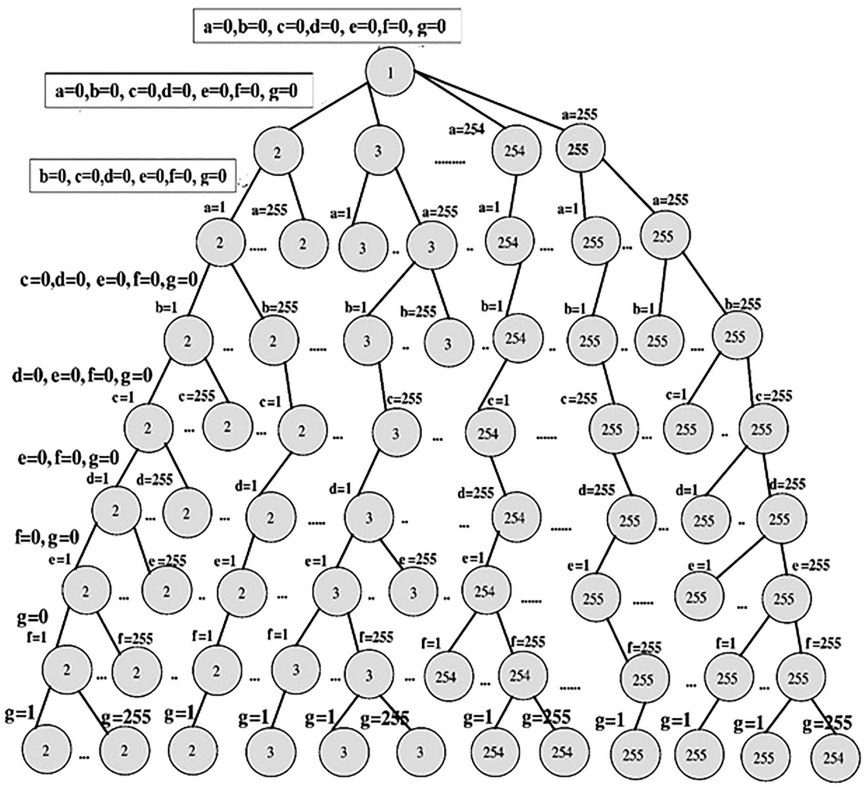

Figure 8.7 An example of SD-RAC address allocation.

addresses in dotted decimal format, that is, m.n.o.p.q.r.s.t.a.b.c.d.e.f.g.h wherein m.n.o.p.q.r.s.t is the network prefix (fixed for a network) and a.b.c.d.e.f.g.h is a host identifier (a, b, c, d, e, f, and g are variables). In the figure, the last byte (h) of an IP address is shown within the circle and the other bytes (a, b, c, d, e, f, and g) are shown outside the circle. The root proxy has an IP m.n.o.p.q.r.s.t.0.0.0.0.0.0.0.1, and the host identifier addressing it can assign a range from 1.0.0.0.0.0.0.1 to 255.0.0.0.0.0.0.1 and from 0.0.0.0.0.0.0.2 to 0.0.0.0.0.0.0.255 with network prefix m.n.o.p.q.r.s.t. A proxy having host identifier 1.0.0.0.0.0.0.1 can allocate addresses from 1.1.0.0.0.0.0.1 to 1.255.0.0.0.0.0.1. This scheme can allocate addresses from m.n.o.p.q.r.s.t.0.0.0.0.0.0.0.1 to m.n.o.p.q.r.s.t.255.255.255.255.255.255.255.254 in the network uniquely.

A *filter-based addressing protocol* (FAP) [25] has been proposed by Fernandes et al., which maintains a distributed database stored in filters containing currently allocated addresses in a compact fashion. Bloom filter is used to detect and resolve network partitioning and merging, and sequence filter is used to obtain IP addresses. Then, a node acquires an IP address based on sequence filter, and it needs to run a DAD in the whole network to update sequence filters of all other nodes residing in the network. Here too, we find that the addressing overhead and latency would be high due to DAD process running in the entire network.

8.3 Performance Comparison and Analysis

Table 8.1 presents the qualitative comparison of the existing dynamic address allocation approaches for MANETs [2,33]. Here, n is the total number of mobile nodes, l is the average number of links between nodes, d is the network diameter, and t is the average 1-hop latency. The following metrics are used to analyze the performance of the existing allocation protocols.

Uniqueness: One of the most important metrics in address allocation technique is to guarantee the uniqueness of the allocated addresses. Stateless protocols (i.e., AIPAC [8], PACMAN [10], AAA [7], Wangs scheme [9]) cannot guarantee the uniqueness of allocated addresses. On the contrary, stateful protocols can do that (e.g., MANETconf [11], ODACP [14], Prime DHCP [20], Buddy [18], Cavallis scheme [19], ADIP [22], IDDIP [23], FAP [25], IDSDDIP [24], and SD-RAC [2]).

Addressing Overhead: Addressing overhead of an addressing protocol includes address allocation overhead and network maintenance overhead. Allocation overhead refers to the average number of messages required for an address allocation to a new node. Maintenance overhead refers to the average number of messages received by a node per time slot to maintain the network. In stateless protocols, DAD needs to run throughout the network to verify the uniqueness of an assigned IP address to a new node. Therefore, the allocation overhead for stateless protocols due to flooding is $O(n^2)$. Stateless protocols do not rely on the underlying routing protocols and prefer to use the reactive routing protocol (e.g., AODV [26]) for the

Table 8.1 Performance Comparison of Address Allocation Protocols

Protocol	*IP Family*	*Allocation Approach*	*Addressing Latency*	*Overhead*		*Complexity*	*Robustness*	*Scalability*	*Security*
				Allocation	*Maintenance*				
MANET CONF	IPV4	Stateful	O(2td)	$O(n^2)$	O(2l/n)	High	Low	Low	No
FAP	IPV4	Stateful	O(2td)	$O(n^2)$	O(2l/n)	High	Low	Low	No
AIPAC	IPV4	Stateless	O(2td)	$O(n^2)$	O(2l/n)	Medium	Low	Low	No
PACMAN	IPV4, IPV6	Stateless	O(2td)	$O(n^2)$	O(2l/n)	Medium	Low	Low	No
AAA	IPV4	Stateful	O(2td)	$O(n^2)$	O(2l/n)	Medium	Low	Low	No
ODACP	IPV4	Stateful	O(2td)	O(2l)	O(2l)	Low	Low	Low	No
PROPHET	IPV4	Stateful	O(2t)	O(2l/n)	O(2l/n)	High	Low	Medium	No
PRIME DHCP	IPV4	Stateful	O(2t)	O(2l/n)	O(2l/n)	Low	Medium	Medium	No
BUDDY	IPV4, V6	Stateful	O(2t)	O(2l/n)	$O(n^2)$	High	Medium	Low	No
CAVALI [19]	IPV4, IPV6	Stateful	O(2t)	O(2l/n)	$O(n^2)$	High	Medium	Low	Yes

(*Continued*)

Table 8.1 (*Continued*) Performance Comparison of Address Allocation Protocols

Protocol	*IP Family*	*Allocation Approach*	*Addressing Latency*	*Overhead*		*Complexity*	*Robustness*	*Scalability*	*Security*
				Allocation	*Maintenance*				
WANG [9]	IPV4, IPV6	Stateless	O(2t)	$O(n^2)$	O(2l/n)	Medium	Low	Low	Yes
ADIP	IPV4	Stateful	O(2t)	O(2l/n)	O(2l/n)	Low	Medium	Medium	Yes
IDDIP	IPV4	Stateful	O(2t)	O(2l/n)	O(2l/n)	Low	Medium	Medium	Yes
IDSDDIP	IPV6	Stateful	O(2t)	O(2l/n)	$O(n^2)$	Low	Medium	High	Yes
SDRAC	IPV6	Stateful	O(2t)	O(2l/n)	O(2l/n)	Low	High	High	Yes

network maintenance. The maintenance overhead of stateless allocation protocols is $O(2l/n)$ as the reactive routing protocol needs to exchange a one-hop broadcast message to track the neighbors periodically.

On the other hand, most of the stateful address allocation protocols need to send a one-hop broadcast message to get an address from the network. Therefore, allocation overhead for these protocols is $O(2l/n)$. However, the allocation overhead of MANETcon [11] and FAP [25] is $O(n^2)$. This is because both of the protocols require a positive ACK from all known nodes indicating that the address is available for use. The allocation overhead of ODACP [14] is $O(2l)$ as each node has to register with an address authority. Stateful allocation protocols use either a reactive routing protocol or a proactive routing protocol for the network maintenance. Therefore, if the addressing protocol uses the proactive routing protocol, then the network maintenance overhead is $O(n^2)$; otherwise, the overhead is $O(2l/n)$.

Addressing Latency: Addressing latency is the time between points when a node requests for an address and when it acquires the address from the network. DAD mechanism is used by stateless allocation protocols where an *Address Request* message is flooded in the whole network. A timer for Address Reply is set based on the network diameter, and a new node can configure itself when the timer times out. Therefore, the addressing latency of the stateless protocols is $O(2td)$.

In stateful allocation, most of the protocols send a one-hop broadcast message for an address allocation; hence, their addressing latency is the round-trip delay time, that is, $O(2t)$. However, the addressing latency of FAP [25], ODACP [14], and MANETconf protocols is $O(2td)$. The addressing protocols that consider the security aspects have an additional latency.

Complexity: The addressing protocol should use the network resources (e.g., power and memory of nodes, bandwidth of the network) as minimal as possible during address allocation. The complexity of the stateless allocation protocols is considered to be medium as they generate addresses from some random numbers. Under the stateful address allocation, FAP [25], MANETconf [11], Prophet [12], Cavalli et al. [19], and Buddy [18] protocols are highly complex to implement and synchronize the addresses among the nodes in the network. In ODACP [14], each node needs to register with the centralized leader node and eliminates the need of flooding the messages, hence reducing the message complexity. Other protocols under the stateful category have low computational complexity as these protocols do not require maintaining the address blocks and complex functions to generate addresses. These protocols generate addresses by the existing network nodes acting as proxies. This reduces the complexity and memory requirements of these protocols even further.

Robustness: Robustness refers to the adaptability of an addressing protocol in a dynamic network environment, including network partitioning and merging. The addressing protocol is considered to be highly robust if it can guarantee the address uniqueness even when the network gets partitioned and merges again. Here, two scenarios need to be considered when a new node joins the network. In the first

scenario, a network grows independently and never partitions into multiple networks or merges with the other partitions or networks. In the second scenario, a network starts independently, and subsequently, the network partitions into multiple networks due to node mobility. Further, new nodes can join any partition and these partitions may merge with each other. In stateless allocation, Prophet and FAP, the addresses are generated randomly; hence, there is a chance of address conflict in both of the above scenarios. Therefore, the robustness of these protocols is considered to be low. On the other hand, most of the stateful allocation protocols ensure address uniqueness in the first scenario, but the address conflict may exist in the second scenario. Thus, the robustness of these protocols is considered to be medium. SD-RAC [2] protocol ensures address uniqueness in both scenarios. Therefore, the protocol is considered to be highly robust.

Scalability: The scalability of an addressing protocol is said to be high if the performance of the protocol does not degrade much in terms of addressing latency and addressing overhead even when the network size is large. The addressing overhead and the addressing latency of the stateless allocation protocols and MANETconf [11], FAP [25], and ODACP [14] protocols under the stateless allocation category are $O(n^2)$ and $O(2td)$, respectively. Therefore, these protocols are considered to be of low scalability. In stateful allocation, if a new node can acquire its address from a neighbor node and the addressing belongs to IPv4 family, then most of the nodes acting as proxies may exhaust their address spaces very quickly in a large network. This increases the addressing overhead and the addressing latency of the protocols. The scalability of these protocols is medium. On the contrary, if a new node can acquire its address from a neighbor node and the addressing belongs to IPv6 family, then the nodes acting as proxies have larger address spaces. Therefore, these protocols are highly scalable.

Security: The addressing protocol must provide security against the attacker and ensure that only the authorized nodes are configured and granted access to the network resources. Under the stateless allocation, only the protocol [9] proposed by Wang et al. considers the security aspects. In stateful allocation, ADIP [22], IDDIP [23], IDSDDIP [24], SD-RAC [2], and the protocol [19] proposed by Cavalli et al. consider the security at the time of address allocation process. Table 8.2 presents the secure address allocation protocols for MANETs.

As discussed earlier, MANET can be an isolated network or can be connected with Internet (and also with IoT) using gateway nodes [34–37]. The above address allocation protocols can allocate addresses to the nodes locally. This network uses these allocated addresses for unicast communication within the network. However, it uses network address translator (NAT) to translate the local address to the global address whenever the network needs to connect with IoT. The gateway node runs the NAT protocol for the connectivity between MANET and IoT.

Table 8.2 Secure Address Allocation Protocols

Protocols	*Attacks*					*Security Mechanisms*
	Address Spoofing	*Address Exhaustion*	*Address Conflict*	*False Deny*	*Negative Reply*	
CAVALI [19]	Protected	Protected	Protected	Protected	Protected	Challenge response
WANG [9]	Protected	Protected	Not protected	Protected	Protected	Self-authentication
ADIP	Protected	Not protected	Protected	Protected	Not protected	Trusted third party
IDDIP	Protected	Not protected	Protected	Protected	Protected	Self-authentication
IDSDDIP	Protected	Not protected	Protected	Protected	Protected	Trusted third party
SDRAC	Protected	Protected	Protected	Protected	Protected	Self-authentication

8.4 Conclusions

A number of address allocation protocols for mobile ad hoc networks have been presented in this chapter. Here, the existing allocation protocols have been categorized into *stateful* and *stateless*. Their performances are shown in terms of addressing overhead, latency, robustness, scalability, and security. It can be seen that most of the address allocation protocols do not consider the security aspect. As a result of this trend, there would be several types of potential attacks during the phase of address allocation to the nodes. These protocols also assume that the gateway node provides the connectivity between the MANET and IoT using NAT. However, there will be always several challenges to connect the MANET with IoT. Efficient mechanisms need to be devised to do the needful using optimal resources and cost given the configuration of future network settings.

References

1. P. Chatterjee, U. Ghosh, I. Sengupta, and S. K. Ghosh, "A trust enhanced secure clustering framework for wireless ad hoc networks," *Wireless Networks*, vol. 20, no. 7, pp. 1669–1684, 2014.
2. U. Ghosh and R. Datta, "A secure addressing scheme for large-scale managed MANETs," *IEEE Transactions on Network and Service Management*, vol. 12, no. 3, pp. 483–495, 2015.
3. R. Droms, "Dynamic host configuration protocol," *RFC 2131*, March 1997.
4. U. Ghosh and R. Datta. "A novel signature scheme to secure distributed dynamic address configuration protocol in mobile ad hoc networks," in *Proceedings 2012 IEEE Wireless Communications and Networking Conference (WCNC)*, (Paris, France), pp. 2700–2705, 2012.
5. N. H. Vaidya, "Weak duplicate address detection in mobile ad hoc networks," in *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc02)*, (Lausanne, Switzerland), pp. 206–216, June 2002.
6. K. Weniger, "Passive duplicate address detection in mobile ad hoc networks," in *Proceedings of IEEE WCNC*, (Florence, Italy), February 2003.
7. C. E. Perkins, J. T. Malinen, R. Wakikawa, E. M. Belding-Royer, and Y. Sun, "Ad hoc address autoconfiguration," *IETF Internet Draft draft-ietf-manet-autoconf-01.txt*, November 2001.
8. M. Fazio, M. Villari, and A. Puliafito, "Aipac: Automatic ip address configuration in mobile ad hoc networks," *Computer Communications*, vol. 29, no. 8, pp. 1189–1200, 2006.
9. P. Wang, D. S. Reeves, and P. Ning, "Secure address auto-configuration for mobile ad hoc networks," in *Proceedings of 2nd Annual International Conference MobiQuitous*, (San Diego, CA), pp. 519–522, 2005.
10. K. Weniger, "Pacman: Passive autoconfiguration for mobile ad hoc networks," *Special issue, IEEE JSAC, Wireless Ad Hoc Networks*, vol. 23, pp. 507–519, March 2005.

11. S. Nesargi and R. Prakash, "Manetconf: Configuration of hosts in a mobile ad hoc network," in *Proceedings of IEEE INFOCOM*, (New York), pp. 1059–1068, 2002.
12. H. Zhou, L. M. Ni, and M. W. Mutka, "Prophet address allocation for large scale MANETs," in *Proceedings of IEEE INFOCOM*, (San Francisco, CA), pp. 1304–1311, 2003.
13. Y. Sun and E. M. Belding-Royer, "Dynamic address configuration in mobile ad hoc networks," *UCSB Technical Report*, pp. 2003–2011, June 2003.
14. Y. Sun and E. M. Belding-Royer, "A study of dynamic addressing techniques in mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 4, pp. 315–329. doi:10.1002/wcm.215, April 2004.
15. M. Taghiloo, M. Dehghan, J. Taghiloo, and M. Fazio, "New approach for address auto-configuration in manet based on virtual address space mapping (vasm)," in *Proceedings of IEEE ICTTA 2008*, (Damascus, Syria), 7–11 April 2008.
16. M. Tajamolian, M. Taghiloo, and M. Tajamolian, "Lightweight secure ip address auto-configuration based on vasm," in *Proceedings of 2009 International Conference on Advanced Information Networking and Applications Workshops*, (Bradford, UK), pp. 176–180, 2009.
17. X. Chu, Y. Sun, K. Xu, Z. Sakander, and J. Liu, "Quadratic residue based address allocation for mobile ad hoc networks," in *IEEE International Conference on Communications (ICC)*, vol. 158, (Beijing, China), pp. 2343–2347, 19–23 May 2008.
18. M. Mohsin and R. Prakash, "Ip address assignment in a mobile ad hoc network," in *Proceedings of IEEE MILCOM*, September 2002.
19. A. Cavalli and J. Orset, "Secure hosts auto-configuration in mobile ad hoc networks," *Elsevier Ad Hoc Networks*, vol. 3, no. 5, pp. 656–667, 2005.
20. Y. Hsu and C. Tseng, "Prime dhcp: A prime numbering address allocation mechanism for manets," *IEEE Communications Letters*, vol. 9, no. 8, August 2005.
21. U. Ghosh and R. Datta, "Mmip: A new dynamic ip configuration scheme with mac address mapping for mobile ad hoc networks," in *Proceedings of Fifteenth National Conference on Communications 2009*, (IIT Guwahati, India), January 2009.
22. U. Ghosh and R. Datta, "Adip: An improved authenticated dynamic ip configuration scheme for mobile ad hoc networks," *International Journal of Ultra Wideband Communications and Systems*, vol. 1, pp. 102–117, 2009.
23. U. Ghosh and R. Datta, "A secure dynamic ip configuration scheme for mobile ad hoc networks," *Elseiver Ad Hoc Networks*, vol. 9, no. 7, pp. 1327–1342, 2011.
24. U. Ghosh and R. Datta. "IDSDDIP: A secure distributed dynamic IP configuration scheme for mobile ad hoc networks," *International Journal of Network Management*, vol. 23, no. 6, pp. 424–446, 2013.
25. N. C. Fernandes, M. D. D. Moreira, and O. C. M. B. Duarte, "An efficient and robust addressing protocol for node autoconfiguration in ad hoc networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 99, p. 1, 2013.
26. C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (aodv) routing," *draft-ietf-manet-aodv-11.txt*, June 2002 (work in progress).
27. M. Taghiloo, J. Taghiloo, and M. Dehghan, "A survey of secure address autoconfiguration in MANET," in *Proceedings of 10th IEEE International Conference on Communication Systems (ICCS)*, pp. 1–5, October 2006.

28. W. Xiaonan and Q. Huanyan, "Cluster-based and distributed ipv6 address configuration scheme for a MANET," *Wireless Personal Communications*, vol. 71, no. 4, pp. 3131–3156, doi:10.1007/s11277-013-0995-1, August 2013.
29. T. R. Reshmi and K. Murugan, "Secure and Reliable Autoconfiguration Protocol (SRACP) for MANETs," *Wireless Personal Communication*, vol. 89, no. 4, pp. 1243–1264, doi:10.1007/s11277-016-3314-9, August 2016.
30. N. Wangi, R. Prasad, M. Jacobsson, and I. Niemegeers, "Address autoconfiguration in wireless ad hoc networks: protocols and techniques," *IEEE Wireless Communications*, vol. 15, pp. 70–80, February 2008.
31. S. Ni, Y. Tseng, Y. Chen, and J. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the ACM/IEEE MOBICOM*, (Seattle, WA), pp. 151–162, 1999.
32. C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance vector routing (dsdv) for mobile computers," *SIGCOMM Compututer Communication Review*, vol. 24, no. 4, pp. 234–244, 1994.
33. S. Kim and J. Chung, "Message complexity analysis of mobile ad hoc network address autoconfiguration protocols," *IEEE Transactions on Mobile Computing*, vol. 7, no. 3, pp. 358–371, March 2008.
34. R. Kushwah, S. Tapaswi, and A. Kumar, "A detailed study on internet connectivity schemes for mobile ad hoc network," *Wireless Personal Communication*, vol. 104, no. 4, pp. 1433–1471, doi:10.1007/s11277-018-6093-7, February 2019.
35. C. E. Perkins, J. T. Malinen, R. Wakikawa, A. Nilsson, and A. J. Tuominen, "Internet connectivity for mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, 465–482, 2002.
36. P. Bellavista, G. Cardone, A. Corradi, and L. Foschini, "Convergence of manet and wsn in iot urban scenarios," *IEEE Sensors Journal*, vol. 13, pp. 3558–3567, October 2013.
37. B. KameswaraRao and A. S. N. Chakravarthy, "A taxonomical review on manet networks for iot based air pollution controls," *International Journal of Computer Science and Technology*, vol. 8, no. 3, pp. 44–49, July–September 2017.