© 2015 IJSRSET | Volume 1 | Issue 6 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

An Economical and Strong Addressing Protocol for Node Machine Configuration in Unintended Networks

[1]Dr. CH. G. V. N. Prasad, [2]B. Sanjay Kumar, [3]P. Jayanthi Yadav

[1]Professor & HOD, Department of Computer Science & Engineering in Sri Indu College of Engineering & Technology, Hyderabad, India
[2]pursuing M.Tech (CSE), Sri Indu College of Engineering& Technology, Affiliated to JNTU-Hyderabad, India
[3]pursuing M.Tech (CSE), Sri Indu College of Engineering& Technology, Affiliated to JNTU-Hyderabad, India

ABSTRACT

Address assignment could be a key challenge in unintended networks because of the dearth of infrastructure. Autonomous addressing protocols need a distributed and self-managed mechanism to avoid address collisions during a dynamic network with weakening channels, frequent partitions, and joining/leaving nodes. we have a tendency to propose and analyze a light-weight protocol that configures mobile unintended nodes supported a distributed address information hold on in filters that reduces the management load and makes the proposal strong to packet losses and network partitions. We have a tendency to evaluate the performance of our protocol, considering connection nodes, partition merging events, and network low-level formatting. Simulation results show that our protocol resolves all the address collisions and conjointly reduces the management traffic compared to antecedently planned protocols.

Keywords: Ad Hoc Networks, Electronic Network Management

I. INTRODUCTION

A. Networking

Networking is the word basically relating to computers and their connectivity. It is very often used in the world of computers and their use in different connections. The term networking implies the link between two or more computers and their devices, with the vital purpose of sharing the data stored in the computers, with each other. The networks between the computing devices are very common these days due to the launch of various hardware and computer software which aid in making the activity much more convenient to build and use.

The Above Figure (1.1) Shows That Structure of Networking between the different computers, the networks between the computing devices are very common these days due to the launch of various hardware and computer software which aid in making the activity much more convenient to build and use.

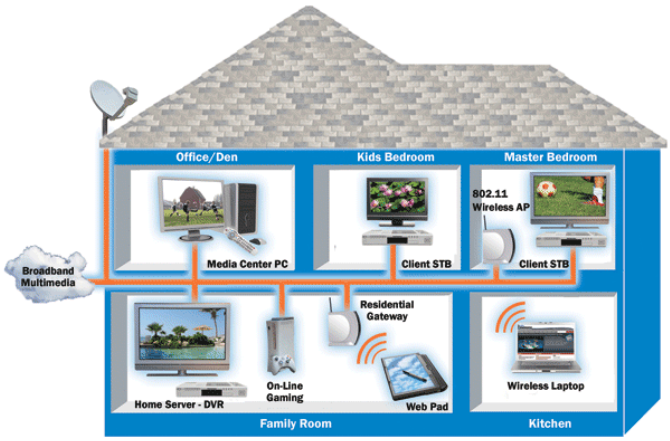

Figure 1: Structure of Networking between the different computers

B. Network Working

General Network Techniques :- When computers communicate on a network, they send out data packets without knowing if anyone is listening. Computers in a network all have a connection to the network and that is called to be connected to a network bus. What one

computer sends out will reach all the other computers on the local network.

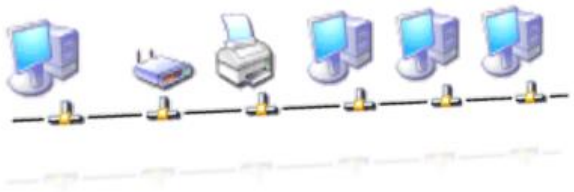

Figure 1.2

The Figure **(1.2)** shows that the Connecting Procedure For Networking of computers communicate on a network, they send out data packets without knowing if anyone is listening. Computers in a network all have a connection to the network and that is called to be connected to a network bus.

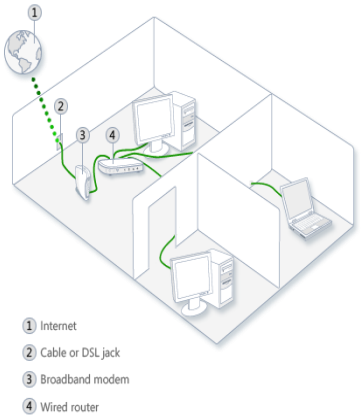

Figure 1.3

Above Figure (1.3) show the clear idea about the networking functions for the different computers to be able to distinguish between each other, every computer has a unique ID called MAC-address (Media Access Control Address). This address is not only unique on your network but unique for all devices that can be hooked up to a network. The MAC-address is tied to the hardware and has nothing to do with IP-addresses. Since all computers on the network receives everything that is sent out from all other computers the MAC-addresses is primarily used by the computers to filter out incoming network traffic that is addressed to the individual computer.

When a computer communicates with another computer on the network, it sends out both the other computers MAC-address and the MAC-address of its own. In that way the receiving computer will not only recognize that this packet is for me but also, who sent this data packet so a return response can be sent to the sender.

C. On an Ethernet Network

As described here, all computers hear all network traffic since they are connected to the same bus. This network structure is called multi-drop.

One problem with this network structure is that when you have, let say ten (10) computers on a network and they communicate frequently and due to that they sends out there data packets randomly, collisions occur when two or more computers sends data at the same time. When that happens data gets corrupted and has to be resent. On a network that is heavy loaded even the resent packets collide with other packets and have to be resent again. In reality this soon becomes a bandwidth problem. If several computers communicate with each other at high speed they may not be able to utilize more than 25% of the total network bandwidth since the rest of the bandwidth is used for resending previously corrupted packets. The way to minimize this problem is to use network switches.

D. Characteristics of Networking

The following characteristics should be considered in network design and ongoing maintenance:

Availability is typically measured in a percentage based on the number of minutes that exist in a year. Therefore, uptime would be the number of minutes the network is available divided by the number of minutes in a year.

Cost includes the cost of the network components, their installation, and their ongoing maintenance.

Reliability defines the reliability of the network components and the connectivity between them. Mean time between failures (MTBF) is commonly used to measure reliability.

Security includes the protection of the network components and the data they contain and/or the data transmitted between them.

Speed includes how fast data is transmitted between network end points (the data rate).

Scalability defines how well the network can adapt to new growth, including new users, applications, and network components.

Topology describes the physical cabling layout and the logical way data moves between components.

E. Types of Networks

Organizations of different structures, sizes, and budgets need different types of networks. Networks can be divided into one of two categories

- **Peer-to-peer**
- **Server-based networks**

F. Peer-to-Peer Network

A peer-to-peer network has no dedicated servers; instead, a number of workstations are connected together for the purpose of sharing information or devices. Peer-to-peer networks are designed to satisfy the networking needs of home networks or of small companies that do not want to spend a lot of money on a dedicated server but still want to have the capability to share information or devices like in school, college, cyber café.

G. Server-Based Networks

In server-based network data files that will be used by all of the users are stored on the one server. With a server-based network, the network server stores a list of users who may use network resources and usually holds the resources as well.

This will help by giving you a central point to set up permissions on the data files, and it will give you a central point from which to back up all of the data in case data loss should occur.

H. Network Communications

Computer networks use signals to transmit data, and protocols are the languages computers use to communicate. Protocols provide a variety of communications services to the computers on the network. Local area networks connect computers using a shared, half-duplex, baseband medium, and wide area networks link distant networks. Enterprise networks often consist of clients and servers on horizontal segments connected by a common backbone, while peer-to-peer networks consist of a small number of computers on a single LAN.

I. Advantages of Networking

- **Easy Communication:**

It is very easy to communicate through a network. People can communicate efficiently using a network with a group of people. They can enjoy the benefit of emails, instant messaging, telephony, video conferencing, chat rooms, etc.

- **Ability to Share Files, Data and Information:**

This is one of the major advantages of networking computers. People can find and share information and data because of networking. This is beneficial for large organizations to maintain their data in an organized manner and facilitate access for desired people.

- **Sharing Hardware:**

Another important advantage of networking is the ability to share hardware. For an example, a printer can be shared among the users in a network so that there's no need to have individual printers for each and every computer in the company. This will significantly reduce the cost of purchasing hardware.

- **Sharing Software:**

Users can share software within the network easily. Networkable versions of software are available at considerable savings compared to individually licensed version of the same software. Therefore large companies can reduce the cost of buying software by networking their computers.

- **Security:**

Sensitive files and programs on a network can be password protected. Then those files can only be accessed by the authorized users. This is another important advantage of networking when there are concerns about security issues. Also each and every user has their own set of privileges to prevent those accessing restricted files and programs.

- **Speed:**

Sharing and transferring files within networks is very rapid, depending on the type of network. This will save time while maintaining the integrity of files.

II. METHODS AND MATERIAL

A. Existing System

As other wireless networks, ad hoc nodes also need a unique network address to enable multihop routing and full connectivity. Address assignment in ad hoc networks, however, is even more challenging due to the self-organized nature of these environments. Centralized mechanisms, such as the Dynamic Host Configuration Protocol (DHCP) or the Network Address Translation (NAT), conflict with the distributed nature of ad hoc networks and do not address network partitioning and merging.

DISADVANTAGES OF EXISTING SYSTEM:

- A crucial and usually unaddressed issue of ad hoc networks is the frequent network partitions.
- Network partitions, caused by node mobility, fading channels and nodes joining and leaving the network, can disrupt the distributed network control.
- Network initialization is another challenging issue because of the lack of servers in the network.

B. Proposed System

In this paper, we propose and analyze an efficient approach called Filter-based Addressing Protocol (FAP). The proposed protocol maintains a distributed database stored in filters containing the currently allocated addresses in a compact fashion. We consider both the Bloom filter and a proposed filter, called Sequence filter, to design a filter-based protocol that assures both the univocal address configuration of the nodes joining the network and the detection of address collisions after merging partitions. Our filter-based approach simplifies the univocal address allocation and the detection of address collisions because every node can easily check whether an address is already assigned or not.

We also propose to use the hash of this filter as a partition identifier, providing an important feature for an easy detection of network partitions. Hence, we introduce the filters to store the allocated addresses without incurring in high storage overhead. The filters are distributed maintained by exchanging the hash of the filters among neighbors. This allows nodes to detect with a small control overhead neighbors using different filters, which could cause address collision. Hence, our proposal is a robust addressing scheme because it guarantees that all nodes share the same allocated list.

BENIFITS OF PROPOSED SYSTEM:

- ✓ FAP achieves low communication overhead and low latency, resolving all address collisions even in network partition merging events.
- ✓ The use of filters methods because they reduce the number of tries to allocate an address to a joining node, as well as they reduce the number of false positives in the partition merging events, when compared to other proposals, which reduces message overhead.

C. Methodology

i. Network Initialization

The network initialization procedure deals with the auto configuration of the initial set of nodes. Two different scenarios can happen at the initialization: the joining nodes arrive one after the other with a long enough interval between them, called gradual initialization, or all the nodes arrive at the same time, called abrupt initialization. Most protocols assume the gradual scenario with a large time interval between the arrival of the first and the second joining nodes. For example, the protocol proposed by Fan and Subramani assumes that the first node is alone to choose a partition identifier. Then, the following joining nodes are handled by the first node through the joining node procedure. If all nodes join the network approximately at the same time, each node will choose a different partition identifier. This triggers many partition merging procedures simultaneously, which creates a high control load and can cause inconsistencies in the address allocation procedure, generating address collisions. We argue that address allocation protocols must operate without any restriction to the way the nodes join the network. Our filter-based proposal fits well for both gradual and abrupt initialization scenarios, using Hello and AREQ messages. The Hello message is used by a node to advertise its current association status and partition identifier. The AREQ message is used to advertise that a previously available address is now allocated. Each AREQ has an identifier number, which is used to differentiate AREQ messages generated by different nodes, but with the same address.

ii. Node Ingress

After the initialization, each node starts broadcasting periodic Hello messages containing its address filter signature. Upon the reception of a Hello, neighbours evaluate whether the signature in the message is the same as its own signature to detect merging events. Only the nodes that have already joined the network are able to send Hello messages, receive a request of a node to join the network, and detect merging events. In this

Node Ingress module, a node turns on, it listens to the medium for a period .If the node listens to a Hello, there is at least one node with an address filter, and the network already exists. Hence, the node knows that it is a joining node instead of an initiator node. The joining node then asks for the source of the first listened Hello message (the host node) to send the address filter of the network using an Address Filter (AF) message. When the host node receives the AF, it checks bit I, which indicates whether the message is being used for a node-joining procedure or a partition-merging procedure. If I=1, the message came from a joining node. Then, the host node answers the request with another AF with bit R set to 1, indicating that the AF is an answer to a previous filter request. When the joining node receives the AF reply message, it stores the address filter, chooses a random available address, and floods the network with an AREQ to allocate the new address. When the other nodes receive the AREQ, they insert the new address in their filters and update their filter signatures with the hash of the updated filter.

iii. Network Merging Event

Merging events are also detected based on Hello and AF messages. Nodes in different partitions choose their address based only on the set of addresses of their partition. Hence, nodes indifferent partitions can select the same address, which may cause collisions after the partitions merged. In FAP, when a node receives a Hello, it checks whether the filter signature on the message is different than its current signature. If so, the node knows that they have different sets of allocated addresses. If there is more Tp than since the last merging event, a new merging procedure is started. In this procedure, both nodes exchange AF messages to disseminate the filters of the two partitions. First, each node checks whether its address is equal or greater than the address of the other node. The node with the greatest address, or both nodes, in case addresses are equal, starts the process. The node starting the process sends an AF message with its current address filter to the other node, which stores the received filter and sends back an AF message with the filter of its partition.

Then, both nodes flood their partitions with a Partition message, so that all nodes update their filters with the other partition data. Upon reception of the Partition message, each node must check the bit M on the Partition message to verify if it is on the lowest priority partition. The lowest priority partition is selected as the smallest partition or, if both partitions are of the same size, it is selected as the partition of the node that started the process. Each node on the lowest-priority partition must check whether its address is on the other partition filter to detect collisions. If there is a collision, the node randomly chooses an available address in both filters and floods the network with an AREQ to allocate the new address. If the node receives an AREQ with the same address that it has chosen, but with a different sequence number, it chooses another address because another node has also chosen the same address. Finally, all the nodes merge the other partition filter with its own filter, insert the addresses received in the AREQs into the new filter, and update the filter signature

iv. Node Departure

When a node leaves the network, its address should become available for the other nodes. If the departing node is correctly shut down, it floods the network with a notification to remove its address from the address filter. If the departing node does not notify the network, the address remains allocated in the filters, which can make the available addresses scarce with time. This can be identified in the address filter by the fraction of bits set to 1 in the Bloom and in the Sequence filter and by the fraction of counters greater than one in the Counter Bloom Filter. Therefore, every node verifies this fraction in their address filters every time the filter is updated.

v. Filtering Techniques

The best filter for FAP depends on network characteristics such as the estimated number of nodes in the network and the number of available addresses. It also depends on the false-positive and false-negative rates of the filter. Bloom filters do not present false negatives, which mean that a membership test of an element that was inserted into the filter is always positive. These filters, however, present a false-positive probability. Hence, a membership test of an element that was not inserted into the Bloom filter may be positive.

The size of the Bloom filter is not determined by the address range, but by the maximum number of elements to be inserted into the filter, which is the upper-bound estimate of the number of active nodes in the network.

On the other hand, the Sequence filter is deterministic and, as a consequence, neither false positives nor false negatives are created. The size of the Sequence filter depends only on the size of the address range, and on the address size. The address range size is defined by the number of bits in address suffix. The Bloom filter size is constant to the address range size and grows with the number of elements whereas the Sequence filter follows the opposite: The filter size is constant to the number of elements and increases with the address range size. As a result, the Bloom filter is more suitable for an extensive address range, whereas the Sequence filter is more adequate to a large number of elements.

III. CONLCUSION

We proposed a distributed and self-managed addressing protocol, called Filter-based Addressing protocol, which fits well for dynamic ad hoc networks with fading channels, frequent partitions, and joining/leaving nodes. Our key idea is to use address filters to avoid address collisions, reduce the control load, and decrease the address allocation delay. We also proposed to use the hash of the filter as the partition identifier, providing an easy and accurate feature for partition detection with a small number of control messages. Moreover, our filter-based protocol increases the protocol robustness to message losses, which is an important issue for ad hoc networks with fading channels and high bit error rates. The use of the hash of the filter instead of a random number as the partition identifier creates a better representation of the set of nodes. Hence, a change in the set of nodes is automatically reflected in the partition identifier. This identifier is periodically advertised, allowing neighbors to recognize if they belong to different sets of nodes. In the other proposals, a mechanism to change the arbitrated partition identifier is requested, which increases the complexity and the packet overhead of the protocol. The proposed protocol efficiently resolves all address collisions even during merging events, as showed by simulations. This is achieved because FAP is able to detect all merging events and also because FAP is robust to message losses. FAP initialization procedure is simple and efficient, requiring a control load similar to the control load of DAD, which is a protocol with a small overhead but that does not handle network partitions. Moreover, FAP presents smaller delays in the joining node procedure and on network partition merging events than the other proposals, indicating that the proposed protocol is more suitable for very dynamic environments with frequent partition merging and node joining events.

IV. REFERENCES

[1] N. C. Fernandes,M.D.Moreira, and O. C. M. B. Duarte, "A self-organized mechanism for thwarting malicious access in ad hoc networks," in Proc. 29th IEEE INFOCOM Miniconf., San Diego, CA, Apr. 2010, pp. 1–5.

[2] N. C. Fernandes,M.D.Moreira, and O. C. M. B. Duarte, "An efficient filter-based addressing protocol for auto configuration of mobile ad hoc networks," in Proc. 28th IEEE INFOCOM, Rio de Janeiro, Brazil, Apr. 2009, pp. 2464–2472.

[3] C. E. Perkins, E. M. Royers, and S. R. Das, "IP address auto configuration for ad hoc networks," Internet draft, 2000.

[4] Z. Fan and S. Subramanian, "An address auto configuration protocol for IPv6 hosts in a mobile ad hoc network," Comput. Commun., vol. 28, no. 4, pp. 339–350, Mar. 2005.

[5] S. Nesargi and R. Prakash, "MANETconf: Configuration of hosts in a mobile ad hoc network," in Proc. 21st Annu. IEEE INFOCOM, Jun. 2002, vol. 2, pp. 1059–1068.

[6] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in Proc. IEEE Symp. Security Privacy, May 2005, pp. 49–63.

[7] S. Thomson and T. Narten, "IPv6 stateless address autoconfiguration," RFC 2462, 1998.

[8] M. Fazio, M. Villari, and A. Puliafito, "IP address auto configuration in ad hoc networks: Design, implementation and measurements," Comput. Netw, vol. 50, no. 7, pp. 898–920, 2006.

[9] N. H. Vaidya, "Weak duplicate address detection in mobile ad hoc networks," in Proc. 3rd ACM MobiHoc, 2002, pp. 206–216.