

Simple Homenet Naming and Service Discovery Architecture

draft-tldm-simple-homenet-naming-00

Ted Lemon <mellon@fugue.com>

Daniel Migault <daniel.migault@ericsson.com>



Status

- First draft, copied from previous homenet naming architecture doc
- Didn't catch some remnants of the document; these will be fixed in next rev
- All stateful stuff and security is gone; no external visibility of names
- Intended to be able to support stateful and security as an additional layer
- Relies on DNSSD Discovery Proxy (draft-dnssd-hybrid-06)

Discovery Proxy Differences

- Proxy split into two functional blocks: querying proxy and relaying proxy
- No user-supplied link names, so link names are generated via HNCP
-
- Relaying proxy:
 - Translates DNS to mDNS with no rewriting or aggregation

Querying Proxy

- Answers DNS protocol requests from hosts
- Generates queries to every relaying proxy
- Combines responses
- Rewrites names, *.local* -> *.x.homenet*
- Queries come in using DNS protocol, and go out using DNS protocol
- Query aggregation is done for hosts that don't support DNS push
- DNS push support offered for hosts that do
- One or more querying proxies per homenet; every HNR supports

Relaying Proxy

- Receives DNS protocol queries from querying proxies
- Sends mDNS protocol requests to link
- Relays mDNS responses back to querying proxies
- No name rewriting
- Assumes DNS push support on querying proxies
- One relaying proxy per link; if two HNRs connect to one link, HNCP picks
- All HNRs support relaying proxy

Name conflicts

- Names are unique per link via mDNS name defense protocol
- What about conflicts across links?
- What about link names/zone names?
- Could resolve conflicts in the Querying proxy by revealing link subdomain when conflicts exist
- Could just always present link subdomain
- How big a problem is this anyway?

Causes of naming conflicts

- Two devices with the same name by accident
- Attacker tries to take name of existing device
- One device is present on two links serially
- One device is present on two links simultaneously
- mDNS doesn't provide unique host identifier to disambiguate
- Can't use lladdr because might be different on different links
- I am not really satisfied with Discovery proxy solution
- How do others feel?

Regrets for less-simple architecture

- No security model
- No registration protocol
- No clean way to enumerate all services
- No place to collect such an enumeration
- mDNS is a flawed protocol, and we aren't fixing it

For the WG to decide

- Do we prefer this, or the more complex and wonderful naming architecture?
- Do we care that we aren't proposing a cleaner registration protocol, and therefore effectively kicking that can down the road?
- Do we do disambiguation in the UI or the infrastructure?
- Do we do ugly presentation or clean presentation; clean presentation is more work to implement and has more potential problems, but is probably better for the user nearly all of the time.