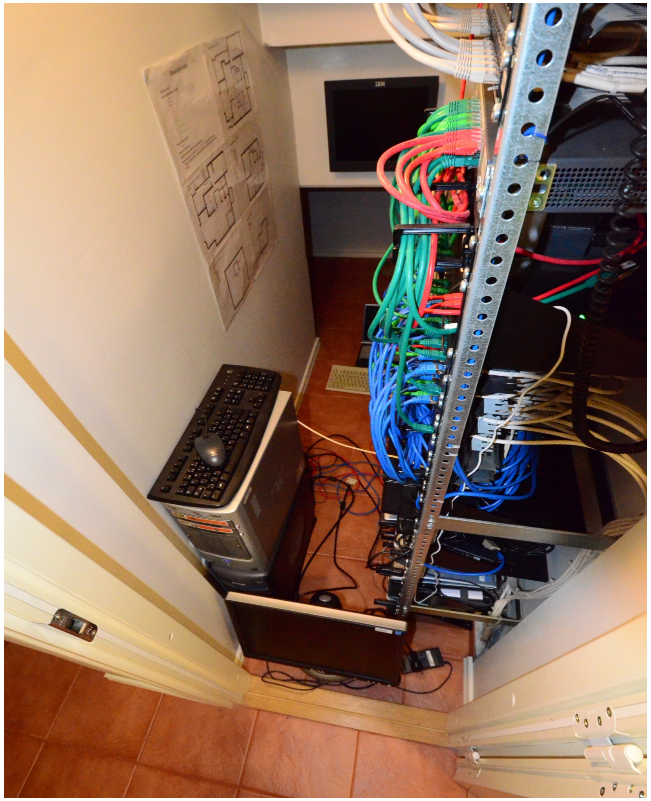
Prefix Assignment in a Home Network with OSPFv3

draft-arkko-homenet-prefix-assignment-02.txt

Jari Arkko, Ericsson

Acee Lindem, Ericsson

Benjamin Paterson, Cisco



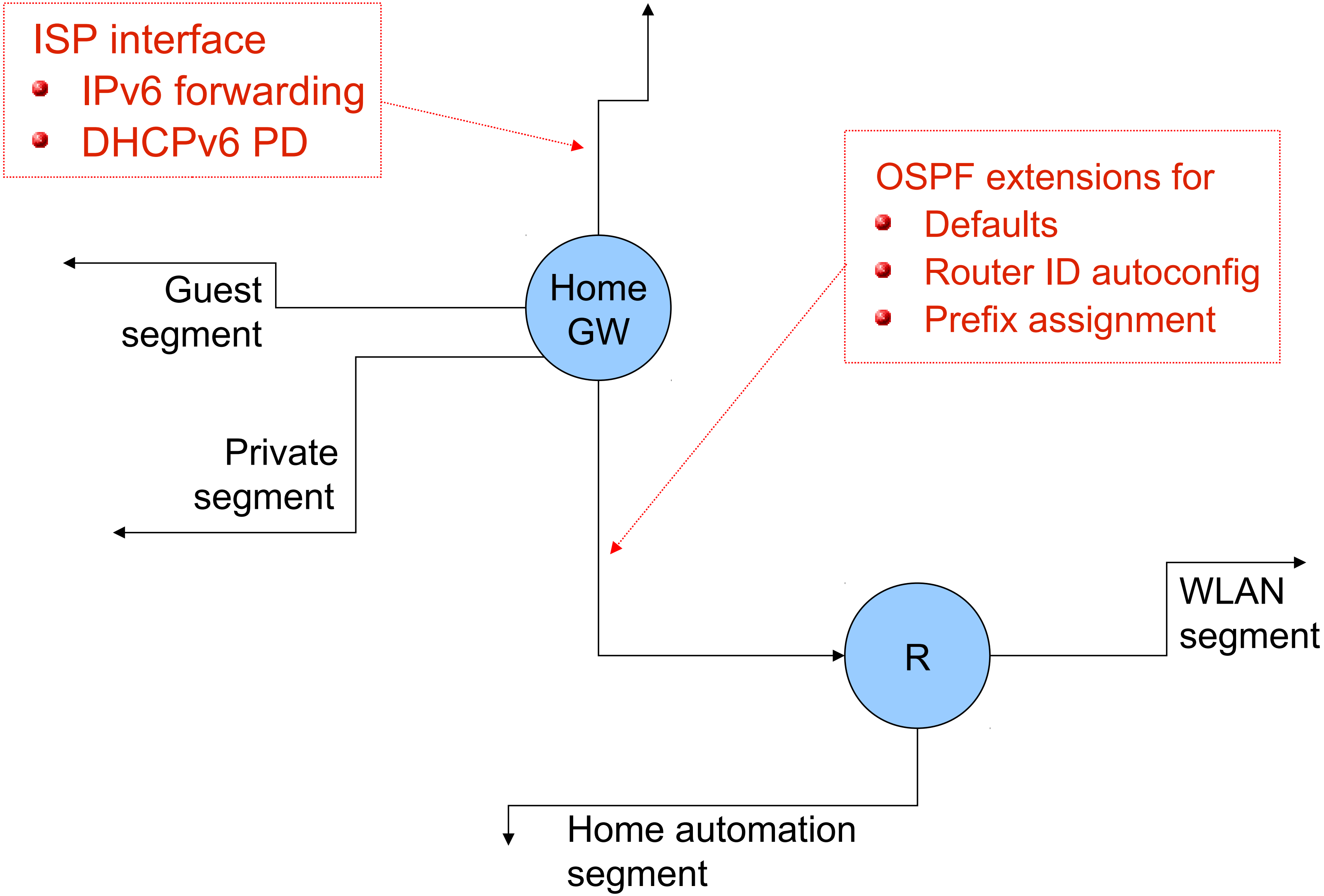
The Dream

No matter how many boxes you have
And how you connect them

- Networks shall have address space
- Routers shall know where to send packets
- Names resolve to addresses
- Human touch is not required

  [Especially by my mother!]

OSPF-Based Home Networking



ISP interface

- IPv6 forwarding
- DHCPv6 PD

OSPF extensions for

- Defaults
- Router ID autoconfig
- Prefix assignment

Home GW

Guest segment

Private segment

R

WLAN segment

Home automation segment

Status Report

Two implementations up and running!

Generally, seems to work well

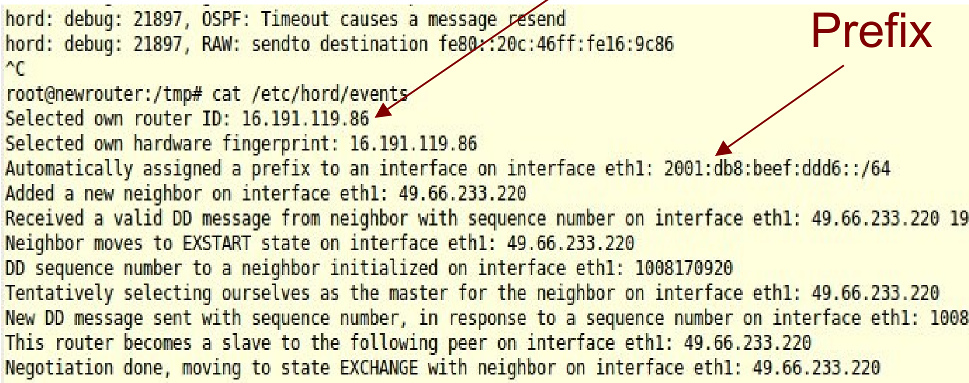
But implementations are early & incomplete

The protocol design is still morphing
(but that was the point of these exercises)

Basic Approach

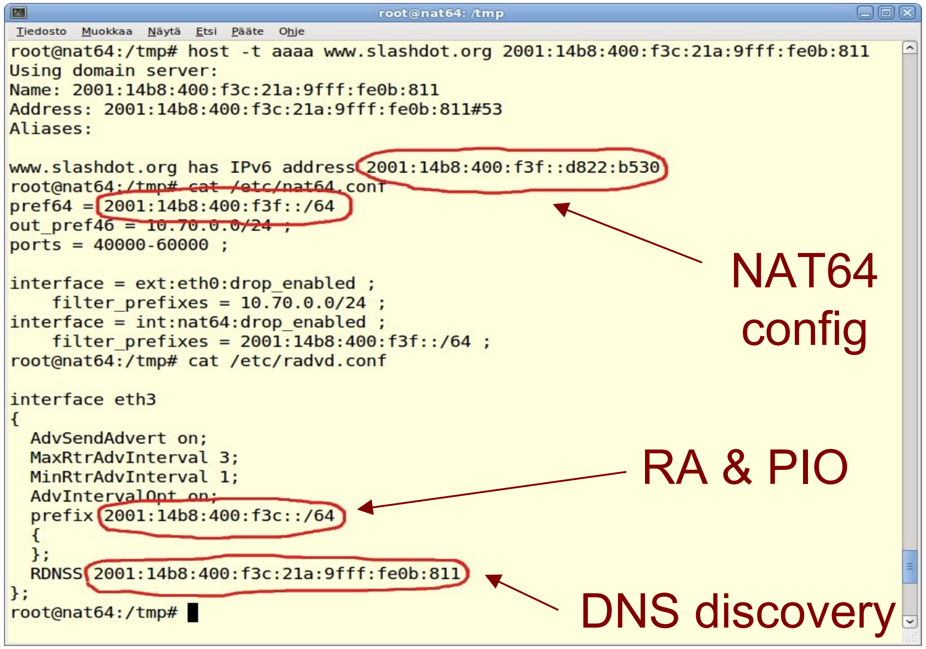
- Bootstrap OSPFv3 itself using draft-acee
- Discover an aggregated prefix for the home (e.g., via DHCPv6 PD)
- Flood the aggregated prefix to all routers via OSPFv3, using a TLV within the AC LSA
- Run a distributed algorithm to assign /64 prefixes out of the aggregated prefix to all subnets
- Flood these assignments via OSPFv3, using another TLV within the AC LSA
- Assignments registered in flash memory (stability)
- Prefixes can be used for normal IPv6 traffic, NAT64, sensor GWs, ...

Router ID



Prefix

```
hord: debug: 21897, OSPF: Timeout causes a message resend
hord: debug: 21897, RAW: sendto destination fe80::20c:46ff:fe16:9c86
^C
root@newrouter:/tmp# cat /etc/hord/events
Selected own router ID: 16.191.119.86
Selected own hardware fingerprint: 16.191.119.86
Automatically assigned a prefix to an interface on interface eth1: 2001:db8:beef:ddd6::/64
Added a new neighbor on interface eth1: 49.66.233.220
Received a valid DD message from neighbor with sequence number on interface eth1: 49.66.233.220 19
Neighbor moves to EXSTART state on interface eth1: 49.66.233.220
DD sequence number to a neighbor initialized on interface eth1: 1008170920
Tentatively selecting ourselves as the master for the neighbor on interface eth1: 49.66.233.220
New DD message sent with sequence number, in response to a sequence number on interface eth1: 1008
This router becomes a slave to the following peer on interface eth1: 49.66.233.220
Negotiation done, moving to state EXCHANGE with neighbor on interface eth1: 49.66.233.220
```



```
root@nat64:/tmp# host -t aaaa www.slashdot.org 2001:14b8:400:f3c:21a:9fff:fe0b:811
Using domain server:
Name: 2001:14b8:400:f3c:21a:9fff:fe0b:811
Address: 2001:14b8:400:f3c:21a:9fff:fe0b:811#53
Aliases:

www.slashdot.org has IPv6 address 2001:14b8:400:f3f::d822:b530
root@nat64:/tmp# cat /etc/nat64.conf
pref64 = 2001:14b8:400:f3f::/64
out_pref46 = 10.70.0.0/24 ;
ports = 40000-60000 ;

interface = ext:eth0:drop_enabled ;
    filter_prefixes = 10.70.0.0/24 ;
interface = int:nat64:drop_enabled ;
    filter_prefixes = 2001:14b8:400:f3f::/64 ;
root@nat64:/tmp# cat /etc/radvd.conf

interface eth3
{
  AdvSendAdvert on;
  MaxRtrAdvInterval 3;
  MinRtrAdvInterval 1;
  AdvIntervalOpt on;
  prefix 2001:14b8:400:f3c::/64
  {
  };
  RDNSS 2001:14b8:400:f3c:21a:9fff:fe0b:811
};
root@nat64:/tmp# █
```

NAT64 config

RA & PIO

DNS discovery

Draft-02 Updates

- Added an algorithm to generate ULAs (S. 7)
- Replaced the old algorithm for prefix assignment with a new one (S. 6.3)
- Added an explicit discussion of hysteresis (S. 8)
- Added a requirement to support DNS discovery (S. 4.2)
- Described the design choices (S. 5)
- Added Benjamin as an author
- Various small bug fixes and editorial changes

Prefix Assignment Algorithm in Draft-02

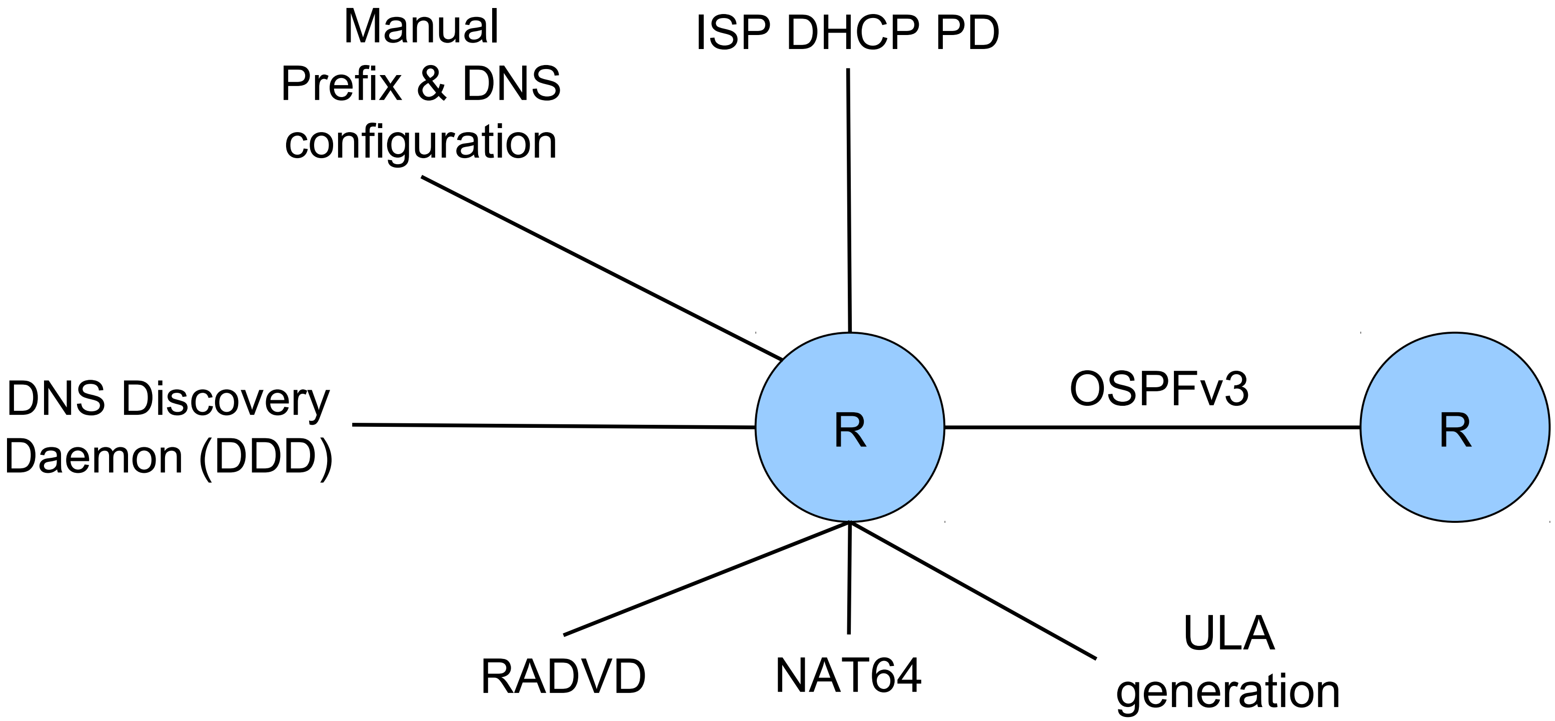
- Assigns /64 prefixes out of an aggregated prefix (e.g., a /56)
- One or several routers in the network know the aggregated prefix(es), all routers co-operate to make the assignments
- The algorithm is triggered by changes in the LSA database or the set of interfaces this router has
- Benjamin's thesis demonstrates some properties relating to the algorithm (convergence, some aspects of correctness, ...)

Experiences

Here are some experiences:

- The technology seems to work as intended
- Our understanding of the problems developed as the work continued, e.g., on conflicts, naming, interfaces
- Relatively easy to implement, 2-4 KLOC on top of an existing OSPFv3 implementation
- It is important to think about interfaces to other systems

Interactions with Other Parts



Topics for Further Discussion 1

- Hysteresis, algorithms, ULA generation probably need more review & experience
- Interactions with other systems need to be described in greater detail
- One aspect of basic network service is name resolution. How does a host deep in the network resolve, e.g., www.ietf.org?
  - Not an OSPF problem
  - I discover name servers by looking at DHCP, RAs, and other means in my implementation
  - (But if a router discovers a DNS server, how does it tell other routers about this find?)

Topics for Further Discussion 2

- Do we need a priority mechanism to decide allocations when there is not enough space?
  - Or even (shock!) a > 64 bit prefix solution?
- Alternative designs for LSAs used by the algorithm:
  - Approach 1: TLVs within the AC LSA (draft-02)
  - Approach 2: Just use intra-area-prefix LSAs
  - Approach 3: A provisional assignment LSA followed by actual allocation LSAs (app. 1 or 2)