Dynamic Address Assignment

In the early days of the Internet, every host was assigned a unique IP address that was hard-coded into its configuration files. As the Internet grew in popularity among less technically savvy users, and diskless devices became prevalent, it became impractical to manually change the IP address of each machine every time you changed networks. (This became even more important when laptops and wireless networking became commonplace.)

To solve this problem, a number of protocols were developed culminating in the modern Dynamic Host Configuration Protocol (DHCP) standard, IPv6 neighbor discovery, DHCPv6, and link-local addressing. These protocols are described further in the sections that follow.

Dynamic Host Configuration Protocol (DHCP) and DHCPv6

In IPv4, DHCP provides a means for a computer or other device to ask a central server for an IPv4 address suitable for use on the network. Depending on the server, the IPv4 address can be either assigned randomly from a pool of available addresses or assigned manually by an administrator based on the requesting host's MAC address.

In IPv6, the DHCPv6 protocol can optionally be used in a similar way, though IPv6 addresses can also be acquired through stateless address autoconfiguration, as described in the next section.

A DHCP server can also provide additional information needed for routing—the subnet mask and router address, for example—as well as domain name servers, directory servers, and other arbitrary data as defined by various extensions to the protocol.

The way DHCP works (at a high level) can be summarized as follows: the client broadcasts a request for an IPv4 address. After the server assigns one, the client is said to own a "lease" for that IP address that lasts a particular period of time. The client may renew that lease at any time until the expiration date. The server is not obligated to agree to extend the lease, but most servers do.

Generally speaking, if the client is connecting for the first time, this exchange is a series of broadcast UDP packets. If the client is renewing an existing (still valid) IP lease, the exchange can be sent with unicast UDP packets (at the client's discretion).

The client may continue to send requests for additional data (DNS servers, network volume mounts, Active Directory domain controllers, and so on) by sending requests with extra options and waiting for the reply.

Neighbor Discovery and IPv6 Address Assignment

The IPv6 protocol (built on top of ICMPv6) provides built-in support for neighbor discovery. Neighbor discovery serves two purposes:

- Discovering the hardware address of other hosts on the same physical network. This takes the place of ARP.
- Stateless address autoconfiguration (SLAAC). This is an alternative to much of the functionality provided by DHCP.

When a host first connects to a network, it initially uses a self-assigned IPv6 address, which is globally unique by design. This address allows the host to send neighbor discovery requests to solicit a router. The router then provides the same sort of information that would appear in a DHCP offer under IPv4—the IP address of the router, the network prefix (which is roughly comparable to the network address and subnet mask), the IP addresses of DNS servers, and so on. From there, the client can construct a routable IPv6 address to use when communicating with the public Internet.

In addition to the self-assigned link-local address, each host constructs at least two additional IPv6 addresses: a permanent address and a privacy address.

The host constructs its permanent address based on the host's physical (link-layer) address. This results in an address whose host part generally remains unchanged as the computer moves from network to network. This address is intended to provide a consistent location where other hosts on the network can send data *to* the host in question.

Additionally, each host constructs one or more privacy addresses. These addresses are generated randomly and change over time. Thus, they are not tied to a particular piece of hardware. By default, outgoing connections are sent using a privacy address.

Note: When registering services for public consumption, however, apps should not use the privacy address because it will change. If you use Bonjour to register your service, it will handle these details for you. Bonjour is described in the next section.

Link-Local Addressing and Bonjour

Domain names are a great solution for permanent servers, but they cost money and require a technically savvy user to configure them. Also, for non-public servers, a globally published domain name makes little sense. Further, standard domain names are fairly challenging to use with dynamic IP addresses assigned by DHCP servers and similar schemes. With an increasing number of dynamically configured networks, another solution was needed.

To provide an alternative, OS X and iOS support Bonjour, an implementation of zero-configuration networking. Bonjour consists of three parts:

- Link-local addresses in IPv4—A means for self-assignment of IP addresses in the absence of a DHCP server or other means of IP address assignment. (Link local address assignment is built in to the IPv6 protocol itself, and thus there is no need for Bonjour to duplicate this functionality.)
- Multicast DNS—A technology for providing DNS resolution when an infrastructure server is not present or when local, unmanaged names are more convenient.
- DNS Service Discovery—A means of registering and discovering services.

The combination of these technologies allows multiple hosts on the same physical network to advertise services and discover one another without the need for a permanent DNS infrastructure.

Typically, Bonjour uses multicast DNS to send a DNS Service Discovery query to every machine on a local network, asking whether any of them provide a particular service. Each machine that provides the requested service then sends a message back to the original machine. The result is that those machines can discover and use services provided by other machines on the network without needing to tell each machine about those services ahead of time.

DNS Service Discovery can be combined with link-local addresses to allow zero-configuration networking—discovering networked devices even if you have no network infrastructure configured at all. For example, you might share files between two computers over an ad hoc wireless network.

The way link-local addresses work is fairly straightforward.

- For IPv4, if a host fails to obtain an address from a DHCP server, the computer chooses a random IP address in a particular range (`169.254.*.*`), then uses a link-specific protocol (such as ARP) to ask who has that address. If another machine responds, it tries a different address. If no machine responds, the host claims that number as its own.
- For IPv6, every interface has a link-local address within the `fe80::` prefix, regardless of whether the interface has another address assigned through SLAAC, DHCPv6, or other mechanisms. This address is computed based on the interface's hardware address if the interface has one (an Ethernet MAC address, for example). If it does not, the address is chosen randomly within that

prefix, and SLAAC’s duplicate address detection protocol is used to guarantee uniqueness. See RFC 4862 for details.

Because these IP addresses are all in the same subnet, every host with a link-local address on a given local area network can talk to every other host without going through a router. Thus, they can all discover one another without the need for any additional infrastructure.

Note: These three Bonjour technologies can also be used independently of one another. In particular, multicast DNS can be used to look up the name of a machine on the local network without discovering services. Similarly, traditional infrastructure DNS servers can provide information about services through Wide-Area Bonjour.