Multiple Provisioning Domains

Ted Lemon <Ted.Lemon@nominum.com>
Motivation

• Many problems occur when a device is multiply connected
• Current IP stack and library implementations do not adequately address these problems
• Vendors with substantial motivation have not solved this problem
• The solution requires fundamental changes in the default architecture of IP stacks.
• We are the people who ought to be architecting this change
Define provisioning domain

- A single collection of configuration information...
  - Address and routing information
  - DNS resolver address
  - Other stack configuration
- ...that comes from a single provider
- May be more than one provisioning domain per link
  - Dual stack: two PDs
  - Device behind multihomed router
Illustrative examples

• Turn on VPN, printer goes away
• Turn on VPN, go to Pirate Bay, get fired
• Connect to captive portal hotspot, mobile data connection stops working
• Connect to LTE and Wifi, LTE is faster, Wifi adequate, next LTE bill is... surprising...
• Maastricht hotel, Wifi does port 80 and port 443 really fast, and nothing else, I have to choose
• Wifi and ethernet provide default route, only one works, stack picks the wrong one every time
Three or more related issues

• Sometimes we can get guidance from network
  • If the guidance is right, it can be helpful
• Sometimes guidance is wrong
  • VPN: “send me everything, even your printer traffic!”
  • Captive portal: “send me everything, so that I can pop up a login screen!”
  • Broken network: “send me everything, so that I can dump it in the bit bucket for you”
Related issues cont’d

• Sometimes guidance is helpful but insufficient
  • LTE: I am faster than your WiFi and can handle all your traffic
  • LTE: I am expensive and you shouldn’t use me unless you have to, but I can’t say when that is
Not a “selection” problem

• The traditional response has been to pick one and stick with it until something changes
• This works if both connections are equivalent, but one is always preferable
• That is frequently not the case
Not a “happy eyeballs” problem

• Happy eyeballs is interface-agnostic
• Sometimes you really do want to select an interface
Common underlying problem

- Everything is mixed together in the same bucket
- No way to say “use this provisioning domain”
- No way to say “try this provisioning domain”
Fixing the problem

• Isn’t MIF handling this?
  • Yes, but we can’t do it alone
• Problem touches various configuration protocols
• It also requires features not present in modern stacks and libraries
• I think there is work here to for 6man (RA, source address selection, etc), DHC (DHCPv6), MIF (architecture doc, maybe API doc), probably others
• We need participation
Can you help?

• If not you, then who?
• If not now, then when?
• If you’re interested, please talk to me, to Margaret, and or to Hui
• Ted Lemon <Ted.Lemon@nominum.com>
• Margaret and Hui <mif-chairs@ietf.org>