Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 57 (2015) 907 – 914

Procedia Computer Science

3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)

IPv6 address auto-configuration protocol for mobile Ad Hoc Networks

K. Sahadevaiah[1*], N. Ramakrishnaiah[2], Prasad Reddy P.V.G.D.[3]

[1, 2]*Department of Computer Science & Engineering, University College of Engineering, Jawaharlal Nehru Technological University, KAKINADA-533003, AP (India)*

[3]Department of Computer Science and Systems Engineering, AU College of Engineering, *Andhra University, VISAKHAPATNAM-5300003, AP (India)*

Abstract

The major task of an address autoconfiguration protocol in Mobile Ad hoc Networks (MANETs) is to manage the resource address space efficiently and effectively. An unconfigured node should be able to allocate a unique network address in a timely manner. Once a node leaves the network, its address should be reclaimed for future usage. The lack of a central server and hosts' mobility makes address allocation a challenging task in ad hoc networks. In recent years, various address autoconfiguration protocols have been proposed in the literature. In this paper, we present an IPv6 address configuration scheme that can allocate IPv6 addresses to the authorized hosts for a MANET. Each host can generate unique IPv6 address from its own IPv6 address and can assign those addresses to the new nodes. The scheme has been implemented in network simulator 3.13 with Ubuntu. Simulation outcome has shown that the scheme is robust and IP addresses can be allocated to nodes with an acceptable addressing latency and communication overhead.

Keywords: Mobile Ad Hoc Network, IP Address Assignment, IPv6, Dynamic Address, Autoconfiguration, Duplicate address detection

1. Introduction

Mobile Ad hoc network (MANET) is an independent self-configuring wireless communication infrastructure-less network of mobile nodes connected by wireless links. The major challenges in MANETs are: automatic IP address configuration, scalable routing and security. The autoconfiguration protocols are classified into three categories: stateful approaches, stateless approaches and hybrid approaches [1]. A *stateful approach* assumes the existence of a central entity to keep an address allocation table for whole network and assigns unique IP address for unconfigured node. i.e. the nodes know the network state. *Stateless approaches* do not need a central entity to maintain an address allocation table. Instead, each node selects an address by itself randomly and verifies its uniqueness with the so-called *duplicate address detection* (DAD) procedure. If duplication is detected, at least one of the nodes with duplicate addresses must change its address. *Hybrid approaches* combine both stateful and

* Corresponding author. Tel.: +91 950 266 1429
E-mail address: ksd1868@gmail.com

Peer-review under responsibility of organizing committee of the 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)
doi:10.1016/j.procs.2015.07.507

stateless mechanisms to improve the scalability and reliability of the autoconfiguration, but may result in higher complexity and higher protocol overhead.

A node cannot take part in a communication, unless it has its own IP address. For an automatic IP address assignment in wired networks, nodes may use a dynamic host configuration protocol (DHCP) to acquire an IP address from a centralized server. However, this solution cannot be used in MANETs due to the mobility of nodes and difficulty of maintaining a centralized DHCP server. Many dynamic address configuration protocols have been proposed for MANET. However, most of the protocols rely on passive duplicate address detection (DAD) mechanism to resolve the address conflicts lack a mechanism for authentication.

In this paper, we present an IPv6 address configuration scheme that can allocate IPv6 addresses to the authorized hosts for a MANET. The proposed scheme does not require broadcasting entire MANET for duplicate address detection (DAD). Each host in the network can generates a node ID and a unique IPv6 address for a new authorized host. Thus, a node can be identified by the unique tuple (*node_id, IP address*) [2], which adds a node authentication feature in addition to identification.

The rest of the paper is organized as follows. Section 2 provides system model. Section 3 explains the proposed scheme. Section 4 describes the performance and simulation results. Finally, section 5 concludes the paper.

2. System Model

We consider the formation of an autonomous ad hoc network starting from one node and, then, the other nodes join the network one by one. The autonomous ad hoc network that is considered has no gateway or no external connection to any other network. The nodes are free to move around and can leave or join network at any time. Nodes are identified by unique tuple (*node_id, IPv6 address*). The IPv6 address configuration protocols can configure a node with a unique address. A new node first requests an address from a neighbor node. If the neighbor nodes do not have the address space, then the new node requests an address from a remote node. In the latter case, both the cost and the delay are increased. Every node, except root node, contains a counter variable *count* which specifies number of IP addresses can be allocated by a node. Value of counter variable can be in the range of 10-100.

The Figure 1 depicts how an address is allocated for first 10 nodes. Starting from the root node with an IP fe80:0000:0000:0000:0000:0000:0000:0001. Here, network prefix fe80:0000:0000:0000 is omitted for reaming nodes, as network prefix remains unchanged throughout address allocating process. It may be noted that, allocating of IP address may not follow the same order as shown in below tree. It depends upon neighbours nodes. The below tree depicts the order of address allocation for 10 nodes in our simulation. Node numbers gives the order in which nodes entered into simulation. Hence, node 1 entered first and node 2 arrived after node 1 and so on.

3. Proposed Scheme

The proposed scheme dynamically allocates the IP addresses to the hosts in the network. When a new host wants to join an ad hoc network, it periodically issues a *DETECT* broadcast message along with its signature to its neighbours till it either receives a *PROVIDE* message or a *REFUSE* message. If no *PROVIDE* or *REFUSE* message is received, it means that there are no neighbour nodes, the new host configures itself as a root node to the IP address, say, fe80:0000:0000:0000:0000:0000:0000:0001 and generates a unique network ID (NID) as a network identifier and also node identifier (node_id).

If new host receives the signed *PROVIDE* (provideIP) message from one or more neighbours, it chooses the smallest IP address that is offered to it. After choosing smallest IP address, a signed *CHOOSE* message is sent back to the proxy offering that IP address indicating that new host has chosen the IP provided by proxy. Proxy

node acknowledges new node by sending *ACK* message back to it. The other *PROVIDE* (provideIP) messages sent by neighbour proxies are ignored by new node. On receiving the signed *CHOOSE* message, signed *ACK* message is sent to the new host. After receiving the signed *ACK* message from the chosen proxy, new host performs a final check on the configuration parameters specified in the *ACK* message and configures itself. If some packets are lost due to host mobility or channel error during the address allocation process, the situation is handled by maintaining timers. The address allocation process is explained through the steps: a, b, c, d, e, f, g and h.

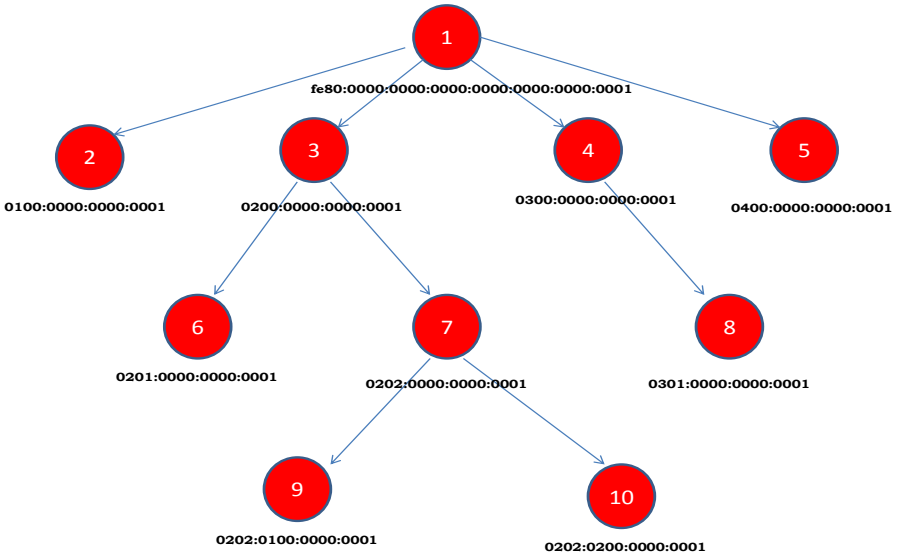
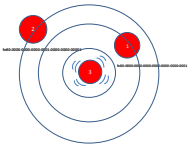


Figure 1 Allocation of IPv6 addresses for 10 nodes

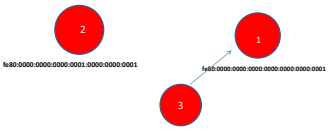
(a) New host sending *DETECT* broadcast message

(b) New host configured itself as a root with an IP address



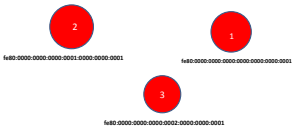
(c) Newly arrived host 3 issuing *DETECT* broadcasting message to its neighbours broadcast message

(d) Neighbour hosts sending unica *PROVIDE* message to host 3 after receivin *DETECT* message



(f) Host 3 sending *CHOOSE* message to host 1thus selecting IP offered by host 1

(g) Host 1 sending *ACK* message to host 3



(h) After receiving *ACK* message from host 1, host 3 configures to the IP offered by host 1

IPv6 Address Allocation:

IPv6 has 128 bits addressing in which most significant 64 bits represent network prefix and least significant 64 bits represent host address or interface id. For example, assume that IPv6 is represented as a. b. c. d. e. f. g. h. m. n. o. p. q. r. s. t. Here a. b. c. d. e. f. g. h represents network prefix and m. n. o. p. q. r. s. t represents host address or interface id. Both network prefix and host address are represented in decimal format for better understanding. A host in the network can have address from 0.0.0.0.0.0.0.1 to 255.255.255.255.255.255.255.254.

At first, root node configures itself to, say, a. b. c. e. d. f. g. h. 0. 0. 0. 0. 0. 0. 0. 1 and it can allocate address from a. b. c. d. e. f. g. h. 1. 0. 0. 0. 0. 0. 0. 1 to a. b. c. d. e. f. g. h. 255. 0. 0. 0. 1 and from a. b. c. e. d. f. g. 0. 0. 0. 0. 0. 0. 0. 2 to a. b. c. e. d. f. g. h. 0. 0. 0. 0. 0. 0. 0. 255. A host having an address a. b. c. e. d. f. g. h. 1. 0. 0. 0. 0. 0. 0. 1 can assign addresses from a. b. c. d. e. f. g. h. 1. 1. 0. 0. 0. 0. 0. 1 to a. b. c. d. e. f. g. h. 1. 255. 0. 0. 0. 0. 0. 0. 1. A host having an address a. b. c. e. d. f. g. h. 1. 0. 0. 0. 0. 0. 0. 2 can assign addresses from a. b. c. d. e. f. g. h. 1. 1. 0. 0. 0. 0. 0. 0. 2 to a. b. c. d. e. f. g. h. 1. 255. 0. 0. 0. 0. 0. 2. A host having an address a. b. c. e. d. f. g. h. 255.255.255. 255.255.255.0.255 can assign addresses from 255.255.255.255.255.255.1.255 to 255.255.255.255.255.255.255.254 with network prefix a. b. c. d. e. f. g. h.

4. Experimental Results and Analysis

The proposed scheme has been implemented in *network simulator* 3.13 with Ubuntu. The radio transmission range of each node is set to 250m. A node joins the simulation for every 4 seconds. SHA-1 is used as hash function to generate tag. RSA digital signature mechanism of key size 1024 is used as public key digital signature mechanism. The performance of the proposed address allocation scheme is evaluated with the metrics: addressing latency, network deployment time and communication overhead. The simulation snapshots for hosts 50 and 100 are shown

in Figure 2 and Figure 3.

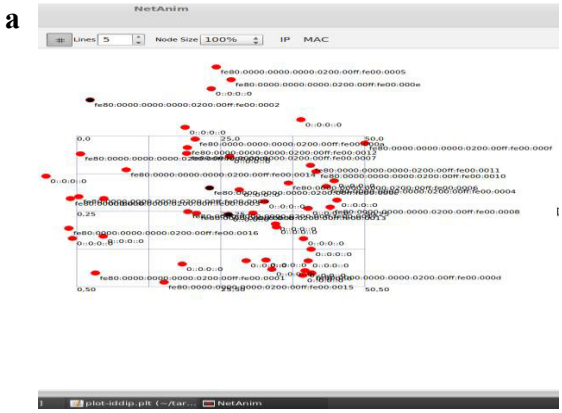
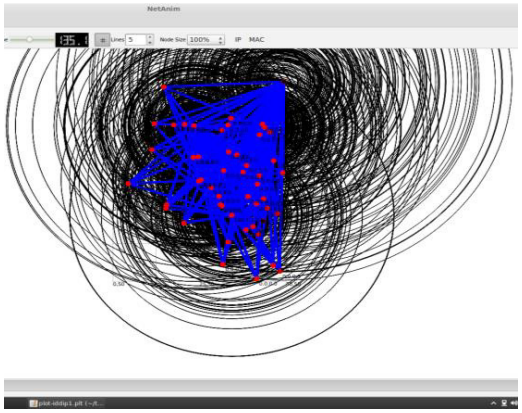


Figure 2 (a) Simulation for 50 nodes



(b) Nodes communication in the simulation

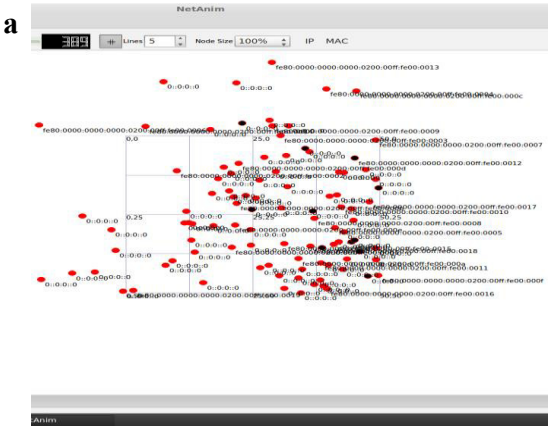
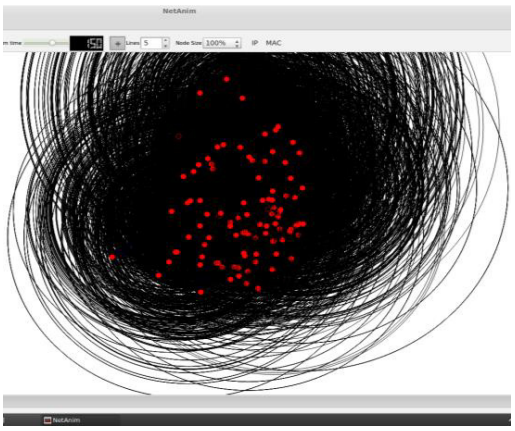


Figure 3 (a) Simulation for 100 nodes



(b) Nodes communication in the simulation

Addressing Latency: Addressing latency is the time taken to acquire an IP address by a node after its arrival. The Table 1 shows the average addressing latency per node. The resulting data of Table 1 is plotted using MATLAB version 6.5 and is shown in Figure 4. From the Figure 4, it is clear that addressing latency is increasing with increase in network size.

Communication Overhead: The Table 2 shows the average numbers of packets exchanged per node. The resulting data of Table 2 is plotted using MATLAB version 6.5 and is shown in Figure 5. The communication overhead increases over increase in the number of nodes in MANET.

Network Deployment: Network deployment time gives the total time taken for all the nodes to acquire IP address and configure themselves. The Table 3 shows the network deployment time. The resulting data of Table 3 is plotted using MATLAB version 6.5 and is shown in Figure 6.

Table 1. Average addressing latency per node

Number of Nodes	Average Latency Per Node(Sec)
10	*0.3*
20	0.5
30	0.7
40	1.1
50	1.3
60	1.5
70	1.7
80	2.1
90	2.3
100	2.5

Table 2. Average number of packets exchanged per node

Number of Nodes	Average Number of Packets Exchanged
10	*8*
20	10
30	12
40	14
50	16
60	18
70	20
80	22
90	24
100	26

Table 3 Network deployment time

Number of Nodes	Deployment Time (Sec)
10	*43*
20	90
30	141
40	204
50	265
60	330
70	399
80	488
90	567
100	650

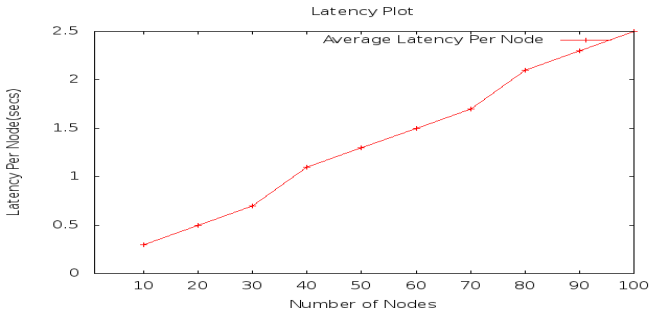


Figure 4 Average node addressing latency

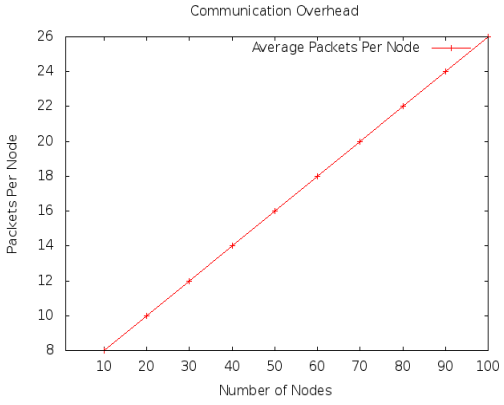


Figure 5 Average number of packets exchanged per node

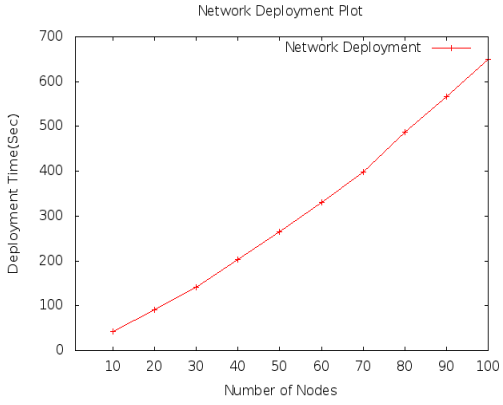


Figure 6 Network deployment time

5. Conclusion

In this paper, we present an IPv6 address configuration scheme that can allocate IPv6 addresses to the authorized hosts for a MANET. The scheme ensures that only authorized host will be configured. The request messages for the address assignment need not be flooded all over the network, thereby saving considerable bandwidth. Each host can generate and assign a unique IP6 address for a new host. The proposed scheme has been implemented in network simulator 3.13 with Ubuntu. The simulation results shows that the proposed scheme is robust and IP addresses can be allocated to nodes with an acceptable addressing latency and communication overhead.

References

[1] Kilian Weniger and Martina Zitterbart, "*Address Autoconfiguration in Mobile Ad Hoc Networks: Current Approaches and Future Directions"*, IEEE Network, Page No. 6-11, July/August 2004.

[2] P. Wang, D. S. Reeves and P. Ning, "*Secure Address Auto-configuration for Mobile Ad Hoc Networks*", Proceedings of 2nd Annual International Conference, MobiQuitous, 2005.

[3] Uttam Ghosh and Raja Datta, "*A Secure Dynamic IP Configuration Scheme for Mobile Ad Hoc Networks"*, Elsevier, Ad Hoc Networks, Vpl. 9, pp. 1327–1342, 2011.

[4] Uttam Ghosh and Raja Datta, "MMIP: A New Dynamic IP Configuration Scheme with MAC Address Mapping for Mobile Ad hoc networks", National Conference on Communications, IIT Guwahati, January 2009.

[5] M Fazio, M. Villari, and A. Puliafito, "AIPAC: Automatic IP Address Configuration in Mobile Ad Hoc Networks," Performance Evaluation of Wireless Networks and Communications, vol. Computer Communications 29, Issue 8, pp. 1189–1200, 15 May 2006.

[6] H.Zhou, L. M. Ni, and M. W. Mutka, "Prophet address allocation for large scale MANETs ", INFOCOM, pp. 1304–1311, 2003.

[7] M.Tajamolian, M.Taghiloo, and M.Tajamolian, "Lightweight secure ip address auto-configuration based on vasm," 2009 International Conference on Advanced Information Networking and Applications Workshops, pp. 176–180, Waina 2009.

[8] A. Cavalli and J. Orset, "Secure hosts auto-configuration in mobile ad hoc networks, " Data Communication and Topology Control in Ad Hoc Networks, vol. Ad Hoc Networks 3, Issue 5, pp. 656–667, 2005.

[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, pp. 519–522, 2005.