Link Layer Addresses Assignment Mechanism for DHCPv6

- IETF-104 (Prague)
- Wednesday, 27 March 2019
- Bernie Volz
- draft-bvtm-dhc-mac-assign

Last Edit: 2019-03-20 14:15 EDT (BV)
• RFC 7241 defines cooperation between IEEE 802 and IETF and there are periodic discussions
• IEEE 802c split “local” MAC address space into 4 quadrants to provide for different allocation schemes
• IEEE 802cq is working on defining allocation mechanisms
• Several from IETF leadership (Ralph Droms, Russ Housley, Suresh Krishnan) thought that DHCPv6 might be usable as a MAC address allocation (802cq) mechanism
Background (2/3)

- Ralph Droms (IETF) reached out to Bernie
- Tomek and Bernie discussed and decided to work on issue, published draft-bvtm-dhc-mac-assign
- More background about 802c/cq in Pat Thaler’s “Emerging IEEE 802 Work on MAC Addressing” slides from IETF-96 ([https://datatracker.ietf.org/meeting/96/materials/slides-96-edu-ieee802work-0/](https://datatracker.ietf.org/meeting/96/materials/slides-96-edu-ieee802work-0/))
- Tomek and Bernie presented to IEEE RAN meeting in late May 2018
- Call for adoption to DHC WG failed (Jan 2019)
IETF-IEEE 802 coordination call in February discussed work

- While DHCPv6 link layer address assignment does not fulfill all requirements of P802.1CQ, it is a sound solution for some use cases, and P802.1CQ intends to adopt it
Use Cases for MAC addresses

• Hypervisor – to allocate the Virtual Machines
  – Lots of VMs
  – May have short or long life
  – May be possible to reuse addresses for different network segments based on data center

• IoT devices
  – Often short lived/disposable
  – Little need for global MAC address

• Individual clients
Why DHCPv6?

• Existing infrastructure: protocol, network, tools
• Servers already know how to manage and assign resources
• Protocol easily extensible
• We’re in DHC WG …
Focused on Hypervisor use case where Hypervisor needs a block of MAC addresses to assign to VMs

Can also be used by actual clients, but requires:

- IPv6 support
- A short-term temporary MAC address for link-local IPv6 address to request DHCPv6 assigned MAC address
- Client should use a non-link-layer address for DUID (DUID-EN or DUID-UUID)
Defines 2 New DHCPv6 Options

• IA_LL (Identity Association for Link Layer Addresses) Option
  – Similar to IA_NA and IA_PD
  – Used as container option for requested / assigned link-layer addresses

• LLADDR (Link Layer Addresses) Option
  – Similar to IAADDR and IAPREFIX
  – Used to request/assign link-layer addresses
### IA_LL Option

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

• IAID identifies instance of IA_LL option to allow for many
• T1 is renewal time (from “now” in seconds)
• T2 is rebinding time (from “now” in seconds)
• IA_LL-options contains one or more IA_LL options
### LLADDR Option

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OPTION_LLADDR</td>
<td>option-len</td>
<td></td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>link-layer-type</td>
<td>link-layer-len</td>
<td></td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>link-layer-address</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>extra-addresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>valid-lifetime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Link-layer-type and link-layer-len specify requested link-layer address
- Link-layer-address specifies starting address requested or assigned
- Extra-address specifies number of additional addresses (0 for single address)
- Valid-lifetime lifetime of assignment (from “now” in seconds)
- LLaddr-options could contain future options specific to assignment
Client / Server Operation (1)

- DHCPv6 essentials the same as address / prefix delegation
- But a bit simpler overall
  - Confirm, Decline, and Information-Request client messages not used
Client / Server Operation (2)

• For hypervisor model
  – Hypervisor is client, but does not use resulting link-layer addresses
  – Hypervisor could obtain large blocks or one link-layer address per VM as needed
  – Hypervisor provides link-layer address to VMs
  – VMs could do standard DHCPv6 for IPv6 addresses/delegated prefixes or DHCPv4
Client / Server Operation (3)

- If end client (e.g. IoT) wants a link-layer address
  - Could use Temporary MAC address for anonymity (see https://mentor.ieee.org/802.11/dcn/02/11-02-0109-00-000i-temporary-mac-address-for-anonymity.ppt) to do DHCPv6 to get “long term” link-layer address assignment
  - Clarify client must not use DUID-LL/LLT based on temporary MAC
  - Client then uses assigned link-layer address for normal DHCPv6, DHCPv4, …
Next Steps

• Is this work of interest to DHC?
  – Call for adoption failed in Jan 2019
  – IEEE would like to see work progress (intends to adopt for some use cases)
  – Should we try for another call for adoption?
• Provide feedback to authors about draft
• Carlos will discuss an extension to this work
Question or comments …

&

THANKS