IPv6 Operations
Zeroconf

ITL

Information Sources

- zeroconf IETF working group (closed)
  - http://www.zeroconf.org/
- zeroconf was driven by Apple, which has implemented the specs under the names “Rendevous” and “Bonjour”
  - http://developer.apple.com/networking/bonjour/specs.html

Routers

- Cisco IOS 12 and up supports IPv6
  - Interface commands
    - ipv6 enable
    - ipv6 address ...
  - Routing
    - ipv6 route ...
    - ipv6 router rip
  - Information
    - show ipv6 interface ....
    - show ipv6 route

Hosts

- Mac OS
  - IPv6 built-in
  - “Configure IPv6” button in the network control panel
- Linux
  - IPv6 built-in
  - ifconfig eth0 inet6 ...

Windows

- Microsoft started a developers preview of the integrated IPv6 stack in Win2000
  - Command line configuration utilities
  - ping6 and tracert6
  - Updated Internet Explorer
- Same code ships (disabled) with WinXP
- IPv6 enabled by default in Win 2003
- netsh utility used for configuration

ZEROCONF components

- Address assignment
  - standard in IPv6
  - retrofitted into IPv4
- DNS without a server
  - mDNS (Multicast DNS)
- Service location discovery

Side Note: Neighbor Discovery Protocol

- ND replaces ARP in IPv6.
- Key Concept: The solicited-node multicast address (RFC 2373)
  - Start with FF02:0:0:0:0:1:FF00::/104 (a multicast prefix)
  - Fill the low-order 24 bits in the multicast address with the 24 low-order bits from every configured address, e.g. 4037::01:800:200E:8C6C -> FF02::1:FF0E:8C6C

ND continued

- A node must join all solicited-node multicast groups created from its addresses
- Since all global addresses have the same interface ID field, they should all be in the same multicast group
- Only a small number of hosts will “see” each ND solicitation.

Zeroconf Address Assignment

- IPv6
  - Link-Local FE80::/16
  - Duplication Address Discovery (DAD)
- IPv4
  - 169.254.0.0/16
    - (first and last 254 addresses are reserved)
  - Random-number based address selection (MAC address used as a seed)
  - ARP-based duplicate discovery (broadcast)

Link-Local Issues

- Multiple interfaces
  - IPv6 has the “scope” syntax (...%fxp0)
  - IPv4 has no standard solution
- Address selection
  - Try to prefer routable addresses
    - stop using the local address when a global one becomes available
  - Local addresses are not globally reachable
- Always ARP/ND for link-locals

We have an Address -- now what?

- Communication via link-local is a pain
  - Need to look up raw addresses
  - Need to type addresses in directly
- DNS would be nice, but
  - there is no DNS server available, or
  - if there is a server I don't know where it is
- Side-note
  - DHCP may be used, but that is a "heavy" solution for one piece of information. The router advertisement tells the host to go on to DHCP6.

Local Name Discovery

- Has long been used on Mac OS, Windows, and Novell
  - NETBIOS Names
  - AppleTalk
- Broadcast-based name announcements
- “Chatty” Protocols

Multicast DNS (mDNS)

- Issue an (almost) standard DNS query
- Target is not a DNS server but a multicast address
  - 224.0.0.251 for IPv4
  - FF02::FB for IPv6
- Partially implemented clients send standard DNS queries to/from port 53
- mDNS clients send to/from port 5353
  - and the DNS packet semantics (not structure) changes slightly

mDNS Implementation

- Client wants to resolve a name
  - Multicast the query
- One or more members of the multicast group reply with an answers
  - One reply for unique information (name to address)
  - Many replies for shared information (services)
- Replies are multicast to allow all clients to use the answer

mDNS Implementation

- Names in the ".local." domain are resolved by mDNS (by default)
  - No Hierarchy is implied or allowed
  - Replies must have an IP TTL of 255
    - Protect against attackers injecting malicious answers from outside the network
- Apple Mac OS 9 and OS X
  - "Bonjour" built-in (IPv4 and IPv6)
- Linux
  - mDNSResponder or avahi

Name/Content Assignment

- Unique information, e.g. host names
  - Host creates a name it wants to use
  - Issues a query to see if there is a conflict
  - Host who got the name first “wins”
  - In a race condition (two hosts start using the same name at the same time) the one with the lower address “wins”
- Shared information
  - Host responds to queries as appropriate