AN EFFICIENT AND ROBUST ADDRESSING PROTOCOL FOR NODE AUTO CONFIGURATION IN AD HOC NETWORKS

Abstract:

Address assignment is a key challenge in ad hoc networks due to the lack of infrastructure. Autonomous addressing protocols require a distributed and self-managed mechanism to avoid address collisions in a dynamic network with fading channels, frequent partitions, and joining/leaving nodes. We propose and analyze a lightweight protocol that configures mobile ad hoc nodes based on a distributed address database stored in filters that reduces the control load and makes the proposal robust to packet losses and network partitions. We evaluate the performance of our protocol, considering joining nodes, partition merging events, and network initialization. Simulation results show that our protocol resolves all the address collisions and also reduces the control traffic when compared to previously proposed protocols.

Existing system:

As other wireless networks, ad hoc nodes also need a unique network address to enable multi hop 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:

The lack of servers hinders the use of centralized addressing schemes in ad hoc networks. Does not take into account network partitions and is not suitable for ad hoc networks.

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.

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.

Our proposal aims to reduce the control load and to improve partition merging detections without requiring high storage capacity. These objectives are achieved through small filters and an accurate distributed mechanism to update the states in nodes. Furthermore, we propose the use of the filter signature (i.e., a hash of the filter) as a partition identifier instead of random numbers. The filter signature represents the set of all the nodes within the partition. Therefore, if the set of assigned addresses changes, the filter signature also changes. Actually, when using random numbers to identify the partition instead of hash of the filter, the identifier does not change with the set of assigned addresses. Therefore, filter signatures improves the ability to correctly detect and merge partitions.
Advantages of proposed system:

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.

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 collisions. Hence, our proposal is a robust addressing scheme because it guarantees that all nodes share the same allocated list.

Analysis and simulation experiments show that FAP achieves low communication overhead and low latency, resolving all address collisions even in network partition merging events. These results are mainly correlated to the use of filters 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.

Challenges:

A crucial and usually unaddressed issue of ad hoc networks is the frequent network partitions.

As other wireless networks, ad hoc nodes also need a unique network address to enable multi hop routing and full connectivity.

Address assignment in ad hoc networks, however, is even more challenging due to the self-organized nature of these environments.

System requirements:

Hardware requirements:

- Processor - Pentium –III
- Speed - 1.1 Ghz
✓ RAM - 256 MB (min)
✓ Hard Disk - 20 GB
✓ Key Board - Standard Windows Keyboard
✓ Mouse - Two or Three Button Mouse
✓ Monitor - SVGA

Software requirements:

- Operating System: LINUX
- Tool: Network Simulator-2
- Front End: O. TCL (Object Oriented Tool Command Language)

Reference: