MIT2000-06
CarNet: A Scalable Wireless Network Infrastructure

Robert Morris and M. Frans Kaashoek

1. Problem and Goals

The Internet has evolved in a way that sacrifices dynamism in favor of scale: it groups nodes into an addressing and routing hierarchy that inhibits movement, and it depends on fixed physical infrastructure that inhibits rapid deployment. We propose to design and deploy a new network architecture that achieves both dynamism and scalability. Such an architecture would enable new kinds of applications and would be easier to deploy than existing technology.

We desire the following kinds of dynamism. First, the network should be self-configuring at all levels. Second, the network should not rely on any fixed or wired infrastructure, since such reliance would hinder deployment. Third, nodes should be able to move. Fourth, applications should be able to interact easily with a changing set of nearby resources. Providing these forms of dynamism without sacrificing scalability is our main research goal.

We plan to test and improve our ideas by designing, implementing and deploying a working prototype. An ideal prototype would be incrementally deployable, would involve mobility, would suggest novel applications, and would not force undue focus on issues such as battery life and equipment weight. We intend to place nodes in cars, where they will support applications such as cooperative highway congestion monitoring, fleet tracking, and discovery of nearby destinations. For this reason we call the proposed project CarNet.

2. Approach

CarNet will need to provide scalability and self-configuration at a number of different levels.

At the lowest level, nodes will communicate with each other using broadcast radios. This is likely to require sophisticated power and spectrum management to cope with variable densities of nodes. We intend to adopt relevant ideas developed for the cellular telephone network; the challenge will be to adapt them to an ad-hoc environment.

Nodes that are not in direct radio contact will need to send packets along paths through intermediate nodes. Finding such paths using conventional general-purpose routing protocols would scale badly, since they flood topology descriptions over the entire network. However, a radio-based network's topology will correspond closely to geography. This allows packets to be addressed to geographical locations, and makes forwarding by intermediate nodes as simple as sending a packet to the neighbor geographically closer to the destination. Geographical routing scales well because intermediate nodes need only know their own and their neighbors' locations in order to forward packets. Each node can discover its location by consulting a GPS receiver, or by asking a nearby GPS-equipped node. Though the basic idea is simple, a useful implementation would have to cope gracefully with ``holes'' in the distribution of nodes and with moving nodes.

Geographical routing requires that sending nodes discover the location of destinations. That is, the network must provide a database that maps each node's permanent ID to its current geographical location. The database ought not to depend on any special fixed infrastructure that might hinder easy deployment or scalability; instead, it should be distributed over all the nodes. The database ought to be robust: the failure of a few individual nodes should affect the reachability of at most a few other nodes. Finally, the location database should be efficient: the closer two nodes are, the more quickly they should be able to learn each others' locations.

We are working on distributed location server algorithms that fulfill these requirements, using the following general approach. All nodes agree on a hash function H(id) that maps each permanent node ID to a set of geographical coordinates. Each node X sends location updates to
a subset of H(X), favoring those that are closest. The location updates are accepted and stored by the node closest to each coordinate; those nodes act as X's location servers. When node Y wishes to find X's current location, it sends queries to some of the positions in H(X), trying nearby coordinates first. As soon as Y queries a coordinate to

which X sent an update, Y will learn X's position. This description omits many details that increase the probability that Y will successfully contact one of X's location servers.

We plan to deploy equipment running the above algorithms in cars. Each car would have a node consisting of an embedded Linux computer, an 802.11 radio, a GPS receiver, and displays for the driver and/or passengers. The nodes will be able to run standard Internet applications, such as web browsers, using CarNet to contact the wired Internet. We will also produce applications that take advantage of this style of networking. Perhaps the most interesting example would display current traffic congestion levels on a highway map, allowing the driver to plan a low-congestion route. The congestion information would be acquired from nearby nodes' reports of their positions and speeds.

3. Impact

The primary impact of this work will be an architecture and protocols to support very large ad-hoc wireless networks. Such an architecture will enable the use of data network technology in a number of important situations. First, in large networks of mobile nodes. Second, in networks where completely automatic configuration (and reconfiguration) is necessary, such as networks of appliances or sensors. Third, in situations where wired data connections are awkward, expensive, or non-existent; this might include ordinary Internet access in some areas. Fourth, in applications in which purely incremental deployment costs are desirable, ruling out fixed infrastructure.