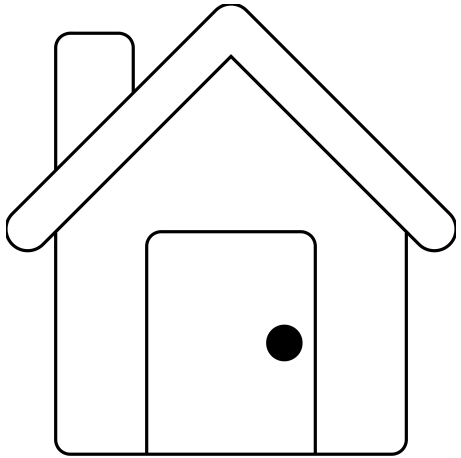
Home Networking Control Protocol



draft-stenberg-homenet-hncp-00

Steven Barth & Markus Stenberg

Experience with OSPFv3

Pros

- Known-stable and robust protocol
- Already contains mechanisms to synchronize arbitrary data

Issues

- Slow to converge (can take unexpectedly long from a user perspective)
- No stable upstream support for auto-configuration or source-dest-routing
- Difficulty to get support for transporting non-routing data upstream
- Doubts about everyone agreeing on OSPF

Running Code: “hnetd”

Upstreamed to OpenWrt-routing package feed

- available as nightly snapshot packages via opkg

Works with IPv6 and IPv4

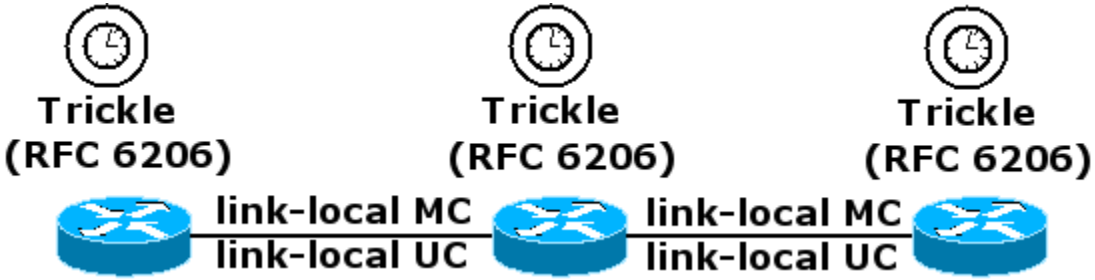
- IPv4 has some multihoming limitations vs. IPv6

(Mostly) portable to other Linux or POSIX-based OS

Full reference source-code + some instructions available

- See: www.homewrt.org (GPLv2, funded by Cisco Tech Fund)

Data Synchronization



Nodes exchange UDP datagrams via link-local IPv6

Nodes announce hash of ordered data-set via multicast

- Using Trickle algorithm as trigger

Iff nodes disagree about state, synchronize via unicast

Data Model

Each router has

- A unique identifier
- A synchronized config data-set (ordered set of TLVs)
  - + The Relative time since last publishing the set
  - + A sequence number (“version”) of the set
  - + A hash-value of the set for easy comparison
- A public/private key-pair for authentication
  - Required for best security, but can operate without
  - Designed to integrate with draft-behringer-homenet-trust-bootstrap

Security

1. No authentication
   no configuration required
   security level comparable to unauthenticated RA, DHCPv6, etc.

2. IPsec with pre-shared key
   some configuration required
   shared trust for all routers in homenet + optional encryption

3. Public / private key authentication (no confidentiality)
   suitable key management architecture and bootstrapping required
   individual trust for each router in homenet

Payloads, Extensions and Glue

1. Prefix & Address Assignment
   TLVs for draft-pfister-homenet-prefix-assignment

2. DNS & DNS-Service Discovery
   TLVs for draft-stenberg-homenet-dnssd-hybrid-proxy-zeroconf

3. DHCP(v6)-encoded auxiliary information

   DNS-servers, SIP, NTP, etc.

Discovering Borders

Interface is external if at least one of the following applies:

(loosely based on draft-kline-homenet-default-perimeter)

1. It is defined fixed-external by the router (e.g. dedicated WAN-Port).
2. A delegated IPv6-prefix could be acquired.
3. An IPv4-address could be acquired via DHCP (see below though!)
4. HNCP security is enabled and there are unauthenticated routers.

Avoiding clashes with other HNCP-routers

(allows HNCP routers to serve PD to RFC 6204/7084 routers and IPv4 to work)

1. DHCP(v6) clients MUST include the userclass “HOMENET”
2. DHCP(v6) server MUST ignore requests with the userclass “HOMENET”

Routing Protocols

Any routing protocol can be used “as is”, as long as:

1. The protocol supports source-dest forwarding
   (e.g. draft-baker-rtgwg-src-dst-routing-use-cases)

2. The implementation is “self-configuring”
   (enabled without user intervention, defaults that work, etc.)

HomeWRT implementation currently supports Babel, as the version available at this time in OpenWrt best meets these requirements.

Routing selection and fallback

“Graceful Fallback”

- In case no common routing protocol is available...
- Build simple tree-topology by breadth-first traversing the PA information
- Likely not optimal, but packets still flow

Selecting a routing protocol

- Routers announce their routing capabilities and preferences
- Routers select the most-preferred protocol supported by all
- HNCP currently draft identifies: RIP, Babel, OSPF, IS-IS

Thank You

Do you have questions or feedback?

We'll be around for more discussions or in case you are interested in trying out the implementation.

Please also visit www.homewrt.org for source code, binaries and some documentation.