



Prefix assignment and other configuration information

Ole Trøan
Homenet Interim, Philadelphia

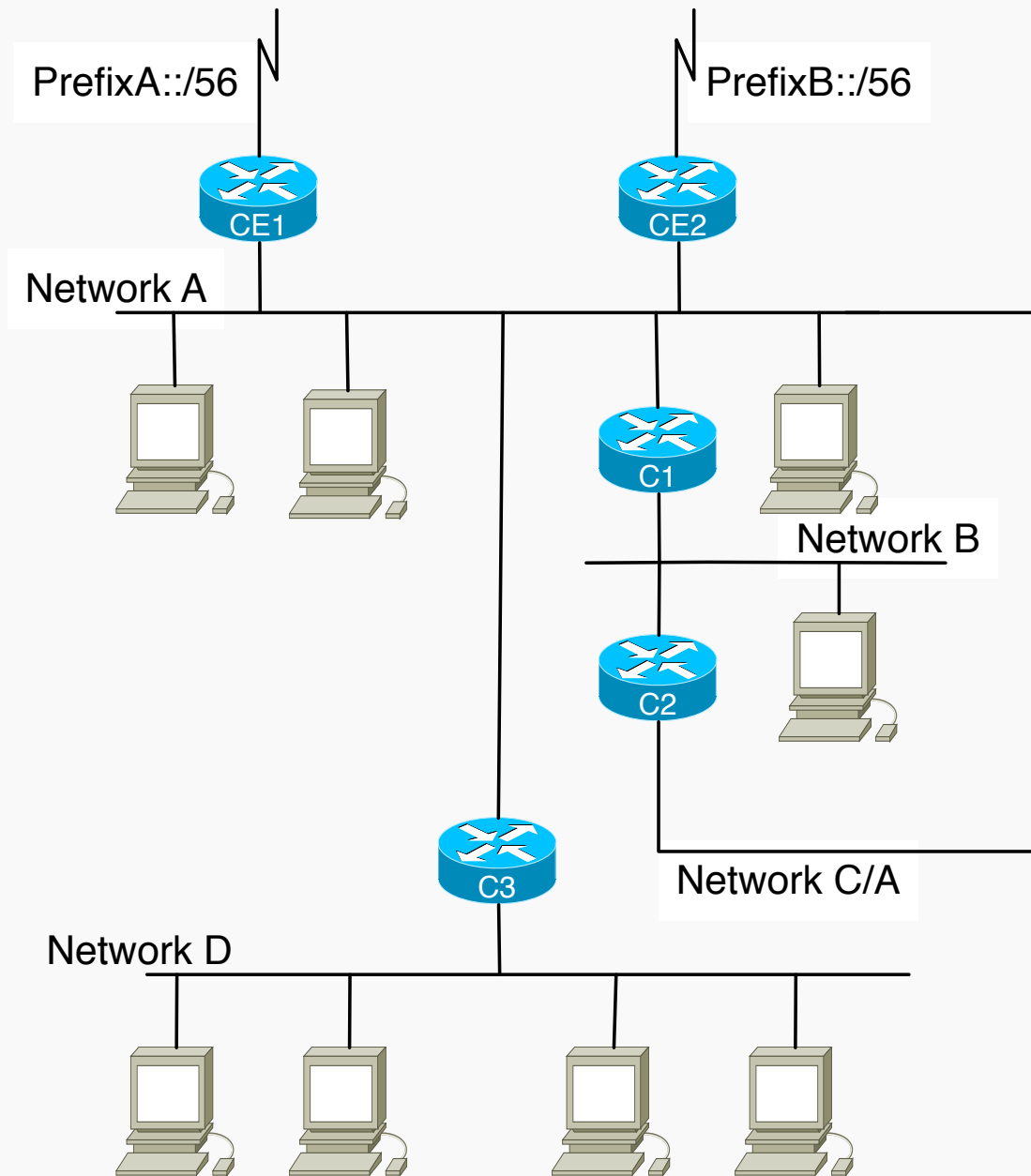
2011-10-06



Discussion

- Requirements: 40 minutes
- Solutions: 40 minutes
- Summary / Scoring: 40 minutes





Requirements:	Must,Nice,Don't care
Arbitrary topology (or only directed graph)	
Multiple sources of information	
Stable prefix/CI over time	
Single prefix per link per delegated site prefix	
Partitioning and merging	
Efficient use of prefix space	
Media type independent	
Autonomous configuration	
Require no host changes	
Detect administrative boundaries	
Not require dedicated upstream/downstream ports	
Route aggregation	
Works for both IPv4 and IPv6	

Existing solutions:

1. Layer 2/RFC6204 topology – null solution
2. Multilink subnet routing
Hosts routes combined with ND flooding (draft-ietf-ipv6-multilink-subnets-00)
3. Hierarchical DHCP Prefix Delegation
[RFC3633 + draft-chakrabarti-homenet-prefix-alloc-00](#)
4. Flat DHCP Prefix Delegation
[RFC3633 + draft-baker-homenet-prefix-assignment-00](#)
5. Zeroconf OSPF
[draft-chelius-router-autoconf-00](#) / [draft-dimitri-zospf-00](#)
<http://conferences.sigcomm.org/sigcomm/2007/ipv6/1569041157.pdf>
6. NAT (IPv6 NAT or NPT66)
7. Others



Flooding versus Request/Reply

- Request/Reply may require a God server
Make handling multiple sources of information difficult
- Request/Reply requires some way of discovery of the “server”.
- Flooding is distributed state. All routers in the network has the same view of the network. Assuming link-state routing here.
Requires “collision detection”
- Both assumes faith sharing that the node injecting site-prefix or being the server are co-located with the border router

#2 Multilink subnet routing

- draft-ietf-ipv6-multilink-subnets (issues in RFC4903)
- Single /64 covering the whole site.
- Allows host mobility within site.
- Breaks prefix (/64) based policy rules
- Host routes advertised in an IGP
- Requires flooding of ND solicits for host discovery / address resolution or ND register, or “SAVI” - registration
Flooding has issues in looped topologies.
- No on-link prefix.
- /64 flooded in routing protocol or by RA proxy.

#3 Hierarchical DHCP PD

- Uses existing RFC3633 + [draft-chakrabarti-homenet-prefix-alloc-00](#)
- Splits the delegated prefix into pieces. Router acts as a requesting router upstream and a delegating router downstream
- Inefficient use of address space.
- Requires a gateway with notion of upstream and downstream interfaces
- Ends up with multiple prefixes for links with multiple routers
- How does it handle loops / arbitrary topologies?
- Doesn't work well with multiple sources of information

#4 Flat DHCP Prefix delegation

- RFC3633 + [draft-baker-homenet-prefix-assignment-00](#)
- Central DHCP server(s). Other internal routers are DHCP relays *or* there is server discovery
- Efficient use of address space
- Handles arbitrary topologies if the DHCP server aka God server is all-knowing. E.g. participates in a link state IGP.
- Doesn't handle multiple sources of information unless all DHCP servers are discovered.
- Requires gateway to have defined upstream and downstream interfaces.
- May result in multiple prefixes on links with multiple routers

#5 Zeroconf OSPF

- [draft-dimitri-zospf-00](#) (part of the zerouter effort)
- New LSA in OSPF advertising a site prefix
- Each router picks a subnet id, does collision detection.
- Designated router 'owns' assigning a prefix for that link
- Handles multiple sources of information
- Partitioning and merging may result in renumbering
- Quite efficient use of address space
- Flooding of "site" prefixes. Routers are autonomous and makes up their own prefixes for each link. SLPAC...
- Border discovery, anything that is a zOSPF neighbour is within the site.

#6 IPv6 NAT

- RFC5684 – Unintended Consequences of NAT Deployments
- “Just like IPv4”
- Problems with arbitrary topologies
- NPT66 requires a separate prefix to NPT into
So requires prefix assignment anyway?

#7 Other work

- MANET – YAAP
draft-herberg-autoconf-yaap-00
- NOA-OLSR
draft-mase-manet-autoconf-noaolsr-01
- NEMO?
- RPL?
- Stateless Prefix Autoconfiguration
Using RAs