## Router Capabilities (RC) Protocol

A new protocol that enables the automatic configuration of a complex home network by allowing home routers to advertise their capabilities is contemplated.

Home routers would use a new protocol, the Router Capabilities (RC) protocol, to determine the edge of the home network and to advertise and aggregate the home's need for IPv6 prefixes, thus supporting a non-uniform, dynamic, hierarchical architecture.
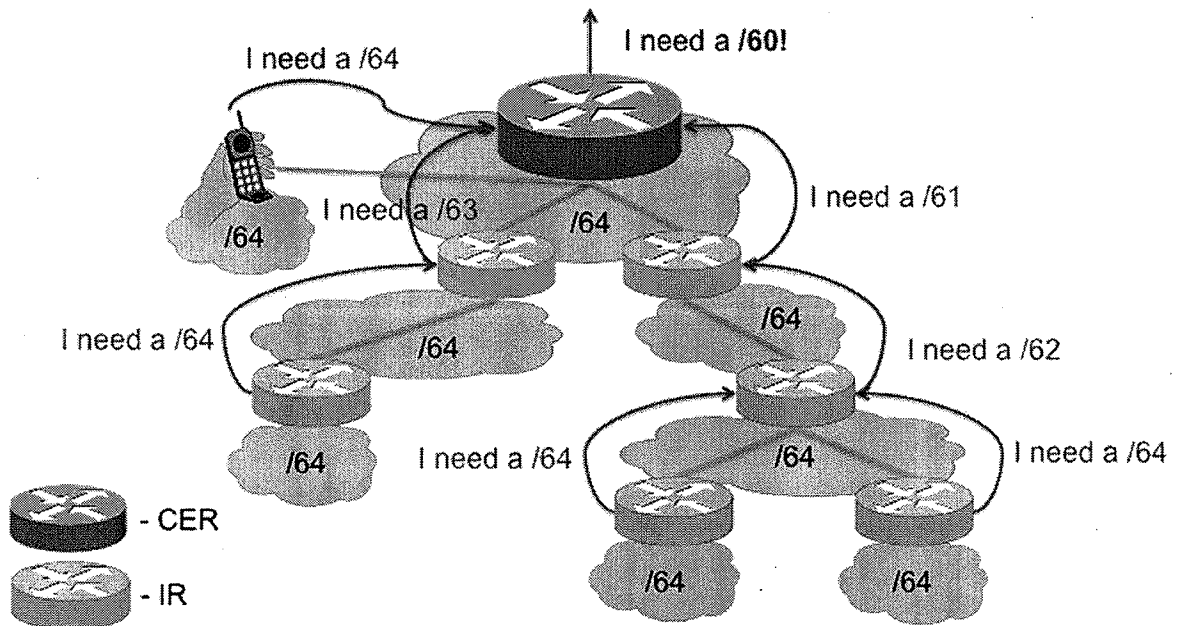
Procedure:
1. All home routers announce themselves on all interfaces
   a. RC is link-local communication sent to all-routers (FF02::2)
2. Customer Edge Router (CER) is determined by not receiving RC on WAN
3. After discovery, Internal Routers (IRs) announce their needed prefix size "up" towards the root of the tree
   a. Each IR aggregates downstream requests when present
4. CER aggregates all requests and sends that as a "hint" to the ISP DHCPv6 server
5. Home routers hand down prefix' using DHCPv6 based on requests received through RC
6. Home routers install a default "up" route and routes for all directly attached and downstream prefix' based on DHCPv6 leases

This Router Capabilities Protocol could also be generalized to send up other relevant information as well (e.g. Firewall state etc.) in the future.

One of the most efficient architectures for a complex home network is a non-uniform, dynamic, hierarchy. The problem with this network model is that routers do not send Router Advertisements (RAs) when there is no WAN prefix present and without being triggered by RAs, downstream routers will not initiate DHCPv6. This means that there is no way for a home network to determine from the bottom up what its specific prefix needs are. The Router Capabilities (RC) protocol provides a mechanism to learn prefix requirements from the bottom up.

Home networks are becoming more complex, and are expected to continue to do so. Configuring a multi-router home network without any user intervention is a difficult problem to solve. The Router Capabilities (RC) Protocol described herein is one solution. It enables efficient use of the IPv6 address space and delegation of prefixes by the operator.