Homenet Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 17, 2015

S. Barth

October 14, 2014

HNCP - Security and Trust Management
draft-barth-homenet-hncp-security-trust-01

Abstract

This document describes threats and a security and trust bootstrap mechanism for the Home Networking Control Protocol (HNCP).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 17, 2015.

Copyright Notice

Table of Contents

1. Introduction

HNCP is designed to make home networks self-configuring, requiring as little user intervention as possible. However this zero-configuration goal usually conflicts with security goals and introduces a number of threats.

This document describes imminent threats and different security and trust management mechanisms to mitigate them.

2. Requirements language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Scope

This draft is based on HNCP as described in [I-D.ietf-homenet-hncp] and the additional threats it introduces. Many of these already exist in a similar form in current single-link home networks due to the usually unauthenticated use of protocols like NDP [RFC4861] or DHCPv6 [RFC3315]. This document intentionally does not cover these and other Homenet-related threats not explicitly introduced by HNCP.

HNCP is a generic state synchronization mechanism carrying information with varying threat potential. This draft will mainly consider the currently specified payloads:

Network topology information such as homenet routers and their adjacent links

Address assignment information such as delegated and assigned prefixes for individual links

Naming and service discovery information such as auto-generated or customized names for individual links and routers

IGP capabilities and preferences of individual routers

4. Border Determination

In general an HNCP-router determines the internal or external state on a per-link scale and creates a firewall-perimeter and allows HNCP- and IGP-traffic based on the individual results. These are provided by either automatic border discovery or a predefined configuration indicated by e.g. the link-type, a physically dedicated (labeled) port or the administrator.

Threats concerning automatic border discovery cannot be mitigated by encrypting or authenticating HNCP-traffic itself since external routers do not participate in the protocol and often cannot be authenticated by other means. These threats include propagation of forged uplinks in the homenet in order to e.g. redirect traffic destined to external locations and forged internality by external routers to e.g. circumvent the perimeter firewall.

It is therefore imperative to either secure individual links on the physical or link-layer or preconfigure the adjacent interfaces of HNCP-routers to an adequate fixed-category in order to secure the homenet border. Depending on the security of the external link eavesdropping, man-in-the-middle and similar attacks on external traffic can still happen between a homenet border-router and the ISP, however these cannot be mitigated from inside the homenet.

5. HNCP Payload Security

Once the homenet border has been established there are several ways to secure HNCP against internal threats like manipulation or eavesdropping by compromised devices on a link which is enabled for HNCP-traffic. If left unsecured attackers may cause arbitrary spoofing or denial of service attacks on HNCP-services such as address assignment or service discovery. Furthermore they may manipulate routing or external connection information in order to perform eavesdropping or man-in-the-middle attacks on outbound traffic. The following security mechanisms are defined to mitigate these threats:

5.1. Isolated router-to-router links

Given that links containing HNCP routers can be sufficiently secured or isolated it is possible to run HNCP in a secure manner without using any form of authentication or encryption. Detailed interface categories like "leaf" or "guest" can be used to integrate not fully trusted devices to various degrees into the homenet by not exposing them to HNCP and IGP traffic or by using firewall rules to prevent them from reaching homenet-internal resources.

5.2. Authentication and Encryption of HNCP-traffic

The end-to-end mechanism DTLS [RFC6347] is used to authenticate and encrypt all HNCP unicast-traffic in order to protect its potentially sensitive payload. Methods for establishing and managing trust for this mechanism are described in the following section.

HNCP also uses multicast signaling to announce changes of HNCP information but will not send any actual payload over this channel. An attacker may learn hash-values of HNCP-information and may be able to trigger unicast synchronization attempts between routers on the local link this way. An HNCP-router should therefore limit its unicast synchronizations attempts to avoid a multicast-induced denial-of-service.

6. Trust Management for Authentication and Encryption

6.1. Pre-shared secret based trust

A PSK-based trust model is a simple security management mechanism that allows an administrator to deploy devices to an existing network by configuring them with a pre-defined key, similar to the configuration of an administrator password or WPA-key. Although limited in nature it is useful to provide a user-friendly security mechanism for smaller homenets.

6.2. PKI-based trust

A PKI-based trust-model enables more advanced management capabilities at the cost of increased complexity and bootstrapping effort. It however allows trust to be managed in a centralized manner and is therefore useful for larger networks with a need for an authoritative trust management.

6.3. Certificate-based trust consensus

The certificate-based consensus model is designed to be a compromise between trust management effort and flexibility. It is based on X.509-certificates and allows each connected device to give a verdict on any other certificate and a consensus is found to determine whether a device using this certificate or any certificate signed by it is to be trusted.

6.3.1. Trust Verdicts

Trust Verdicts are statements of HNCP-devices about the trustworthiness of X.509-certificates. There are 5 possible verdicts in order of ascending priority:

0 Neutral: no verdict exists but the homenet should find one

1 Cached Trust: the last known effective verdict was Configured or Cached Trust

2 Cached Distrust: the last known effective verdict was Configured or Cached Distrust

3 Configured Trust: trustworthy based upon an external ceremony or configuration

4 Configured Distrust: not trustworthy based upon an external ceremony or configuration

Verdicts are differentiated in 3 groups:

Configured verdicts are used to announce explicit verdicts a device has based on any external trust bootstrap or predefined relation a device has formed with a given certificate.

Cached verdicts are used to retain the last known trust state in case all devices having configured verdicts about a given certificate have been disconnected or turned off.

The Neutral verdict is used to announce a new device intending to join the homenet so a final verdict for it can be found.

The current effective trust verdict for any certificate is defined as the one with the highest priority from all verdicts announced for said certificate at the time. A device MUST be trusted for participating in the homenet if and only if the current effective verdict for its own certificate or any one in its certificate hierarchy is (Cached or Configured) Trust and none of the certificates in its hierarchy have an effective verdict of (Cached or Configured) Distrust. In case a device has a configured verdict which is different from the current effective verdict for a certificate the current effective verdict takes precedence in deciding trustworthiness however the device still retains its configured verdict in its configuration.

6.3.2. Trust Cache

Each device maintains a trust cache containing the current effective trust verdicts for all certificates currently announced in the homenet. This cache is used as a backup of the last known state in case there is no device announcing an configured verdict for a known certificate. It SHOULD be saved to a non-volatile memory at reasonable time intervals to survive a reboot or power outage.

Every time a device (re)joins the homenet or detects the change of an effective trust verdict for any certificate it will synchronize its cache and store the new effective verdict overwriting any previously cached verdicts. Configured verdicts are stored in the cache as their respective cached counterparts, Neutral verdicts are never stored.

6.3.3. Announcement of Verdicts

A device always announces any configured trust verdicts it has established by itself. It also announces cached trust verdicts it has stored in its trust cache if one of the following conditions applies:

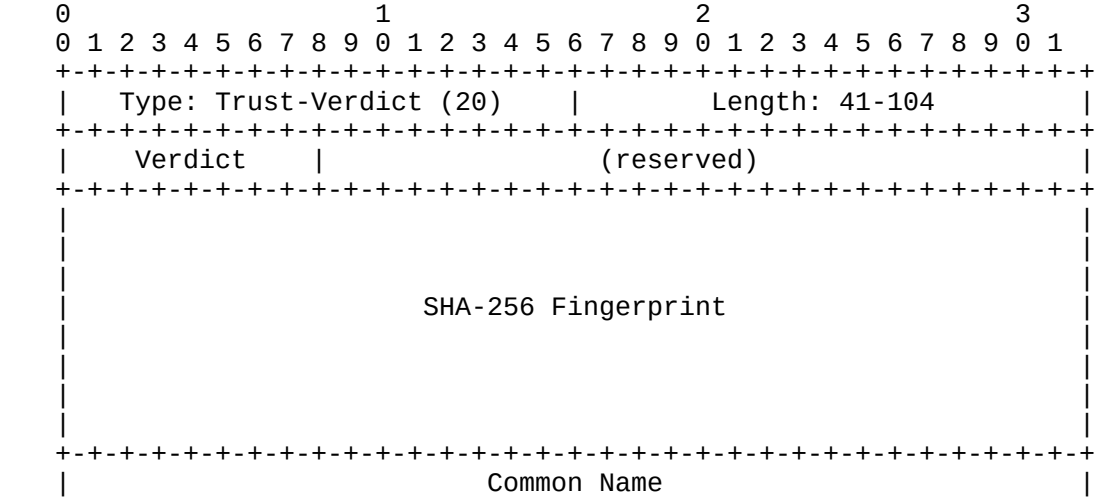
The stored verdict is Cached Trust and the current effective verdict is Neutral or does not exist.

The stored verdict is Cached Distrust and the current effective verdict is Cached Trust.

A device rechecks these conditions whenever it detects changes of announced trust verdicts anywhere in the network.

Upon encountering a device with a hierarchy of certificates for which there is no effective verdict a router announces Neutral verdicts for all certificates found in the hierarchy until an effective verdict different from Neutral can be found for any of the certificates or a reasonable amount of time (10 minutes is suggested) with no reaction and no further connection attempts has passed. Such verdicts SHOULD also be limited in rate and number to prevent denial-of-service attacks.

Trust verdicts are announced using Trust-Verdict TLVs:



Verdict represents the numerical index of the verdict.

(reserved) is reserved for future additions and MUST be set to 0 when creating TLVs and ignored when parsing them.

SHA-256 [RFC6234] Fingerprint contains the fingerprint of the certificate.

Common Name contains the variable-length (1-64 bytes) common name of the certificate.

6.3.4. Bootstrap Ceremonies

The following methods are defined to establish trust relationships between HNCP-routers and router certificates. Trust establishment is a two-way process in which the existing homenet must trust the newly added device and the newly added device must trust at least one of its neighboring routers. It is therefore necessary that both the newly added device and an already trusted device perform such a

ceremony to successfully introduce a device into a homenet. In all cases an administrator MUST be provided with external means to identify the device belonging to a certificate based on its fingerprint and a meaningful common name.

6.3.4.1. Trust by Identification

A device implementing certificate-based trust MUST provide an interface to retrieve the current set of effective trust verdicts, fingerprints and names of all certificates currently known and set configured trust verdicts to be announced. Alternatively it MAY provide a companion HNCP-device or application with these capabilities with which it has a pre-established trust relationship.

6.3.4.2. Preconfigured Trust

A device MAY be preconfigured to trust a certain set of device or CA certificates. However such trust relationships MUST NOT result in unwanted or unrelated trust for devices not intended to be run inside the same network (e.g. all other devices of that manufacturer).

6.3.4.3. Trust on Button Press

A device MAY provide a physical or virtual interface to put one or more of its internal network interfaces temporarily into a mode in which it trusts the certificate of the first HNCP-device it can successfully establish a connection with.

6.3.4.4. Trust on First Use

A device which is not associated with any other homenet-router MAY trust the certificate of the first HNCP-device it can successfully establish a connection with. This method MUST NOT be used when the device has already associated with any other HNCP-router.

7. Other homenet protocols

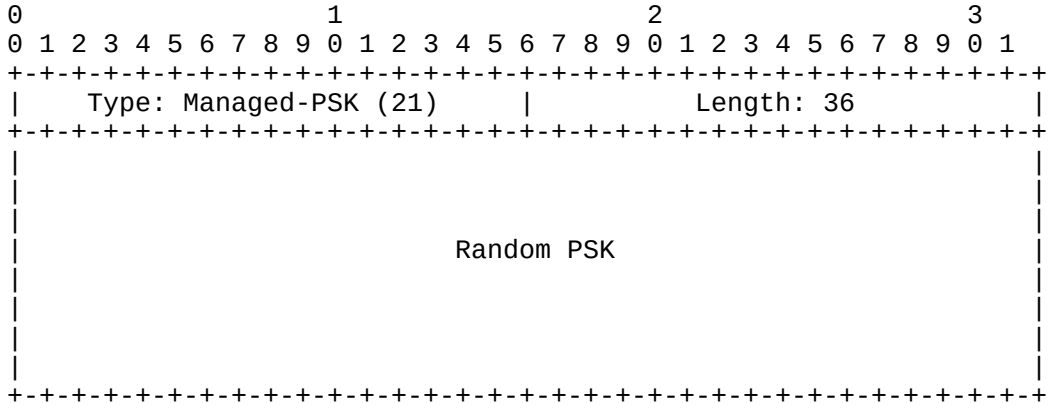
An IGP is usually run alongside HNCP in a homenet therefore the individual security aspects of the respective protocols must be considered. It can however be summarized that current candidate protocols (namely Babel, OSPFv3, RIP and IS-IS) provide - to a certain extent - similar security mechanisms. All mentioned protocols do not support encryption and only support authentication based on pre-shared keys natively. This influences the effectiveness of any encryption-based security mechanism deployed by HNCP as homenet routing information is usually not confidential.

As a PSK is required to authenticate IGP-traffic and potential other protocols, HNCP is used to create and manage it. The key length is defined to be 32 Bytes to be reasonably secure. The following rules determine how a key is managed and used:

If no Managed-PSK-TLV is currently being announced, an HNCP-router creates one with a random key and adds it to its node-data.

In case multiple routers announce such a TLV at the same time, all but the one with the highest router-ID stop advertising it and adopt the remaining one.

The router currently advertising the Managed-PSK-TLV must generate and advertise a new random one whenever the HNCP security mechanism stops trusting one or more trusted devices - i.e. HNCP is secured with a PSK itself and it was changed or a certificate has changed from trusted to distrusted.



PSKs for individual protocols are derived from the random PSK through the use of HMAC-SHA256 [RFC6234] with a pre-defined per-protocol HMAC-key in ASCII-format. The following HMAC-keys are currently defined to derive PSKs for the respective protocols:

"ROUTING": to be used for IGPs. If a Random PSK exists then the derived PSK MUST be used to secure the chosen IGP.

8. Security Considerations

8.1. Revocation of Trust

Revoking trust in a protocol intended for bootstrapping is non-trivial, since neither an accurate clock nor network connectivity to

retrieve authenticated revocation information can be assumed in all situations.

The Certificate-based trust consensus mechanism defined in this document allows for a consenting revocation, however in case of a compromised device the trust cache may be poisoned before the actual revocation happens allowing the distrusted device to rejoin the network using a different identity. Stopping such an attack might require physical intervention and flushing of the trust caches. However such an attack is often times more easily detectable than threats discussed earlier in this document such as a silent manipulation of routing information and related man-in-the-middle attacks.

9. IANA Considerations

IANA should add HNCP TLV types with the following contents:

20: Trust-Verdict

21: Managed-PSK

10. References

10.1. Normative references

[I-D.ietf-homenet-hncp]
Stenberg, M. and S. Barth, "Home Networking Control Protocol", draft-ietf-homenet-hncp-01 (work in progress), June 2014.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

10.2. Informative references

[RFC6234] Eastlake, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

Appendix A. Draft source

As usual, this draft is available at https://github.com/fingon/ietf-drafts/ in source format (with nice Makefile too). Feel free to send comments and/or pull requests if and when you have changes to it!

Appendix B. Acknowledgements

Thanks to Markus Stenberg, Pierre Pfister and Mark Baugher for their contributions to the draft and Xavier Bonnetain for ideas on a web of trust and PSK-management in I-D.bonnetain-hncp-security-00.

Author’s Address

Steven Barth

Email: cyrus@openwrt.org

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 15, 2015

S. Cheshire
Apple Inc.

November 11, 2014

Special Use Top Level Domain "home"
draft-cheshire-homenet-dot-home-01

Abstract

This document specifies usage of the top-level domain "home", for names that are meaningful and resolvable within some scope smaller than the entire global Internet, but larger than the single link supported by Multicast DNS.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 15, 2015.

Copyright Notice

1. Introduction

Globally unique domain names are available to individuals and organizations for a modest annual fee. However, there are situations where a globally unique domain name is not available, or has not yet been configured, and in these situations it is still desirable to be able to use DNS host names [RFC1034] [RFC1035], DNS-Based Service Discovery [RFC6763], and other facilites built on top of DNS.

In the absence of available globally unique domain names, Multicast DNS [RFC6762] makes it possible to use DNS facilities with names that are unique within the local link, using the "local" top-level domain.

This document specifies usage of a similar top-level domain, "home", for names that have scope larger than a single link, but smaller than the entire global Internet.

Author's Note [to be removed when document is published]: The purpose of this draft is not to propose some novel new usage for ".home" names. The purpose is to learn more about the current widespread use of ".home" names, and to document and formalize that usage.

Evidence [ICANN1][ICANN2] indicates that ".home" queries frequently leak out and reach the root name servers. We speculate that this is because of widespread usage of ".home" names in home networks, for example to name a printer "printer.home." When a user takes their laptop to a public Wi-Fi hotspot, attempts by that laptop to contact that printer result in fruitless ".home" queries to the root name servers. It would be beneficial for operators of public Wi-Fi hotspots to recognize and answer such queries locally, thereby reducing unnecessary load on the root name servers, and this document would give those operators the authority to do that. Readers who are aware of other usages of ".home" names, that are not compatible with the rules proposed here, are encouraged to contact the authors with information to help revise and improve this draft.

It is expected that the rules for ".home" names outlined here will also be suitable to meet the needs of the IETF HOMENET Working Group, though that is not the primary goal of this document. The primary goal of this draft is to understand and document the current usage. If the needs of the IETF HOMENET Working Group are not met by this document codifying the current de facto usage, then the Working Group may choose to reserve a different Special Use Domain Name [RFC6761] which does meet their needs. With luck that may not be necessary, and a single document may turn out to be sufficient to serve both purposes. In any case, the HOMENET Working Group is likely to be a good community in which to find knowledge about how ".home" names are currently used.

2. Conventions and Terminology Used in this Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

3. Mechanism

Typical residential home gateways configure their local clients via DHCP [RFC2131]. In addition to the client's IP address, this DHCP configuration information typically also includes other configuration parameters, like the IP address of the recursive (caching) DNS server the client is to use, which is usually the home gateway's own address (the home gateway is also a DNS cache/relay).

For a home network consisting of just a single link (or several physical links bridged together to appear as a single logical link to IP) Multicast DNS [RFC6762], which requires no configuration, is sufficient for client devices to look up the dot-local host names of peers on the same home network, and perform DNS-Based Service Discovery (DNS-SD) [RFC6763] of services offered on that home network.

For a home network consisting of multiple links that are interconnected using IP-layer routing instead of link-layer bridging, link-local Multicast DNS alone is insufficient because link-local Multicast DNS requests, by design, do not cross between links. (This was a deliberate design choice for Multicast DNS, since even on a single link multicast traffic is expensive -- especially on Wi-Fi links -- and multiplying the amount of multicast traffic by flooding it across multiple links would make that problem even worse.) In this environment, unicast DNS requests (as may be facilated by use of ".home" names instead of ".local" names) should be used for cross-link name resolution and service discovery.

For residential home networks, Zero Configuration [ZC] operation is desirable, without requiring any manual configuration from the user. A client device learns about its network environment in a variety of ways. It builds a list of network-recommended DNS search domains using DHCP options 15 (Domain Name option [RFC2132]) and 119 (Domain Search option [RFC3397]). It builds a list of network-recommended DNS-SD browsing domains by sending domain enumeration queries [RFC6763].

For organizations and individuals with a registered globally unique domain name under their control, hosts and services can be given

names within that domain. Client devices can be configured to use that globally unique domain name as their DNS search domain and/or DNS-SD browsing domain [RFC6763]. For example, at IETF meetings the network configures client devices to use "meeting.ietf.org." as their DNS search domain and DNS-SD browsing domain. This domain name is globally unique and under the control of the IETF. It is entered into the DHCP and DNS servers manually by the IETF meeting network administrators, and then communicated automatically via the network to client devices.

When a suitable globally unique domain name is available, as at IETF meetings, manual configuration of that name in a residential home gateway (or equivalent enterprise equipment) is appropriate. The network infrastructure then communicates that information to clients, without any additional manual configuration required on those clients.

However, many residential customers do not have any registered globally unique domain name available. This may be because they don’t want to pay the annual fee, or because they are unaware of the process for obtaining one, or because they are simply uninterested in having their own globally unique domain. This category also includes customers who intend to obtain a globally unique domain, but have not yet done so. For these users, it would be valuable to be able to perform cross-link name resolution and service discovery using unicast DNS without requiring a globally unique domain name.

To facilitate zero configuration operation, residential home gateways should be sold preconfigured with the default unicast domain name "home". This default unicast domain name is not globally unique, since many different residential home gateways will be using the name "home" at the same time, but is sufficient for useful operation within a small collection of links. Such residential home gateways SHOULD offer a configuration option to allow the default (non-unique) unicast domain name to be replaced with a globally unique domain name for cases where the customer has a globally unique domain available and wishes to use it.

This use of the the top-level domain "home" for private local use is not new. Many home gateways have been using the name this way for many years, and it remains in widespread use, as evidenced by the large volume of invalid queries for "home" reaching the root name servers [ICANN1][ICANN2]. The current root server traffic load is due to things like home gateways configuring clients with "home" as a search domain, and then leaking the resulting dot-home queries upstream. In large part what the document proposes is, "stop leaking dot-home queries upstream." This document codifies the existing practice, and provides formal grounds basis for ISPs to legitimately

block such queries in order to reduce unnecessary load on the root name servers.

4. Security Considerations

Users should be aware that names in the "home" domain have only local significance. The name "My-Printer.home" in one location may not reference the same device as "My-Printer.home" in a different location.

5. IANA Considerations

[Once published, this should say] IANA has recorded the top-level domain "home" in the Special-Use Domain Names registry [SUDN].

5.1. Domain Name Reservation Considerations

The top-level domain "home", and any names falling within that domain (e.g., "My-Computer.home.", "My-Printer.home.", "_ipp._tcp.home."), are special [RFC6761] in the following ways:

1. Users may use these names as they would other DNS names, entering them anywhere that they would otherwise enter a conventional DNS name, or a dotted decimal IPv4 address, or a literal IPv6 address.

   Since there is no global authority responsible for assigning dot-home names, devices on different parts of the Internet could be using the same name. Users SHOULD be aware that using a name like "www.home" may not actually connect them to the web site they expected, and could easily connect them to a different web page, or even a fake or spoof of their intended web site, designed to trick them into revealing confidential information. As always with networking, end-to-end cryptographic security can be a useful tool. For example, when connecting with ssh, the ssh host key verification process will inform the user if it detects that the identity of the entity they are communicating with has changed since the last time they connected to that name.

2. Application software may use these names the same way it uses traditional globally unique unicast DNS names, and does not need to recognize these names and treat them specially in order to work correctly. This document specifies the use of the top-level domain "home" in on-the-wire messages. Ideally this would be purely a protocol-level identifier, not seen by end users. However, in some applications domain names are seen by end users,

and in those cases, the protocol-level identifier "home" becomes visible, even for users for whom English is not their preferred language. For this reason, applications MAY choose to use additional UI cues (icon, text color, font, highlighting, etc.) to communicate to the user that this is a special name with special properties. Due to the relative ease of spoofing dot-home names, end-to-end cryptographic security remains important when communicating across a local network, just as it is when communicating across the global Internet.

3. Name resolution APIs and libraries SHOULD NOT recognize these names as special and SHOULD NOT treat them differently. Name resolution APIs SHOULD send queries for these names to their configured recursive/caching DNS server(s).

4. Recursive/caching DNS servers SHOULD recognize these names as special and SHOULD NOT, by default, attempt to look up NS records for them, or otherwise query authoritative DNS servers in an attempt to resolve these names. Instead, recursive/caching DNS servers SHOULD, by default, act as authoritative and generate immediate responses for all such queries. This is to avoid unnecessary load on the root name servers and other name servers.

   The type of response generated depends on the role of the recursive/caching DNS server: (i) Traditional recursive DNS servers (such as those run by ISPs providing service to their customers) SHOULD, by default, generate immediate negative responses for all such queries. (ii) Recursive/caching DNS servers incorporated into residential home gateways of the kind described by this document should act as authoritative for these names and return positive or negative responses as appropriate.

   Recursive/caching DNS servers MAY offer a configuration option to enable upstream resolving of these names, for use in networks where these names are known to be handled by an authoritative DNS server in said private network. This option SHOULD be disabled by default, and SHOULD be enabled only when appropriate, to avoid queries leaking out of the private network and placing unnecessary load on the root name servers.

5. Traditional authoritative DNS servers SHOULD recognize these names as special and SHOULD, by default, generate immediate negative responses for all such queries, unless explicitly configured otherwise by the administrator. As described above, DNS servers incorporated into residential home gateways of the kind described by this document should act as authoritative for these names and return positive or negative responses as appropriate, unless explicitly configured otherwise by the

administrator.

6. DNS server operators SHOULD, if they are using these names, configure their authoritative DNS servers to act as authoritative for these names. In the case of zero-configuration residential home gateways of the kind described by this document, this configuration is implicit in the design of the product, rather than a result of conscious administration by the customer.

7. DNS Registries/Registrars MUST NOT grant requests to register these names in the normal way to any person or entity. These names are reserved for use in private networks and fall outside the set of names available for allocation by registries/ registrars. Attempting to allocate a these name as if it were a normal DNS domain name will probably not work as desired, for reasons 4, 5, and 6 above.

6. Acknowledgments

Thanks to Francisco Arias of ICANN for his review and comments on this draft.

7. References

7.1. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6761] Cheshire, S. and M. Krochmal, "Special-Use Domain Names", RFC 6761, February 2013.

7.2. Informative References

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997.

[RFC3397] Aboba, B. and S. Cheshire, "Dynamic Host Configuration Protocol (DHCP) Domain Search Option", RFC 3397, November 2002.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, February 2013.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, February 2013.

[ICANN1] "New gTLD Collision Risk Mitigation", <https://www.icann.org/en/about/staff/security/ssr/new-gtld-collision-mitigation-05aug13-en.pdf>.

[ICANN2] "New gTLD Collision Occurrence Management", <https://www.icann.org/en/system/files/files/resolutions-new-gtld-annex-1-07oct13-en.pdf>.

[SUDN] "Special-Use Domain Names Registry", <http://www.iana.org/assignments/special-use-domain-names/>.

[ZC] Cheshire, S. and D. Steinberg, "Zero Configuration Networking: The Definitive Guide", O'Reilly Media, Inc. , ISBN 0-596-10100-7, December 2005.

Author's Address

Stuart Cheshire
Apple Inc.
1 Infinite Loop
Cupertino, California 95014
USA

Phone: +1 408 974 3207
Email: cheshire@apple.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: February 13, 2015

C. Donley
M. Kloberdans
CableLabs
J. Brzozowski
Comcast
C. Grundemann
ISOC

August 12, 2014

Customer Edge Router Identification Option
draft-donley-dhc-cer-id-option-04

Abstract

Addressing mechanisms supporting DHCPv6 Prefix Delegation in home networks such as those described in CableLabs' eRouter specification and the HIPnet Internet-Draft require identification of the customer edge router (CER) as the demarcation between the customer network and the service provider network. This document reserves a DHCPv6 option to identify the CER.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 13, 2015.

Copyright Notice

Table of Contents

1. Introduction

Some addressing mechanisms supporting DHCPv6 Prefix Delegation in home networks such as those described in [I-D.grundemann-homenet-hipnet] and [EROUTER] require identification of the customer edge router as the demarcation between the customer network and the service provider network. For prefix delegation purposes, it is desirable for other routers within the home to know which device is the CER so that the customer home network only requests a single prefix from the ISP DHCPv6 server, and efficiently distributes this prefix within the home. CER-ID is a 128-bit string that optionally represents an IPV6 address, or another arbitrary number. The CER-ID maybe treated as a hint to be used with border detection methods. This document reserves a DHCPv6 option to be used to identify the CER.

1.1. Requirements Language

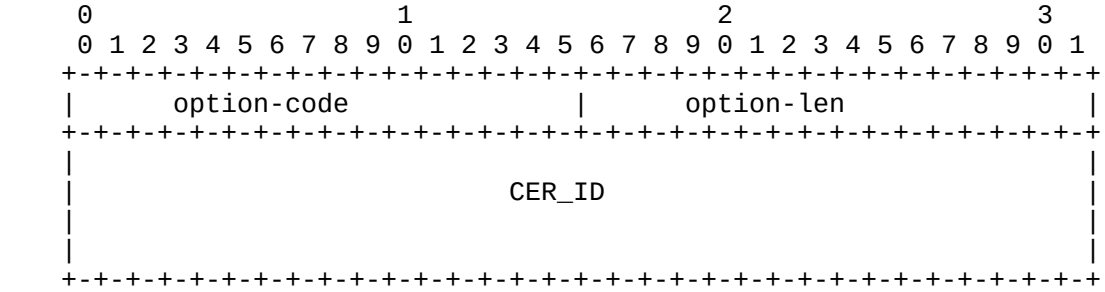
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. CER Identification Option

A Customer Edge Router (CER) sets the CER_ID to the IPv6 address of its LAN interface. If it has more than one LAN IPv6 address, it selects one of its LAN or loopback IPv6 addresses to be used in the CER_ID. An ISP server does not respond with the CER_ID or sets the

CER_ID to ::. Such a response or lack of response indicates to the DHCPv6 client that it is the CER.

The format of the CER Identification option is:



option-code OPTION_CER_ID (TBD).
option-len 36
CER_ID value IPv6 address of CER or ::

Figure 1.

A DHCPv6 client SHOULD include the CER Identification option code in an Option Request option [RFC3315] in its DHCP Solicit messages.

The DHCPv6 server MAY include the CER Identification option in any response it sends to a client that has included the CER Identification option code in an Option Request option. The CER Identification option is sent in the main body of the message to client, not as a sub-option in, e.g., an IA_NA, IA_TA [RFC3315]option.

When sending the CER Identification option, the DHCPv6 server MUST set the CER_ID value to either one of its IPv6 addresses, another identifier, or ::. If a device does not receive the CER Identification Option or receives a CER ID of :: from the DHCPv6 server, it MUST include one of its Globally Unique IPv6 addresses (unless another identifier is used), in the CER_ID value in response to DHCPv6 messages received by its DHCPv6 server that contains the CER Identification option code in an Option Request option. If the device has only one LAN interface, it SHOULD use its LAN IPv6 address as the CER_ID value. If the device has more than one LAN interface, it SHOULD use the lowest Globally Unique address not assigned to its WAN interface.

3. CER-ID Compatibility

CER-ID explicitly indicates that a gateway is, or is not, the demarcation point between public and private networks by containing a reachable IPv6 address, other identifier or a double colon '::' (double colon indicates that the CER-ID sender is NOT the edge router), and as a compliment, can be applied to various border definitions and detection methods such as:

- I.D. Draft-IETF-Homenet-Arch-16 [I-D.ietf-homenet-arch]
- I.D. Draft-Grundemann-homenet-HIPnet-01 [I-D.grundemann-homenet-hipnet]
- I.D. Draft-IETF-Kline-Homenet-Default-Perimeter-01 [I-D.kline-default-perimeter]
- Others, including manual configuration

4. IANA Considerations

IANA is requested to assign an option code from the "DHCP Option Codes" Registry for OPTION_CER_ID. IANA is also requested to maintain a list of authentication options.

5. Security Considerations

The security of a home network is an important consideration. Both the HIPNet [I-D.grundemann-homenet-hipnet] and Homenet [I-D.ietf-homenet-arch] approaches change the operational model of the home network vs. today's IPv4-only paradigm. Specifically, these networks eliminate NAT inside the home network (and only enable it for IPv4 at the edge router, if required), support global addressability of devices, and thus need to consider firewall and/or filter support in various home routers. As the security profile of these home routers can shift based on their position in the network (e.g., edge vs. internal), security can be severely compromised if routers misidentify their border and mistakenly reduce or eliminate firewall rules. If the CER-ID option is used as part of the border detection algorithm, it becomes a natural, but not the only place to enact firewall, NAT, Prefix Delegation and other functions in the home network.

6. Acknowledgements

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

7.2. Informative References

[EROUTER] CableLabs, "CableLabs IPv4 and IPv6 eRouter Specification (CM-SP-eRouter-I12-131120)", April 2014.

[I-D.grundemann-homenet-hipnet] Grundemann, C., Donley, C., Brzozowski, J., Howard, L., and V. Kuarsingh, "A Near Term Solution for Home IP Networking (HIPnet)", draft-grundemann-homenet-hipnet-01 (work in progress), February 2013.

[I-D.ietf-homenet-arch] Chown, T., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", draft-ietf-homenet-arch-16 (work in progress), June 2014.

[I-D.kline-default-perimeter] Kline, E., "Default Border Definition", draft-kline-default-perimeter-01 (work in progress), November 2012.

Authors' Addresses

Chris Donley
CableLabs
858 Coal Creek Cir.
Louisville, CO 80027
US

Email: c.donley@cablelabs.com

Michael Kloberdans
CableLabs
858 Coal Creek Cir
Louisville, CO 80027
US

Email: m.kloberdans@cablelabs.com

John Brzozowski
Comcast
1306 Goshen Parkway
West Chester, PA 19380
US

Email: john_brzozowski@cable.comcast.com

Chris Grundemann
ISOC
Denver CO

Email: cgrundemann@gmail.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 27, 2015

P. Pfister
B. Paterson
Cisco Systems
J. Arkko
Ericsson

October 24, 2014

Prefix and Address Assignment in a Home Network
draft-ietf-homenet-prefix-assignment-01

Abstract

This memo describes a home network prefix and address assignment algorithm running on top of any 'flooding protocol' that fulfills the specified requirements. It is expected that home border routers are allocated one or multiple IPv6 prefixes through DHCPv6 Prefix Delegation (PD) or that prefixes are made available through other means. An IPv4 address can also be assigned and private addresses be used with NAT to provide IPv4 connectivity. In both cases, provided prefixes need to be efficiently divided among the multiple links, and routers need to obtain addresses. This document describes a distributed algorithm for IPv4 and IPv6 prefixes division, assignment and router's address assignment, and specifies how hosts can be given addresses and configuration options using DHCP or SLAAC.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 27, 2015.

Table of Contents

1. Introduction

This memo describes a fully distributed prefix and address assignment algorithm for home networks, running on top of any 'flooding protocol' that fulfills the specified requirements. It is expected that home border routers are allocated one or multiple IPv6 prefixes through DHCPv6 Prefix Delegation (PD) [RFC3633] or that prefixes are made available through other means. When an IPv4 address is assigned, a home private IPv4 prefix may be used with NAT to provide IPv4 connectivity to the whole home, as well as Unique Local Address prefixes [RFC4193] may be used in order to provide internal connectivity whenever global IPv6 connectivity is not available.

Obtained IPv6 or IPv4 prefixes need to be efficiently divided among the multiple links. For the purposes of this document, we refer to this process as prefix assignment. This memo describes an algorithm for such prefix division, assignment and router's address assignment, as well as the way hosts can be given addresses and configuration options using DHCPv4 [RFC2131], DHCPv6 [RFC3315] or SLAAC [RFC4862]. In the rest of this document DHCP refers to both DHCPv4 and DHCPv6.

Although this document recommends the use of 64 bits long prefixes, the algorithm do not require routers to assign prefixes of particular lengths. When a delegated prefix is too small considered the number of links in the home network, higher priority links may be privileged

or smaller prefixes can be assigned in order to avoid prefix scarcity.

The rest of this memo is organized as follows. Section 2 defines the usual keywords, Section 3 outlines the algorithms functioning and features, Section 4 describes how a home router behaves when running the prefix and address assignment algorithm. Requirements for the underlying flooding protocol are detailed in Section 5. The prefix assignment algorithm is detailed in Section 6 and Section 7 focuses on the address assignment algorithm. Section 8 explains the hysteresis principles applied to both prefix and address assignments, Section 9 specifies the procedures for automatic generation of ULA and IPv4 prefixes, Section 10 explains what administrative interfaces are useful for advanced users that wish to manually interact with the mechanisms, Section 11 gives values for the constants used in this document, Section 12 discusses the security aspects and finally, Appendix A provides implementation guidelines for the optional scarcity avoidance mechanism.

The Prefix Assignment Algorithm was first detailed in [I-D.arkko-homenet-prefix-assignment]. This document is a continuation and generalization of that draft to any underlying flooding protocol. It also adds support for arbitrary prefix length, IPv4, scarcity avoidance mechanism or manual configuration.

2. Requirements language

In this document, the key words "MAY", "MUST, "MUST NOT", "OPTIONAL", "RECOMMENDED", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [RFC2119].

3. Prefix and Address Assignment Algorithms' Outline

Given one or multiple prefixes for the entire network, each prefix is subdivided by the prefix assignment algorithm so that every link is given one assignment per available prefix. Assignments are advertised through the whole network using the underlying flooding protocol, collisions are detected and valid assignments are chosen and applied on every link. Once a prefix is applied, hosts and routers may be given addresses. In summary, the algorithm works in four steps:

1. The home is given IPv6 or IPv4 prefixes called Delegated Prefixes (DPs).

2. Each link is provided an Assigned Prefix (AP) from each available Delegated Prefix.

3. Routers internally check for AP's validity and select Chosen Prefixes (CPs).

4. Once a link is given an assignment, routers may get addresses from specified address pools and hosts may be configured using SLAAC or by the per-link elected DHCP server.

This algorithm, which intends to fulfill requirements specified in [I-D.ietf-homenet-arch], has the following features:

- Each delegated prefix is effectively divided so that each link is assigned a reasonable part. If the delegated prefix is too small given the size of the network, prefixes of arbitrary lengths may be used.

- The algorithm is completely distributed. Routers may join or leave and DPs may be added or removed at any time.

- IPv4 connectivity is provided when a home router acquires an IPv4 address and default route from an external source. In this case a private IPv4 delegated prefix is generated and prefixes are assigned similarly to IPv6.

- The network may spontaneously generate and use a Unique Local Address (ULA) prefix.

- Assignments are stable across reboots and some network changes (e.g. adding or removing routers).

- DHCP options like DNS servers, prefix colors [I-D.bhandari-dhc-class-based-prefix], or any upcoming options may be attached to each prefix and may be relayed down to the host when it is given addresses.

- The user can manually assign prefixes to links. Such assignments will take precedence over automatically assigned prefixes.

- Assignments and interfaces can be given priorities. When a delegated prefix is too small, such values may be used to prioritize prefix assignment to certain links.

4. Router Behavior

All home routers participating in the prefix assignment algorithm MUST fulfill the requirements defined in this document and use a common flooding protocol and routing protocol. Classic CPE routers [RFC7084] are supported as downstream routers and dowstream DHCPv6-PD enabled routers are supported as both downstream and uplink routers,

but problems may occur when such router is connected to the home network on both WAN and LAN side. In the later case, finer external interface detection algorithm or static configuration can be used to solve the issue, but these are out of the scope of this document.

4.1. Data structures

Each router MUST maintain a list of all the Delegated Prefixes. These prefixes may be locally generated, as described in Section 4.3, or come from other routers as described in Section 5.4.

Each router MUST maintain a list of all the Assigned Prefixes advertised by other routers. Each AP is learnt through the mechanisms described in Section 5.3 and is defined as a tuple of:

Prefix: The assigned prefix.

Router ID: The identifier of the advertising router.

Link ID: If the assignment is made on a connected link, an interface identifier of the interface connected to that link.

Authoritative bit: A boolean that tells whether the assignment comes from a network authority (DHCPv6 PD, manual configuration, etc...).

Assignment’s Priority: A value between PRIORITY_MIN and PRIORITY_MAX, specifying the assignment’s priority.

The AP list is the result of the information provided by the flooding protocol, as specified in Section 5.3.

The router MUST maintain a list of all prefixes currently chosen to be applied on connected links. They are Chosen Prefixes (CPs) and described by a tuple of:

Prefix: The assigned prefix.

Link ID: An interface identifier of the interface connected to the link on which the assignment is made.

Authoritative bit: A boolean that tells whether the assignment comes from a network authority (DHCPv6 PD, manual configuration, etc...).

Assignment’s Priority: A value between PRIORITY_MIN and PRIORITY_MAX, specifying the assignment’s priority.

Advertised: Whether that assignment is being advertised by the flooding protocol (see Section 5.3).

Applied: Whether that assignment is applied on link's configuration (see Section 4.6).

Chosen Prefixes that are marked as 'Advertised' are distributed to other routers using the flooding protocol and are therefore considered as Assigned Prefixes by other routers. The goal of the Prefix Assignment Algorithm is to ensure that all routers have a consistent view of Assigned Prefixes on each link.

The router MUST maintain a database of its own address assignments, and address assignments made by other routers on connected links as learnt through means described in Section 5.5.

4.2. Routers' Interfaces

Each interface MUST either be considered as internal or external. Prefixes and addresses are only assigned to internal interfaces. The criteria to make this distinction are out of the scope of this document.

If an internal interface becomes external, all prefixes and addresses assigned on the considered interface MUST be deleted and no longer announced, and the prefix assignment algorithm MUST be run.

If an external interface becomes internal, the prefix assignment algorithm MUST be run (see Section 6.1).

Whenever two or more interfaces are connected to the same link, all but one of them SHOULD be ignored by the prefix assignment algorithm. A mechanism to detect such situation SHOULD be provided by the flooding algorithm.

4.3. Obtaining a Delegated Prefix

Delegated Prefixes (Of any kind: Global, ULA, IPv4...) can be obtained or generated through different means:

- It can be delegated by a service provider (DHCPv6 PD, 6rd [RFC5969], etc..).
- It can be provisioned by an administrative authority (user configuration, netconf [RFC6241], etc...).
- A ULA prefix may be spontaneously generated as defined in Section 9.1.

- An IPv4 private prefix may be spontaneously generated as defined in Section 9.2.

DHCP options MAY be attached to a delegated prefix by the router that either generated the prefix or received it through DHCPv6 PD. IPv6 delegated prefix options MUST be encoded as DHCPv6 options. IPv4 delegated prefix options MUST be encoded as DHCPv4 options.

As DHCP options are numerous and new ones may be defined, specifying routers' behavior regarding each option is out of the scope of this document. In order to avoid misconfiguration, routers must follow the two following general rules:

- A router MUST NOT advertise a prefix obtained through DHCPv6 PD if it doesn't understand the all of the provided options.

- A router MUST NOT make or accept any assignment associated to a delegated prefix if it doesn't understand the all of the DHCP options advertised with the delegated prefix.

The mif working group may provide useful inputs concerning the way the home network should handle different prefixes associated with heterogeneous uplinks.

4.4. Network Leader

A router considers itself as the Network Leader if and only if its router ID is greater than all other router IDs in received Prefix Assignments and Delegated Prefixes.

4.5. Designated Router

On a link where custom host configuration must be provided, or whenever SLAAC cannot be used, a DHCP server must be elected. That router is called designated router and is dynamically chosen by the prefix assignment algorithm.

A router MUST consider itself designated router on a given link if either one of the following conditions holds:

- The link's Assigned Prefixes list is empty. i.e. no other router is advertising assignments on the considered link. And, if such information is provided by the flooding protocol, the router has the highest id on the link.

- Considering all APs and advertised CPs on the given link, the router is advertising the one with:

1. The lowest authoritative bit.

2. In case of tie, the lowest priority.

3. In case of tie, the highest router ID.

Note: That particular order (inverted compared to assignments' priority) is motivated by the few cases where a router may override an existing assignment by advertising an assignment of higher priority. In such a case, the designated router should remain the same.

Example: A new router is powered on and connected to another router that was already there (doing DHCP). It sees the assigned prefix for their common link, but also has, in its own configuration, an authoritative assignment for the link. It starts advertising the authoritative assignment, which causes the second router to remove its previous assignment. Thanks to the inverted order, the DHCP server will remain the same.

4.5.1. Sending Router Advertisement

On a given link, the designated router MUST send router advertisements (RAs) including Prefix Information Options for all the Chosen Prefixes associated to that link. SLAAC SHOULD be enabled when possible, unless the configuration states otherwise. Prefixes valid and preferred lifetimes MUST be set to values lower or equal to the associated Delegated Prefix's valid and preferred lifetimes.

Whenever an IPv6 default route is present in the RIB, the designated router MUST advertise itself as default router by specifying a strictly positive valid lifetime. Whenever the last default route is removed, the designated router MUST send an RA with the valid and preferred lifetimes set to zero.

The designated router MUST advertise itself as a router for all IPv6 delegated prefixes using Route Information Options [RFC4191], independently of whether there is a default route or not.

4.5.2. DHCP Server Operations

On a given link, whenever SLAAC can't be used for all assignments, or DHCP configuration options must be provided to hosts, the designated router MUST act as a DHCP server and serve addresses on the given link. A router MUST stop behaving as a DHCP server whenever it is not the link's designated router anymore.

Routers's addresses pool, specified in Section 7, MUST be excluded from DHCP hosts pools.

The valid and preferred lifetimes MUST be set to values lower or equal to the associated Delegated Prefix's valid and preferred lifetimes.

4.6. Applying an Assignment on an Interface

Once a Chosen Prefix is created, a router first waits some time in order to detect possible collisions (Section 8). Afterwards and if no collision is detected, the prefix is applied as follows:

- The router updates its interface configuration so that the prefix is assigned to the considered link.
- The router updates the routing protocol configuration so that it starts advertising the prefix. Depending on the implementation, this step may not be needed as the routing protocol directly gets its configuration information from the interfaces configuration.
- If necessary, the router starts selecting an address for itself as defined in Section 7.
- If the router is the designated router on the considered link, it starts sending the Prefix Information Option with the considered prefix, as specified in Section 4.5.1.
- If the router is the designated router on the considered link and if the prefix requires DHCP configuration, it starts behaving as a DHCP server, as defined in Section 4.5.2, for the considered assigned prefix.

When a prefix assignment is removed, the previous steps MUST be undone. The router MUST also deprecate the prefix, if it had been advertised in Router Advertisements on an interface. The prefix is deprecated by sending Router Advertisements with the PIO's preferred lifetime set to 0 [RFC4861]. Hosts that support DHCP reconfigure extension ([RFC3203], [RFC3315]) and that have been given leases MUST be reconfigured as well.

4.7. DNS Support

DHCP options attached to each delegated prefixes and propagated through the flooding protocol SHOULD contain the DHCP DNS options provided by the ISP (when provided).

Whenever the router knows which DNS server to use, or is acting as a DNS relay, it SHOULD include DNS DHCP options ([RFC3646]) within host's configuration messages and include the Router Advertisement DNS options ([RFC6106]) when sending RAs.

DNS server selection in multi-homed networks is a complex issue that this document doesn't intend to solve. One should look at IETF's mif working-group documents in order to obtain guidelines concerning DNS server selection. It is RECOMMENDED that designated routers turns on a local DNS relay that fetches information from provided DNS servers.

5. Flooding Protocol Requirements

In this document, the Flooding Protocol (FP) refers to a protocol enabling information propagation to the whole network. It was not specified in order to allow the working group to independently decide which routing protocol, configuration protocol, and prefix assignment method to use within the home network. Routing protocol, like OSPFv3 [RFC5340] (With its autoconf extension [I-D.ietf-ospf-ospfv3-autoconfig]) or IS-IS [RFC5308], could be extended in order to fulfill the requirements. An independent protocol, for instance HNCP [I-D.ietf-homenet-hncp], could be used as well.

The specified algorithm can use any protocol that fulfills the requirements specified in this section.

5.1. Router ID

The FP MUST provide a router ID. ID collisions within the network MUST be rare and any conflicts MUST be resolved by the flooding protocol. When the router ID is changed, the FP MUST immediately provide the new ID to the Prefix Assignment Algorithm, which will in turn be run again, without requiring the current state to be flushed.

In the absence of collisions, the router ID MUST NOT be changed, and it SHOULD be stable across reboots, power cycling and router software updates.

5.2. Propagation Delay

The FP MUST provide an approximate upper bound of the time it takes for an update to be propagated to the whole network. This value is referred to as the FLOODING_DELAY. The algorithm ensures that, as long as the upper bound is respected, two identical prefixes will never be applied to different links, and two different prefixes will never be applied to the same link. The algorithm and the network will recover when the upper bound is exceeded, but collisions may

appear in the routing protocol and errors may be propagated to upper layers.

If the FP supports link-local flooding, which is used for router's address assignments, it SHOULD provide an approximate upper bound of the time it takes for an update to be propagated to a single link. This value is referred to as the FLOODING_DELAY_LL. If link-local flooding is not available, or the value is not provided, the assignment algorithm MUST use the FLOODING_DELAY value instead.

5.3. Flooding Assigned Prefixes

The FP MUST provide a way to flood Chosen Prefixes marked as advertised and retrieve prefixes assigned by other routers (APs). Retrieved APs MUST contain all the information specified in Section 4.1.

5.4. Flooding Delegated Prefixes

The FP must provide a way to flood Delegated Prefixes and retrieve prefixes delegated to other routers. Retrieved entries must contain the following information.

Prefix: The delegated prefix.

Router ID: The router ID of the router that is advertising the delegated prefix.

Valid until: A time value, in absolute local time, specifying the prefix validity time.

Preferred until: A time value, in absolute local time, specifying the prefix preferred time.

DHCP information: DHCP options attached to the delegated prefix.

The FP MUST make sure time values are consistent throughout the network (i.e. differences are small compared to Delegated Prefixes lifetimes). If no time synchronization protocol is used, the FP MUST keep track of prefix age across the network and within its database.

5.5. Flooding Routers' Address Assignments

Routers addresses are dynamically allocated, picked from a defined pool, and collisions must be detected using the FP. The FP MUST provide a way to flood routers' addresses. The flooding scope of those values SHOULD be link-local, but as addresses are unique within the home network, this is not mandatory. For each address

assignment, the FP SHOULD provide the identifier of the interface connected to the link the address assignment was advertised on.

6. Prefix Assignment Algorithm

The Prefix Assignment Algorithm is a distributed algorithm that assigns one prefix from each available Delegated Prefix on every link that is considered to be internal by at least one connected router. The algorithm itself does not distinguish between global IPv6, ULA or IPv4 prefixes. IPv4 prefixes are encoded as their IPv4-mapped IPv6 form, as defined in [RFC4291] (i.e. ::ffff:A.B.C.D/X with X >= 96).

When the Prefix Assignment Algorithm is executed, combinations of Delegated Prefixes and internal interfaces are being considered. For the purpose of this discussion, the Delegated Prefix will be referred to as the current Delegated Prefix, and the interface will be referred to as the current Interface. If a delegated prefix is included inside another delegated prefix, it is ignored. This rule intends to ignore prefixes delegated from non-Homenet routers that previously obtained their larger prefix from one of Homenet’s routers.

The algorithm is specified here for the sake of clarity. It can be optimized in some cases. For instance Prefix Assignment deletion might not need to trigger algorithm’s execution if all internal interfaces already have assignements associated to the same Delegated Prefix. Similarly, when an ignored Delegated Prefix is deleted, it is not necessary to run the algorithm. An implementation may work differently than specified here as long as the resulting behavior is identical to the behavior a router implementing this exact algorithm would have.

6.1. When to execute the Prefix Assignment Algorithm

The algorithm MUST be run whenever one of the following event occurs:

- A Delegated Prefix is created or deleted (A DP must be deleted when its lifetime is exceeded).
- A Prefix Assignment is created, deleted or modified.
- The router ID is modified.
- An external link becomes internal, or an internal link becomes external.

It is not required that the algorithm is synchronously run each time such an event occurs. But the delay between the event and the algorithm execution MUST be small compared to FLOODING_DELAY.

6.2. Assignment Precedence

An assignment is said to take precedence over another assignment when:

- The authoritative bit value is higher.
- In case of tie, the priority value is higher.
- In case of tie, the advertising router's ID is higher.

6.3. Testing Assignment's validity

An Assigned Prefix or a Chosen Prefix is said to be valid if all the following conditions are met:

1. Its prefix is included in an advertised Delegated Prefix.
2. The prefix is not included or does not include any other Assigned Prefix with a higher precedence.
3. No other assignment which prefix is included in the same Delegated Prefix, and with a higher precedence, is being advertised on the same link.

6.4. Testing Assignment's availability

A prefix is said to be available if it does not overlap with any other assignment by any other router in the network.

6.5. Accepting an Assigned Prefix

An AP is said to be accepted when the AP is currently being advertised by a different router on a directly connected link, and will be used by the accepting router as a new Chosen Prefix. When a router accepts a neighbor's assignment, it starts a timer as specified in Section 8. A new CP is created from the AP, with:

- The same prefix.
- The same link ID.
- The authoritative bit set to false.

- The same priority.
- The advertised bit value set as specified by the algorithm.
- The applied bit is unset. It is set when the timer elapsed if the entry still exists.

6.6. Making a New Assignment

In situations where a router can make an assignment (see Section 6.9), the following rules are used in the following order:

1. If the configuration specifies a custom behavior (e.g. always ignore/accept a particular delegated prefix), use the configuration entry.
2. If the Delegated Prefix Preferred Lifetime is strictly greater than zero, an assignment MUST be made.
3. If no other prefix has a non-zero Preferred Lifetime, and no assignment is made on the link, an assignment SHOULD be made.
4. Otherwise, a new assignment SHOULD NOT be made.

When the algorithm decides to make a new assignment, it first needs to specify the desired size of the assigned prefix. Although this algorithm intends to remain generic, the use of 64 bits long prefixes is RECOMMENDED (See [I-D.ietf-6man-why64]). The following table MAY be used as default values, where X is the length of the delegated prefix.

If X <= 64: Prefix length = 64

If X >= 64 and X < 104: Prefix length = X + 16 (up to 2^16 links)

If X >= 104 and X < 112: Prefix length = 120 (2^8 addresses per link and more than 2^8 links)

If X >= 112 and X <= 128: Prefix length = 120 + (X - 112)/2 (Link Vs Addresses tradeoff)

When the algorithm decides to make a new assignment, it SHOULD first checks its stable storage for an available assignment that was previously applied on the current interface and is part of the current delegated prefix. If no available assignment can be found that way, the new prefix MUST be randomly selected among a subset of available prefixes (if possible, large enough to avoid collisions).

Hardware specific identifiers may be used to seed a pseudo-random generator.

If no available prefix is found, the assignment fails.

The algorithm leaves much room for implementation specific policies. For instance, static prefixes may be configured as specified in Section 10. If implemented, the router MAY also decide to execute the Prefix Scarcity Avoidance mechanisms, as proposed in Appendix A.

If an available prefix is found, a new assignment is made and a new Chosen Prefix entry is created.

- The prefix value is set to the chosen prefix.
- The link ID is the ID of the link on which the assignment is made.
- The authoritative bit is set to false.
- The priority is set to a value between PRIORITY_AUTO_MIN and PRIORITY_AUTO_MAX (Section 6.8).
- The advertised bit is set.
- The applied bit is unset. It is set when the timer elapsed if the entry still exists.

A new assignment is always marked as advertised when created and therefore immediately provided to the flooding protocol.

6.7. Using Authoritative Prefix Assignments

When some authority (Delegating router, system admin, etc...) wants to manually enforce some behavior, it may ask some router to make an Authoritative Prefix Assignment. Such assignments have their Authoritative bit set, SHALL NOT be overridden, and will appear in other router’s database as Assigned Prefixes with the Authoritative bit set.

There are two kinds of Authoritative Prefix Assignments.

- When an authority wants to assign some particular prefix to some interface, an Authoritative Prefix Assignment MAY be created and consists in a Chosen Prefix which have its Authoritative bit set and which is advertised. Just like normal assignments, it MUST NOT be applied before the delay specified in Section 8 elapsed.

- When an authority wants to prevent some prefix from being used, an Authoritative Assignment MAY be advertised. Such assignments MUST NOT be applied and MUST be advertised through the flooding protocol as assigned to either no-interface, or a fake interface (Depending on the flooding protocol's capabilities).

When a delegated prefix is obtained through DHCPv6 PD with a non-empty excluded prefix, as specified in [RFC6603], an Authoritative Prefix Assignment MUST be created with the excluded prefix.

Note: If the router doesn't understand the excluded prefix DHCPv6 option, the delegated prefix is ignored, as specified in Section 4.3.

6.8. Choosing the Assignment's Priority

When either a new Prefix Assignment is made, or an Authoritative Prefix Assignment is created, the creating router needs to choose which priority value to use. The assignment priority is kept by the designated router when it starts advertising the assignment, and is useful when not enough prefixes are available.

- PRIORITY_DEFAULT SHOULD be used as default.
- Other values between PRIORITY_AUTO_MIN and PRIORITY_AUTO_MAX MAY be dynamically chosen by the implementation.
- Other values between PRIORITY_AUTHORITY_MIN and PRIORITY_AUTHORITY_MAX MUST NOT be used if not specified by an authority (by static or dynamic configuration).
- Other values are reserved.

6.9. Prefix Assignment Algorithm steps

At the beginning of the algorithm, all assignments that do not have their Authoritative bit set are marked as 'invalid', and the router computes for each connected link whether it is the designated router, as specified in Section 4.5.

The following steps are then executed for every combination of delegated prefixes and interfaces.

- If the current interface is external, ignore that interface.
- If the Delegated Prefix is strictly included in another Delegated Prefix, ignore that delegated prefix.

- If the Delegated Prefix is equal to another Delegated Prefix, advertised by some router with an higher router ID than the considered delegated prefix, ignore that delegated prefix.

- Look for a valid Assigned Prefix, advertised by another router on the current interface and included in the current Delegated Prefix.

- Look for a Chosen Prefix associated to the current interface and included in the current Delegated Prefix.

- There are four possibilities at this stage.

    1. If no AP is found, and no CP is found, a new assignment can be made if and only if the router considers itself as the designated router. Whether to create an assignment or not, and which prefix to use, is specified in Section 6.6.

    2. If an AP is found, and no CP is found, the AP MUST be accepted. The new CP's advertised bit MUST be set if and only if the router considers itself as the designated router.

    3. If no AP is found, and a CP is found, the router MUST check if the CP's assignment is valid. If it is, the local assignment is marked as valid and advertised. If it isn't, it is destroyed and the algorithm applies case 1.

    4. If both an AP and a CP are found, the router must check if the prefixes are the same. If they are different and if the CP's Authoritative bit is not set, the CP MUST be deleted and the algorithm applies case 2. If the prefixes are the same, the CP must be updated with the AP's priority value, marked as valid, and advertised if and only if the router considers itself as designated on the link.

In the end all the assignments that are marked as invalid are deleted.

6.10. Downstream DHCPv6 Prefix Delegation support

If some host or non-Homenet router asks for Delegated Prefixes, a router MAY assign a set of prefixes and give them to the client. Such assignments MUST be advertised as either not assigned on any link, or assigned on a stub virtual link connected to the router, depending on the Flooding Protocol capabilities. By default assignments priorities MUST be between PRIORITY_AUTO_MIN and PRIORITY_AUTO_MAX, SHOULD be lower than PRIORITY_DEFAULT, and the authoritative bit MUST not be set. Whenever such an assignment

becomes invalid, DHCPv6 Reconfigure SHOULD be used in order to remove the prefix from DHCPv6 DP client's lease. If DHCPv6 Reconfigure is not supported, leases lifetimes SHOULD be significantly small.

Provided DPs' valid and preferred lifetimes MUST be lower or equal to their associated Delegated Prefix's lifetimes, and associated DHCPv6 data SHOULD be provided to the DHCPv6 PD client.

By default, an assigned prefix SHOULD NOT be provided to a DHCPv6 PD client before the apply timeout has elapsed. But in order to allow faster response delay, a lease MAY first be provided with a lifetime of 2*FLOODING_DELAY seconds, even if the private assignments' apply timeout has not elapsed yet.

7. Address Assignment Algorithm

IPv6 routers always get at least one link-local address per link. Routing protocols and link DHCP servers are able to run with these addresses. In some cases though, a router may need to take one or multiple addresses among one or multiple available Delegated Prefixes. For example:

- The router needs connectivity to the internet (For management, NTP synchronization, etc...).
- The router needs connectivity within the home network (For management, DNS communications, etc...).
- IPv4 addresses are needed (DHCPv4, v4 link-local connectivity, etc...).

When possible, SLAAC MUST be used. In other cases a different mechanism is necessary for routers to get addresses. This document proposes an Address Assignment Algorithm that extends the Prefix Assignment Algorithm and works as follows. Each prefix assignment is associated with a fixed address pool, reserved for router's addresses assignment. The address pool is a prefix which value is deterministically function of the assigned prefix. A router MAY, at any time, decide to assign itself an address from any of its Chosen Prefixes. Just like prefix assignments, address assignments are advertised to other routers and collisions are detected. Routers MUST keep track of Address Assignments made by other routers on connected links by using information provided by the flooding algorithm, as defined in Section 5.5.

7.1. Router's address pools

Given an assigned prefix A/X (where all A's latest '128 - X'th bits are set to 0), the routers reserved address pool is defined as follows:

If X <= 64: SLAAC MUST be used

If X > 64 and X <= 110: The pool is A/112 (2^16 addresses)

If X >= 110 and X <= 126: The pool is A/(X + 2) (One quarter of the available addresses)

If X >= 126: Only the designated router MAY use A/128. Other routers MUST NOT get an address.

In the case of IPv4 prefixes, the network address (first address of the address pool) MUST not be used.

7.2. Address Assignment Algorithm

In this section, we say an address assignment is made by some router when it intends to use, or is using the address specified by this assignment. An assignment, made by some router, MUST be advertised on the link on which the assignment is made. Similarly, an address assignment is said to be applied when the address is pushed to the router's interface configuration. It is unapplied otherwise.

Routers MUST store applied address assignments in their stable storage and reuse the same addresses whenever possible. At least the five previously applied addresses SHOULD be stored for each interface.

For a given prefix assignment, an address is said to be available if it is within the router's address pool associated to the prefix assignment, and it is not being advertised by any other router. If the flooding protocol provides interface identifier in the address assignments, looking for collisions on considered link is enough.

A new address assignment MUST be chosen randomly among available addresses. An address assignment MUST NOT be applied when one of the following condition is true.

- The associated Chosen Prefix is not applied.
- The timer specified in Section 8 has not elapsed yet.

An address assignment must be deleted whenever one of the following condition becomes true.

- The associated Chosen Prefix is deleted or moved to another link.
- Some other router with a higher router ID is advertising the same address on the same link.

8. Hysteresis Principle

The IPv6 Stateless Address Autoconfiguration [RFC4862] states that host addresses can be kept up to 2 hours after a Router Advertisement wit zero lifetime is received. Therefore, routers must be carefull before assigning or deprecating a prefix.

8.1. Prefix and Address assignments

When the flooding protocol is started, the router MUST wait FLOODING_DELAY before executing the prefix assignment algorithm for the first time.

Prefix and address assignment algorithms are distributed. Collisions may occur, but network configuration, routing protocols or upper layers should not suffer from these collisions. For this reason, all assignments that could imply collisions are not immediately applied.

- A router MUST NOT apply a Chosen Prefix before it has waited 2*FLOODING_DELAY. If the entry is valid during the whole waiting time, it MUST be applied to the link it is assigned.
- A router MUST NOT apply an Assigned Address before it has waited 2*FLOODING_DELAY_LL. If the assignment is valid during the whole waiting time, it MUST be applied to the interface it is assigned.

8.2. Delegated Prefixes

Some links may be unreliable, causing repetitive connectivity loss. Such links shouldn’t cause IP reconfiguration.

8.2.1. Unreliable uplink

When a router detects uplink connectivity loss, Delegated Prefixes’ lifetimes from prefixes obtained through the uplink MUST be modified in the following way.

- The Preferred Lifetime is set to 0.

- The Valid Lifetime is set to the minimum between the current Valid Lifetime and two hours.
- The default route associated with the prefix is not advertised anymore.

This behavior is similar to [RFC7084] specifications and provides stable host configuration in case of unreliable uplink.

8.2.2. Unreliable in-home link

When a router stops advertising a Delegated Prefix, it MUST first deprecate that Delegated Prefix by advertising it for DP_DEPRECATE_FACTOR*FLOODING_DELAY seconds with zero valid and preferred lifetimes.

When a router receives a deprecated Delegated Prefix advertisement from the Flooding Protocol, it MUST remove the Delegated Prefix from its Delegated Prefixes list.

When a router stops receiving a Delegated Prefix from the Flooding Protocol, it SHOULD keep using that delegating prefix up to a period of min(remaining Valid Lifetime, DP_KEEP_ALIVE_TIME) seconds.

9. ULA and IPv4 Prefixes Generation

Although DHCPv6 PD and static configuration are regular means of obtaining IPv6 prefixes, routers MAY, in some cases, autonomously decide to generate a delegated prefix. In this section are specified when and how IPv6 ULA prefixes and IPv4 private prefixes may be autonomously generated.

9.1. ULA Prefix Generation

ULA prefixes can be randomly generated as specified in [RFC4193], enabling stable in-home IPv6 connectivity.

In this section, we say a ULA delegated prefix is ’stable’ if it has been the only advertised ULA delegated prefix for at least 2*FLOODING_DELAY seconds. The behaviour specified in the following sections tend to reuse a stable ULA prefix as long as its preferred lifetime is not null.

Additionaly, we say a router is the owner of a spontaneously generated ULA prefix if it randomly created the prefix in the first place. A router SHOULD NOT create more than one prefix this way, and MUST remember all the prefixes they own. As stated in the following sections, only the owner of a prefix can extend its lifetimes.

9.1.1. Choosing the ULA prefix

When a stable ULA prefix is advertised, all routers SHOULD remember that prefix alongwith its associated valid and preferred lifetime. If this prefix stops being advertised (e.g. due to a network split) while its preferred lifetime is not null, the same ULA prefix SHOULD be selected using the same valid and preferred lifetimes.

If there was no stable ULA prefix advertised, or if the preferred lifetime of the prefix was null, a prefix generated as specified in [RFC4193] SHOULD be used. In case the stable storage can't be used or the current date cannot be determined, the prefix MAY be pseudo-randomly generated based on hardware specific values.

9.1.2. Advertising a ULA prefix

A router MAY start advertising a ULA prefix whenever the two following conditions are met:

- It is the network leader.
- There is no other advertised ULA prefix.

If no IPv6 prefix is available at all, the network leader SHOULD start advertising a ULA delegated prefix.

Additionaly, a router SHOULD start advertising its own ULA prefix whenever the three following conditions are met:

- A stable ULA prefix is advertised by another router.
- The router owns the advertised stable ULA prefix.
- The preferred lifetime of the advertised ULA prefix is below 10 minutes.

This allows a router to restart advertising a owned prefix whenever the preferred lifetime is approaching zero. Which later allows him to extend the lifetime of the prefix.

A router MUST stop advertising a spontaenously generated ULA prefix whenever one of the two following condition is met:

- A different ULA prefix is being advertised.
- The same prefix is advertised by another router, and the router doesn't own that prefix.

9.1.3. Extending prefix lifetime

Routers MUST regularly extend the valid and preferred lifetimes of the ULA delegated prefix they advertise and own, so that they never drop to zero.

When a router advertises a prefix it doesn't own, lifetimes are never extended. When the preferred lifetime of the prefix approaches zero, either the owner of the prefix will start advertising the prefix with a non-zero preferred lifetime, or a new prefix will be generated.

9.1.4. Authoritative ULAs

This section doesn't prevent multiple ULA prefixes from existing simultaneously. ULA prefixes may be provided by different means, as specified in Section 4.3. Delegated prefixes that are delegated by a service provider or provisioned by an authority differ from 'spontaneously' generated prefixes. They MUST NOT be withdrawn if another ULA delegated prefix is observed.

When at least one of such ULA prefixes is used, spontaneously generated ULA prefixes are withdrawned.

9.2. IPv4 Private Prefix Generation

A router MAY generate an IPv4 prefix when the two following conditions are met.

- It has an IPv4 address with global connectivity.
- No other IPv4 delegated prefix is advertised by any other router.

A router MUST stop advertising an IPv4 prefix whenever another router with an higher router ID is advertising an IPv4 Delegated Prefix.

The IPv4 private prefix must be included in one of the private prefixes defined in [RFC1918]. The prefix 10/8 SHOULD be used by default but it SHOULD be configurable. In the case the address provided by the ISP is already a private address, a different private prefix SHOULD be used. For instance, if the ISP is giving the address 10.1.2.3, 10/8 or any sub-prefix included in 10/8 SHOULD NOT be used. (For instance, 172.16/12 or 192.168/16 can be selected).

10. Manageability Considerations

The algorithm leaves much room for implementation specific features. For instance, ULA prefix as well IPv4 prefix generation may be

disabled whenever a global IPv6 is made available. This section details a few other possible configuration options.

The implementation MAY allow each internal interface to be configured with a custom priority value. The specified priority SHOULD then be used when creating new assignments on the given interface. If not specified, the default priority SHOULD be used.

The implementation SHOULD allow manual assignments on given links. When specified, and whenever such an assignment is valid, it MUST be advertised as Authoritative Assignments on the given interface.

11. Documents Constants

PRIORITY_MIN	0
PRIORITY_AUTHORITY_MIN	4
PRIORITY_AUTO_MIN	6
PRIORITY_DEFAULT	8
PRIORITY_AUTO_MAX	10
PRIORITY_AUTHORITY_MAX	12
PRIORITY_MAX	15
DP_DEPRECATE_FACTOR	3
DP_KEEP_ALIVE_TIME	60 seconds

12. Security Considerations

Prefix assignment algorithm security entirely relies on flooding protocol security features. The flooding protocol SHOULD therefore check for the authenticity of advertised information. Security modes may be classified in three categories.

1. The flooding protocol is not protected.

2. The flooding protocol’s protection is binary: An allowed router may send any type of packets in the name of other routers.

3. All advertised messages are individually signed by the sender.

Whenever a malicious router attacks an unprotected network, or whenever a malicious router is able to authenticate itself to a network as stated in the second case, it may for example:

- Prevent other routers to get a stable router ID.

- Prevent other routers from making assignments by claiming the whole available address space.

o Redirect traffic to some router on the network.

If a malicious router is able to authenticate itself in a network protected as in the third case, most of the previously listed attacks may still be performed, but traffic could only be redirected toward the origination of the attack, and the source of the attack could be identified.

In any case, in order to protect the network, the routing protocol as well as the way hosts are configured also needs to be protected, hence requiring other link (e.g. WPA) or IP layer (e.g. IPSec-Auth [RFC4302] or SeND [RFC3971]) security solutions.

13. References

13.1. Normative References

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC3203] T'Joens, Y., Hublet, C., and P. De Schrijver, "DHCP reconfigure extension", RFC 3203, December 2001.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[RFC3646] Droms, R., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, December 2003.

[RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, November 2005.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC6106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 6106, November 2010.

[RFC6603] Korhonen, J., Savolainen, T., Krishnan, S., and O. Troan, "Prefix Exclude Option for DHCPv6-based Prefix Delegation", RFC 6603, May 2012.

13.2. Informative References

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[RFC5308] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, October 2008.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, November 2013.

[I-D.ietf-homenet-arch] Chown, T., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", draft-ietf-homenet-arch-11 (work in progress), October 2013.

[I-D.ietf-homenet-hncp]
Stenberg, M. and S. Barth, "Home Networking Control Protocol", draft-ietf-homenet-hncp-00 (work in progress), April 2014.

[I-D.ietf-ospf-ospfv3-autoconfig]
Lindem, A. and J. Arkko, "OSPFv3 Auto-Configuration", draft-ietf-ospf-ospfv3-autoconfig-06 (work in progress), February 2014.

[I-D.ietf-6man-why64]
Carpenter, B., Chown, T., Gont, F., Jiang, S., Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit Boundary in IPv6 Addressing", draft-ietf-6man-why64-06 (work in progress), October 2014.

[I-D.arkko-homenet-prefix-assignment]
Arkko, J., Lindem, A., and B. Paterson, "Prefix Assignment in a Home Network", draft-arkko-homenet-prefix-assignment-04 (work in progress), May 2013.

[I-D.bhandari-dhc-class-based-prefix]
Systems, C., Halwasia, G., Gundavelli, S., Deng, H., Thiebaut, L., Korhonen, J., and I. Farrer, "DHCPv6 class based prefix", draft-bhandari-dhc-class-based-prefix-05 (work in progress), July 2013.

[I-D.chelius-router-autoconf]
Chelius, G., Fleury, E., and L. Toutain, "Using OSPFv3 for IPv6 router autoconfiguration", draft-chelius-router-autoconf-00 (work in progress), June 2002.

[I-D.dimitri-zospf]
Dimitrelis, A. and A. Williams, "Autoconfiguration of routers using a link state routing protocol", draft-dimitri-zospf-00 (work in progress), October 2002.

Appendix A. Scarcity Avoidance Mechanism

Although an ISP should provide enough addresses, an implementation must carefully manage the provided address space. First, when a new assignment is made, the prefix should be selected amongst a set of prefixes so that prefix waste is minimized. Then, a router may decide to execute procedures intended to avoid prefix scarcity. Different approaches are possible. This section intends to provide guidelines for such procedures. They are optional and are compatible with routers that only support basic requirements defined in this document.

A.1. Prefix Wasts Avoidance

Given a Delegated Prefix, different routers may try to assign prefixes of different lengths. Particularly, a non-homenet downstream router may ask for a delegated prefix of significant size, as specified in Section 8.2. Some other routers, like sensors, may also require small prefixes. When randomly selected, a few /80s may easily prevent the assignment of bigger prefixes. Small prefixes should therefore be selected in neighboring areas.

For instance, given a delegated prefix 2001::/56 and an assigned prefix 2001::/64, the best prefix choice in order to reduce prefix space waste is 2001:0:0:1::/64. Other choices are then to be taken in 2001:0:0:2::/63, 2001:0:0:4::/62, 2001:0:0:8::/61, etc...

Creating an efficient prefix selection algorithm may be challenging as it needs to fullfill somehow contradictory requirements:

1. The prefix MUST be chosen amongst available prefixes, which implies that other routers may interfere with the process.

2. The prefix MUST be chosen randomly in a subset of available prefixes. When possible, the subset must be big enough to avoid collisions.

3. The prefix SHOULD be selected amongst prefixes that reduces the prefix space waste.

4. The prefix SHOULD be selected pseudo-randomly.

The following algorithm offers a satisfying tradeoff. Given a Delegated Prefix and the desired prefix length:

1. Compute the minimal subset of available prefixes included in the Delegated Prefix. In the example given previously, the minimal subset was {2001:0:0:1::/64, 2001:0:0:2::/63, ..., 2001:0:0:80::/57}.

2. Compute the set of prefixes of desired length so that:

   * It contains exactly RANDOM_SUBSET_SIZE prefixes, or all the available prefixes if there are less than RANDOM_SUBSET_SIZE available prefixes.

   * Prefixes are picked in the prefixes from the minimal subset of available prefixes which lengths are the highest.

* When multiple subsets are possible, privelege lexicographicaly lowest prefixes.

If RANDOM_SUBSET_SIZE equals 10, the subset would be {2001:0:0:1::/64, 2 /64s in 2001:0:0:2::/63, 4 /64s in 2001:0:0:4::/62, the 3 first /64s in 2001:0:0:8::/61}.

3. First try PSEUDO_RANDOM_TENTATIVE pseudo-random prefixes, computed from the DP, with the given length, based on interface specific hardware values (For instance using values generated like HASH(MAC Address : Counter). The hash function doesn't need to be cryptographic). The first prefix amongst this set that also is in the set computed at step 2 is chosen. If no prefix is found, try next step.

4. Choose a prefix randomly among prefixes in the subset computed at step 2.

This algorithm, defined as a sequence of prefix sets computation, may seem algorithmicaly complex, but it can be efficiently implemented. The key element in order to do so is the ability to iterate efficiently over all the available prefixes.

RANDOM_SUBSET_SIZE should provide sufficiently low collision probability. A value of 256 should be enough in most cases. PSEUDO_RANDOM_TENTATIVE is purely implementation dependent, but shouldn't be too high as the probability of finding an available prefix that way quickly decreases with the number of used prefixes. A value of 10 should be sufficient.

A.2. Increasing Assigned Prefix Length

When a new assignment can't be created, and if not forbidden by the router's configuration, the router MAY increase the size of the desired prefix. For instance, if an available /64 can't be found, the router may look for a /80. Nevertheless, this implies using DHCPv6 instead of SLAAC, which SHOULD be avoided.

A.3. Foreseeing Prefixes Exaustion

The previously proposed solution may be useful in some particular cases, but won't work when no more prefixes are available. A router MAY try to detect when default length prefixes are becoming rare. In such a situation, it MAY decide to allocate a longer prefix, part of an available shorter prefix. For instance, if A/64 is available, but there are not many other available /64, the router can try to allocate A/80. If the allocation doesn't raise any collision, this

procedure will prevent A/64 from being used by other hosts, hence creating a large set of smaller available prefixes to be used.

Such an allocation is considered dynamic. The Authoritative bit MUST NOT be set and the priority MUST be among values authorized as dynamically chosen in Section 6.8.

When different prefixes lengths are being used, the random prefix selection MUST NOT be uniform among all possibilities. Instead, it SHOULD privilege prefixes contained in bigger prefixes that cannot be allocated. For instance, if 2001::/56 is the DP, and 2001:0:0:0:1::/80 is an assigned prefix, other /80 should be randomly chosen in 2001:0:0:0:1::/64 before being chosen in other /64s.

A.4. Cutting an Existing Assignment

When specifically required by an authority (configuration or DHCP), a router MAY decide to un-assign one of its own assignment, in order to cut it in smaller prefixes, or to send an overriding assignment in order to force the network to stop using a particular prefix. Because such a procedure may imply links reconfiguration, it SHOULD be avoided whenever possible.

Such allocation are considered as required by an authority. The Authoritative bit MAY be set and the priority MUST be among values authorized as specified by an authority in Section 6.8.

As an example, if a router can’t find a /64 for a link that, with a high priority, must be given a /64, it chooses a prefix assigned by some other router, to another link, with a lower priority, and creates a new Chosen Prefix with a higher priority. The other router will be forced to remove its own assignment, hence making the new assignment valid.

Appendix B. Acknowledgments

This document is the continuation of the work being done in [I-D.arkko-homenet-prefix-assignment]. The authors would like to thank all the people that participated in the previous document’s development as well as the present one. In particular, the authors would like to thank to Tim Chown, Fred Baker, Mark Townsley, Lorenzo Colitti, Ole Troan, Ray Bellis, Markus Stenberg, Wassim Haddad, Joel Halpern, Samita Chakrabarti, Michael Richardson, Anders Brandt, Erik Nordmark, Laurent Toutain, Ralph Droms, Acee Lindem and Steven Barth for interesting discussions in this problem space. The authors would also like to point out some past work in this space, such as those in [I-D.chelius-router-autoconf] or [I-D.dimitri-zospf].

Authors' Addresses

Pierre Pfister
Cisco Systems
Paris
France

Email: pierre.pfister@darou.fr

Benjamin Paterson
Cisco Systems
Paris
France

Email: benjamin@paterson.fr

Jari Arkko
Ericsson
Jorvas 02420
Finland

Email: jari.arkko@piuha.net

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 5, 2015

J. Jeong
Sungkyunkwan University
J. Park
ETRI

September 1, 2014

DNS Name Autoconfiguration for Home Network Devices
draft-jeong-homenet-device-name-autoconf-01

Abstract

This document specifies an autoconfiguration scheme for DNS names of home network devices. By this scheme, the DNS name of a home network device can be autoconfigured with the device's category and model in a home network. This DNS name lets home residents easily identify each device for monitoring and remote-controlling it in a home network.

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on March 5, 2015.

Table of Contents

1. Introduction

Many appliances (such as smart TV, refrigerator, air conditioner, and washing machine) in a home network have begun to have WiFi capability for monitoring and remote-controlling within a home network or from the Internet. Also, Internet of Things (IoT) devices (such as light, meter, room temperature controller, and sensors) have been installed into home networks for the easy management of home environments.

For the Internet connectivity of home network devices, a variety of parameters (e.g., IPv6 addresses, default routers, and DNS servers) can be automatically configured by Neighbor Discovery (ND) for IP Version 6, IPv6 Stateless Address Autoconfiguration, and IPv6 Router Advertisement (RA) Options for DNS Configuration [RFC4861][RFC4862][RFC6106].

For these home appliances and IoT devices, the manual configuration of DNS names will be cumbersome and time-consuming as the number of them increases rapidly in a home network. It will be good for such DNS names to be automatically configured such that they are readable to home residents.

This document proposes an autoconfiguration scheme for DNS names of home network devices. Since an autoconfigured DNS name contains the device category and model of a device, home residents can easily identify the device. With this device category and model, they will be able to monitor and remote-control each device with mobile smart devices, such as smartphone and tablet.

1.1. Applicability Statements

It is assumed that home network devices have Ethernet or WiFi capability (e.g., IEEE 802.11 series [IEEE-802.11] [IEEE-802.11a] [IEEE-802.11b][IEEE-802.11g] [IEEE-802.11n]) and are connected to a local area network (LAN) or a wireless LAN (WLAN).

Also, it is assumed that each home network device has a factory configuration (called device configuration) having device category (e.g., smart TV, smartphone, tablet, and refrigerator) and model (i.e., a specific model name of the device). This device configuration can be read by the device for DNS name autoconfiguration.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Terminology

This document uses the terminology described in [RFC4861] and [RFC4862]. In addition, four new terms are defined below:

- Device Configuration: A factory configuration that has device category (e.g., smart TV, smartphone, tablet, and refrigerator) and model (i.e., a specific model name of the device).

- DNS Search List (DNSSL): The list of DNS suffix domain names used by IPv6 hosts when they perform DNS query searches for short, unqualified domain names [RFC6106].

- DNSSL Option: IPv6 RA option to deliver the DNSSL information to IPv6 hosts [RFC6106].

4. Overview

This document specifies an autoconfiguration scheme for a home network device using device configuration and DNS search list. Device configuration has device category and device model. DNS search list has DNS suffix domain names that represent DNS domains of the home network having the home network device [RFC6106].

As an IPv6 host, the home network device can obtain DNS search list through IPv6 Router Advertisement (RA) with DNS Search List (DNSSL) Option [RFC4861][RFC6106] or DHCPv6 with Domain Search List Option [RFC3315][RFC3736][RFC3646].

The home network device can construct its DNS name with the concatenation of device category, device model, and domain name. Since there exist more than one device with the same model, the DNS name should have a unique identification to differentiate multiple devices with the same model.

Since both RA and DHCPv6 can be simultaneously used for the parameter configuration for IPv6 hosts, this document considers the DNS name autoconfigurtion in the coexistence of RA and DHCP.

5. DNS Name Autoconfiguration

The DNS name autoconfiguration for a home network device needs the acquisition of DNS search list through either RA [RFC6106] or DHCPv6 [RFC3646]. Once the DNS search list is obtained, the home network device autonomously constructs its DNS name(s) with the DNS search list and its device information.

5.1. DNS Name Format

A DNS name for a home network device has the following format as in Figure 1:

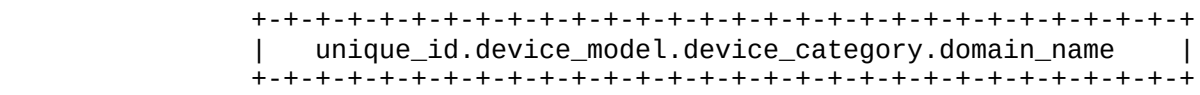


Figure 1: Home Network Device's DNS Name Format

Fields:

unique_id	unique identifier to guarantee the uniqueness of the DNS name in ASCII characters. The identifier MAY be a sequence number or alphanumeric with readability, such as product name.
device_model	device's model name in ASCII characters. It is a product model name provided by the manufacturer.
device_category	device's category name in ASCII characters, such as TV, refrigerator, air conditioner, smartphone, tablet, light, and meter.
domain_name	DNS domain name that is encoded according to the specification of "Representation and use of domain name" of RFC 3315.

5.2. Procedure of DNS Name Autoconfiguration

The procedure of DNS name autoconfiguration is performed through a DNSSL option delivered by either RA [RFC6106] or DHCPv6 [RFC3646].

5.2.1. Procedure of Device Name Generation

When as an IPv6 host a device receives a DNSSL option through either RA or DHCPv6, it checks the validity for the DNSSL option. If the option is valid, the IPv6 host performs the DNS name autoconfiguration with each DNS suffix domain name in the DNSSL option as follows:

1. The host constructs its DNS name with the DNS suffix domain name along with device configuration and a selected identifier (as unique_id) that is considered unique.

2. The host performs the uniqueness test of the constructed DNS name. The uniqueness test is performed through duplicate address detection (DAD) procedure in ND [RFC4861][RFC4862]. See Section 5.2.2 for the detailed test procedure.

3. If the DNS name is proven to be unique, it is used as the device's DNS name and the DNS autoconfiguration is done for the given DNS suffix domain name. Otherwise, go to Step 1.

When the DNS search list has more than one DNS suffix domain name, the IPv6 host repeats the above procedure until all of the DNS suffixes are used for the DNS name autoconfiguration.

5.2.2. Uniqueness Test of Device DNS Name

An IPv6 host generates an IPv6 address with 64-bit prefix from an RA option (or DHCPv6) and 64-bit hash value from the DNS name to be tested. Before using such an IPv6 address associated with the DNS name, the IPv6 host performs the DAD to check whether the address belongs to another IPv6 host or not. Note that the IPv6 host configures the IPv6 address corresponding to the DNS name as its address. If the address belongs to another IPv6 host, it is considered that the DNS name corresponding to the address is occupied by a different host. Thus, the IPv6 host selects another unique identifier (as unique_id) for a DNS name and repeats the uniqueness test of the new DNS name with the identifier.

1. The host computes the hash value of the DNS name to be tested for the uniqueness using a hash function (e.g., MD5 and SHA-1). It takes the first 64 bits of the hash value from most significant bit.

2. The host performs the uniqueness test of the constructed DNS name. The uniqueness test is performed through the DAD procedure in ND [RFC4861][RFC4862].

3. If the DNS name is proven to be unique with no response for the DAD, the device configures the DNS name and the corresponding IPv6 address as its own DNS name and address, respectively, returning the success of the uniqueness test. Otherwise, return the failure of the uniqueness test.

5.2.3. Collection of Device DNS Names

Once as IPv6 hosts the devices have autoconfigured their DNS names, as a collector, any IPv6 node (i.e., router or host) in the same subnet can collect the device DNS names using IPv6 Node Information (NI) protocol [RFC4620].

For a collector to collect the device DNS names without any prior node information, a new NI query needs to be defined. That is, a new ICMPv6 Code (e.g., 3) SHOULD be defined for the collection of the IPv6 host DNS names. The Data field is not included in the ICMPv6 header since the NI query is for all the IPv6 hosts in the same subnet. The Qtype field for NI type type is set to 2 for Node Name.

The query SHOULD be transmitted by the collector to a link-local multicast address for this NI query. Assume that a link-local multicast address SHOULD be defined for device DNS name collection and that all the IPv6 hosts join this link-local multicast address for the device DNS name collection service.

When an IPv6 host receives this query sent by the collector in multicast, it transmits its Reply with a random interval between zero and [Query Response Interval, as defined by Multicast Listener Discovery Version 2 [RFC3810]. This randomly delayed Reply allows the collector to collect the device DNS names with less frame collision probability by spreading out the Reply time instants.

After the collector collects the device DNS names, it collects the IPv6 addresses corresponding to the DNS names by NI protocol [RFC4620]. For DNS name resolution service, the collector can register the pair(s) of DNS name and IPv6 address for each IPv6 host into a recursive DNS server known to the collector using DNS dynamic update [RFC2136].

6. Location-Aware DNS Name Configuration

A DNS name can include location information to let home residents easily identify the physical location of each device. In this document, location is categorized into macro-location and micro-location according to whether the location is a physical location or device.

6.1. Macro-Location-Aware DNS Name

If location information (such as living room, kitchen, and bedroom) is available to a home network device, a keyword for the location can be used to construct a DNS name as subdomain name. This location information lets home residents track the position of mobile devices (such as smartphone, tablet, and vacuum cleaning robot). The physical location of the device is defined as macro-location for DNS naming.

A subdomain name for macro-location MAY be placed between device_category and domain_name of the DNS name format in Figure 1. A localization scheme for device location is beyond the scope of this

document.

6.2. Micro-Location-Aware DNS Name

An IoT device (e.g., refrigerator) can have multiple other IoT devices (e.g., containers of a refrigerator) within itself. A device containing other devices is defined as micro-location for DNS naming.

A subdomain name for micro-location MAY be placed between device_category and domain_name of the DNS name format in Figure 1. A localization scheme for micro-location is beyond the scope of this document.

To denote both macro-location and micro-location into a DNS name, the following format is described as in Figure 2:

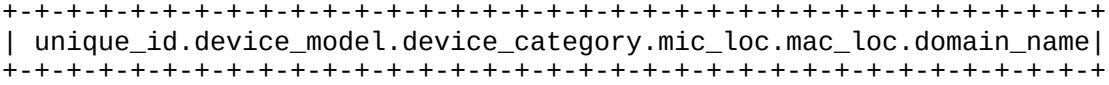


Figure 2: Location-Aware Device DNS Name Format

Fields:

unique_id	unique identifier to guarantee the uniqueness of the DNS name in ASCII characters. The identifier MAY be a sequence number or alphanumeric with readability, such as product name.
device_model	device’s model name in ASCII characters. It is a product model name provided by the manufacturer.
device_category	device’s category name in ASCII characters, such as TV, refrigerator, air conditioner, smartphone, tablet, light, and meter.
mic_loc	device’s micro-location, such as refrigerator.
mac_loc	device’s macro-location, such as kitchen.
domain_name	DNS domain name that is encoded according to the specification of "Representation and use of domain name" of RFC 3315.

7. Security Considerations

This document shares all the security issues of the NI protocol that are specified in the "Security Considerations" section of [RFC4620].

8. Acknowledgements

This work was partly supported by the ICT R&D program of MSIP/IITP [10041244, SmartTV 2.0 Software Platform] and ETRI.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC6106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 6106, November 2010.

[RFC3315] Droms, R., Ed., "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, April 2004.

[RFC3646] Droms, R., Ed., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, December 2003.

9.2. Informative References

[RFC4620] Crawford, M. and B. Haberman, Ed., "IPv6 Node Information Queries", RFC 4620, August 2006.

[RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

[RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997.

[IEEE-802.11] IEEE Std 802.11, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", March 2012.

[IEEE-802.11a] IEEE Std 802.11a, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in the 5 GHZ Band", September 1999.

[IEEE-802.11b] IEEE Std 802.11b, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band", September 1999.

[IEEE-802.11g] IEEE P802.11g/D8.2, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher Data Rate Extension in the 2.4 GHz Band", April 2003.

[IEEE-802.11n] IEEE P802.11n/D9.0, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 5: Enhancements for Higher Throughput", March 2009.

Authors' Addresses

Jaehoon Paul Jeong
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon, Gyeonggi-Do 440-746
Republic of Korea

Phone: +82 31 299 4957
Fax: +82 31 290 5119
EMail: pauljeong@skku.edu
URI: http://cpslab.skku.edu/people-jaehoon-jeong.php

Jung-Soo Park
Electronics and Telecommunications Research Institute
218 Gajeong-Ro, Yuseong-Gu
Daejeon, 305-700
Republic of Korea

Phone: +82 42 860 6514
EMail: pjs@etri.re.kr

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 30, 2015

P. Pfister

October 27, 2014

Multicast enabled Home Network using PIM-SSBIDIR and HNCP
draft-pfister-homenet-multicast-00

Abstract

This document specifies a possible solution enabling multicast routing in a home network. It relies on the Source-Specific Bidirectional variant of the Protocol Independent Multicast routing protocol (PIM-SSBIDIR). HNCP is used to elect the Rendezvous Point address and a Proxy Controller connected to the Rendezvous Point Link. Additionally, PIM-SSBIDIR routers behavior is slightly modified on the Rendezvous Point Link so that the Proxy Controller may know the home-wide subscription state. Note that this document defines one single working solution to the stated problem: Inputs regarding other possibilities are welcome.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 30, 2015.

Copyright Notice

Table of Contents

1. Introduction

IP multicast is used not only for link-local communications but also for site-local exchanges (UPnP [UPnP] or TV over IP). Additionally, we can expect new connected objects will make use of this technique for diverse purposes. Most link types like Ethernet or 802.11 support link-local multicast natively, but a multicast routing protocol is required when multiple links are present. The Protocol Independent Multicast [RFC4601] is one of the most widely used multicast routing protocol. Unfortunately, home networks have some peculiarities that makes it unsuitable without changes.

This document lists the specificities of home networks regarding multicast, the problems resulting from these peculiarities and specifies how homenet routers must behave in order to enable multicast routing for both in-home and ISP originated traffic in multi-homed environments.

The solution makes use of the Source-Specific Bidirectional variant of the Protocol Independent Multicast routing protocol (PIM-SSBIDIR - [pim-ssbidir]) for routing multicast traffic inside the home, and PIM Border Proxies ([pim-border-proxy]) for subscribing on all uplink interfaces. Two new HNCP TLVs are defined. One is used in the Rendezvous Point Address (RPA) and Proxy Controller election process, the other is used for advertising PIM Border Proxies. In addition, PIM-SSBIDIR behavior is slightly modified on the RP Link allowing the Proxy Controller, connected on the RP Link, to acquire the home-wide subscription state.

This document specifies a functional solution enabling multicast routing in multi-homed home networks. Inputs regarding other possibilities are very welcome and expected, so the best design may be adopted.

2. Problem Analysis

Current home networks usually consist of a single link and therefore support link-local multicast using MLDv2 [RFC3810] or IGMPv3 [RFC3376] for both all-source (ASM) and source-specific (SSM) multicast. Future home networks ([I-D.ietf-homenet-arch]) will consist of multiple links, which means multicast routing will be required.

This section discusses home network requirements and problems related to multicast routing.

2.1. Requirements

Future home networks should at least provide the same multicast features as the existing home networks.

In-home traffic: Devices inside the home should be able to send and receive multicast traffic originated inside the home.

ISP to Home traffic: Devices inside the home should be able to receive multicast traffic coming from an ISP.

Home to ISP traffic: Although traffic originated inside the home MUST NOT be forwarded on external interfaces by default, it should not be precluded.

On top of that, home network environments add the following constraints, defined in the Homenet architecture document.

Autoconfiguration: It must function without human interactions.

Multi-Homing: It must support multiple uplinks and therefore multiple default routes.

This document makes no assumptions on the technique used by ISPs to provide multicast traffic. It allows border routers to act as PIM Border Proxies, translating the home-wide subscription state toward every multicast enabled home uplink. Border router default behavior SHOULD consist in using MLDv2 and IGMPv3 on all uplink interfaces. Similarly, multicast enabled ISPs SHOULD listen to MLDv2 and IGMPv3 subscriptions coming from CPEs, and provide multicast traffic accordingly.

Note that this document doesn't preclude the use of different techniques. For example, an ISP-provided CPE may be specifically configured to translate in-home multicast subscriptions into PIM requests on the ISP link. But this is outside the scope of this document.

2.2. Specific Problems

Both PIM Bootstrap Mechanism (PIM BSR - [RFC5059]) and the Homenet Configuration Protocol (HNCP - [I-D.ietf-homenet-hncp]) could be used for autoconfiguration purposes. As HNCP support is already required in all homenet routers, this document proposes to use it instead of its PIM equivalent.

PIM-SM [RFC4601], PIM-BIDIR [RFC5015] and PIM-SSM were designed to function in single routed domains. Extensions allow multiple domains to be connected one with each other, but they all require specific PIM interactions between the domains, and a non-ambiguous knowledge of the next hop router for any multicast source. Given homenet constraints, we encounter the two following problems.

2.2.1. Uplink subscription problem

Initially, PIM reacts to two types of events. MLDv2/IGMPv3 subscriptions and multicast traffic origination. As receiving traffic from the ISP requires a subscription to happen first, border routers need some knowledge of the home-wide subscription state. In a single-homed network, the border router could be the RP, but in a multi-homed network, this subscription information must be shared between all border routers.

2.2.2. Uplink source localization problem

In multi-homed networks, routers have multiple default routes (one for each uplink). Unicast routing is achieved by looking at both

source and destination addresses, but this technique can’t be used when forwarding Join/Prune messages.

When multiple default routes point to different next-hop routers, Source-Specific Join/Prune messages’ next-hop cannot be reliably determined. A possible but not very scalable solution would consist in letting all the routers dynamically know where are every sources located. This document proposes to makes use of PIM-SSBIDIR instead.

3. Homenet Multicast Support Specifications

3.1. General Requirements

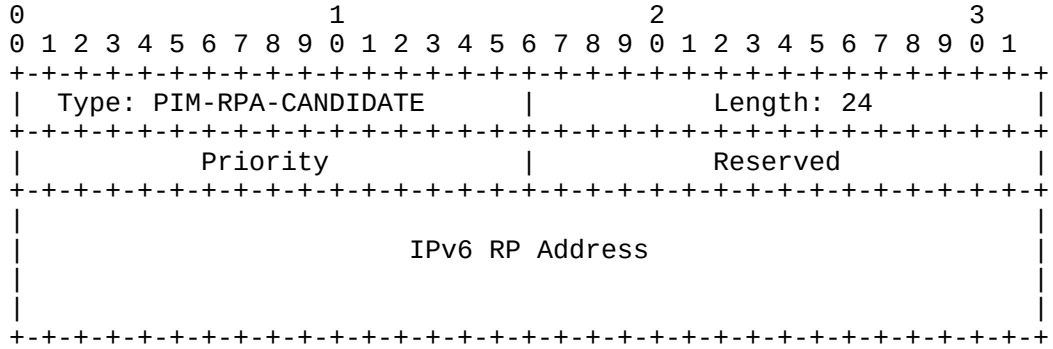
In order to deliver multicast traffic to subscribed devices, all homenet routers MUST implement PIM-SSBIDIR as well as the specifications presented in the present document.

Whenever the present document doesn’t conform to PIM specifications, behavior and configuration values described in this document take precedence.

3.2. Rendezvous Point Address Election Process

PIM-SSBIDIR and PIM-BIDIR both rely on the mapping between group ranges and Rendezvous Point Addresses. In PIM-SSBIDIR a Rendezvous point address doesn’t need to belong to an actual router but rather identify the Rendezvous Point Link. This is still true in the present document, but in addition to the RP Address, HNCP is used to elect a single Proxy Controller, directly connected to the RP Link.

In order to elect the RPA and Proxy Controller, the following HNCP TLV is defined.



PIM RPA Candidate TLV

The Rendezvous Point Address is chosen among all the advertised PIM RPA Candidate TLVs. The TLV with the highest priority is chosen first. In case of tie, the highest RPA address is preferred. The elected Proxy Controller is the router with the highest router ID advertising the elected PIM RPA Candidate TLV.

A router MUST start advertising a PIM RPA Candidate TLV (and thus candidate as Proxy Controller) whenever one of the two following condition is met.

- There is no currently advertised PIM RPA Candidate TLV network-wide.
- All the advertised PIM RPA Candidate TLVs have priority values lower than the one specified in the router's configuration and it is specifically stated by configuration that the router should try overcoming the currently elected RP.

A router MUST stop advertising a PIM RPA Candidate TLV whenever another advertised PIM RPA Candidate TLV takes precedence over its own one.

A router MUST NOT advertise more than one PIM RPA Candidate TLV. An advertised PIM RPA Candidate TLV MUST contain an IPv6 address known by all home routers and associated with a directly connected link. A Priority value of 0 SHOULD be used, unless stated otherwise by dynamic (DHCP, netconf, ...) or static (file) configuration.

When the RP Address is not valid anymore, the elected Proxy Controller MUST replace the advertised RP Address with a new, valid, RP Address. Such an event SHOULD be avoided. Therefore, an address with a long valid lifetime SHOULD be preferred.

3.3. PIM Border Proxy behavior

All routers with at least one uplink interface SHOULD behave as PIM Border Proxies, as specified in [pim-border-proxy], unless specified otherwise by static or dynamic configuration. They SHOULD proxy the received subscription state onto uplink interfaces for all groups of global scope.

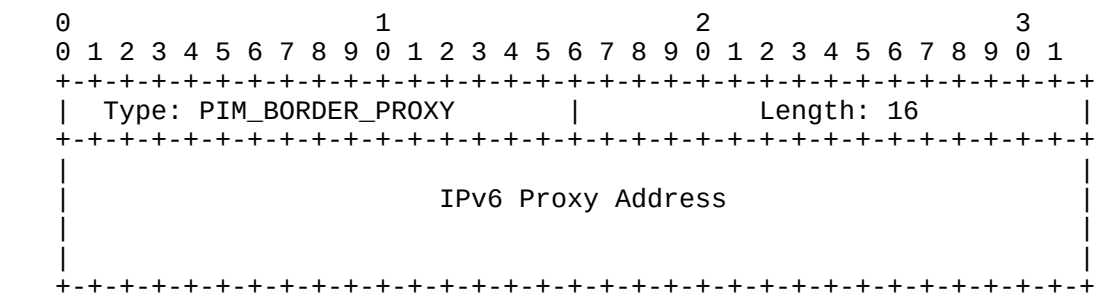
Multicast proxying is a local operation subject to numerous optimizations and configuration, particularly on ISP-provided CPEs. The following list specifies the default behavior.

- All groups with non-global scope SHOULD be ignored.

- The home-wide subscription state SHOULD be proxied on all uplink interfaces.

- The uplink default protocols are MLDv2 for IPv6 groups and IGMPv3 for IPv4 groups.

In addition, PIM Border Proxy routers MUST advertise the following TLV in their HNCP Node State.



PIM Border Proxy TLV

The elected Proxy Controller must behave as specified in [pim-border-proxy]. It MUST establish one peering for each address specified in PIM Border Proxy TLVs. It MUST reflect the home-wide subscription state toward all border proxies, computed based on all per-interface PIM downstream state machines and on-link local subscriptions, as if the RP was reachable on a virtual uplink interface.

3.4. PIM-SSBIDIR changes

This section specifies the changes made to PIM-SSBIDIR, required in the homenet context.

3.4.1. Router’s behavior on the RP Link

PIM-SSBIDIR always forwards the multicast traffic toward the RP Link and therefore never sends Join/Prune packets on the RP Link nor requires routers to listen to local subscriptions on the RP Link. But the elected Proxy Controller needs to know the home-wide subscription state. Which is why router’s behavior is modified on the RP Link.

All routers MUST operate the (*,G), (S,G) and (S,G,rpt) upstream state machines on all their interfaces, including the RP Link. On the RP Link, no DF Election process takes place. When sending Join/

Prune messages on the RP Link, the DF address is replaced with the RP Address.

The elected Proxy Controller MUST as well operate the downstream per-interface (*,G), (S,G) and (S,G,rpt) state machines on the RP Link, as well as enable multicast querying. Other routers connected to the RP Link SHOULD enable both downstream state machines and multicast querying as well in order to improve transition whenever the Proxy Controller would change.

3.4.2. Timing Considerations

PIM is an unreliable protocol. When a Join message is lost, the protocol waits for the next one, which by default comes after 60 seconds. A very typical use case for IP multicast is TV over IP, but we can't expect a user to wait 60 seconds when it changes the TV channel. Therefore, the default period between Join/Prune messages is reduced.

t_periodic: Default = 5 secs.

Similarly, PIM sends Hello messages every 30 seconds, which means dead neighbor detection occurs after 90 seconds. Therefore, the Hello period is reduced.

Hello_Period: Default = 10 secs.

4. Security Considerations

This document mostly relies on HNCP and PIM-SSBIDIR and therefore doesn't add much new threats.

The RP election process could be attacked whenever HNCP is not protected. Similarly, an attacker could advertise numerous PIM Border Proxy TLVs as a Deny of Service attack vector.

In order to operate securely, both HNCP and PIM-SSBIDIR should be secured.

5. IANA Considerations

IANA is kindly requested to reserve two new HNCP TLV identifiers:

- PIM Border Proxy TLV: PIM_BORDER_PROXY
- PIM RPA Candidate TLV: PIM-RPA-CANDIDATE

6. References

6.1. Normative References

[RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

[I-D.ietf-homenet-hncp] Stenberg, M. and S. Barth, "Home Networking Control Protocol", draft-ietf-homenet-hncp-00 (work in progress), April 2014.

[pim-ssbidir] Pierre Pfister, "Source Specific support for Bidirectional Protocol Independent Multicast", October 2014, <http://tools.ietf.org/html/draft-pfister-pim-ssbidir-00>.

[pim-border-proxy] Pierre Pfister, "Protocol Independent Multicast Border Proxying", October 2014, <http://tools.ietf.org/html/draft-pfister-pim-border-proxy-00>.

6.2. Informative References

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC5015] Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)", RFC 5015, October 2007.

[RFC5059] Bhaskar, N., Gall, A., Lingard, J., and S. Venaas, "Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)", RFC 5059, January 2008.

[UPnP] UPnP Forum, "Internet Gateway Device (IGD) Standardized Device Control Protocol V 1.0", November 2001.

[I-D.ietf-homenet-arch] Chown, T., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", draft-ietf-homenet-arch-11 (work in progress), October 2013.

Appendix A. Acknowledgments

The author would like to thank Steven Barth and Mohammed Hawari for their help in the specification and implementation process, as well as Mark Townsley, Stig Venaas, IJsbrand Wijnands and Markus Stenberg for their useful inputs.

Author's Address

Pierre Pfister
Paris
France

Email: pierre@darou.fr