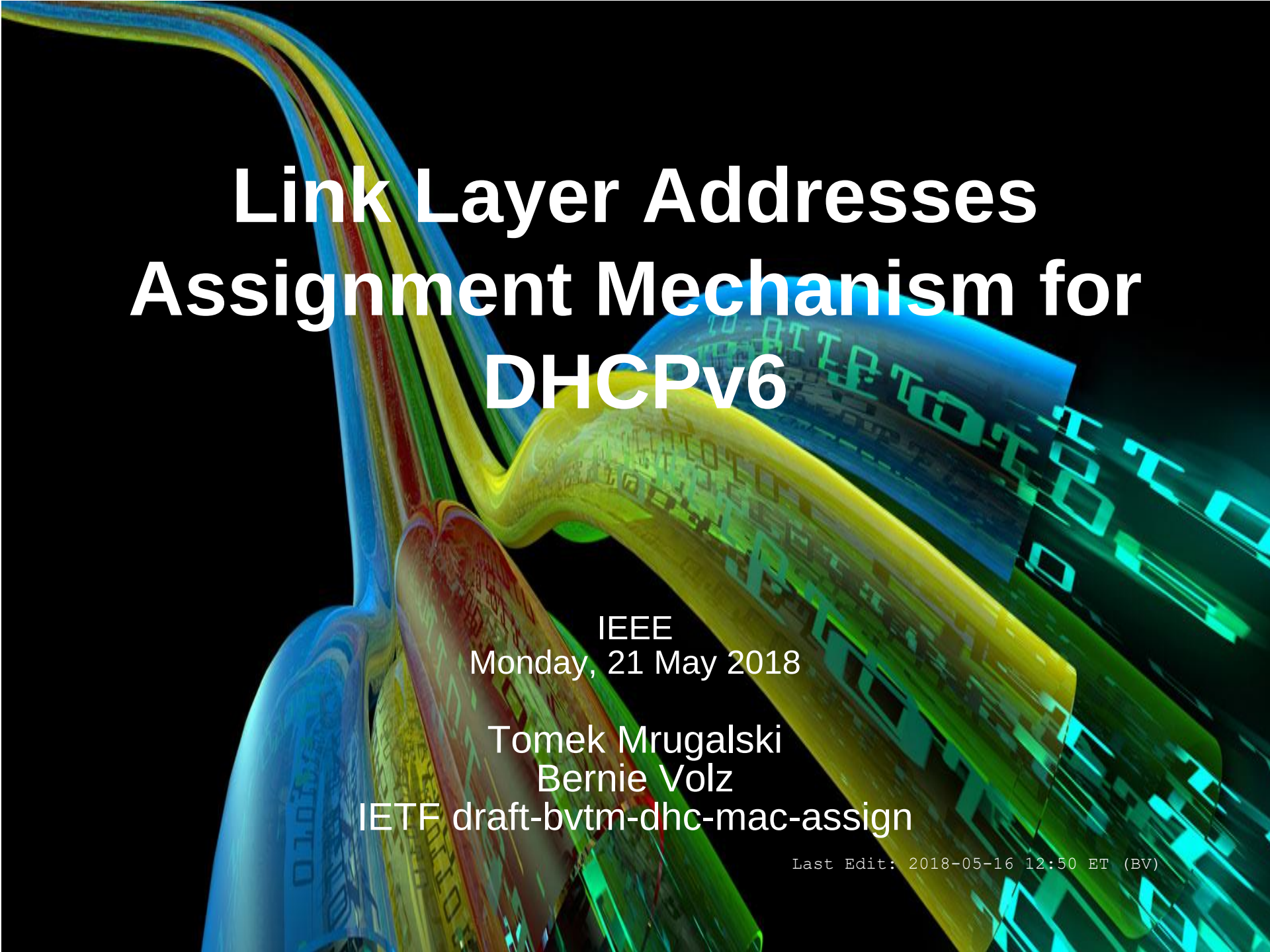


Link Layer Addresses Assignment Mechanism for DHCPv6

IEEE
Monday, 21 May 2018

Tomek Mrugalski
Bernie Volz
IETF draft-bvtm-dhc-mac-assign

Last Edit: 2018-05-16 12:50 ET (BV)

Background (1/2)

- 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 (2/2)

- Ralph Droms (IETF) reached out to Bernie Volz (IETF DHC WG co-chair)
- Tomek Mrugalski (other IETF DHC WG co-chair) and Bernie discussed and decided to work on issue
- Hence, the new I-D: 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/)

Why Not Randomly Assign?

Number of tries	Possible combinations	Collision chance	No collision chance
23 people	365 days	49,95%	50,05%
1024 VMs	2^{24} (One OUI)	3,07%	96,93%
4824 VMs	2^{24} (One OUI)	50,01%	49,99%
1M VMs	2^{44} (Local address quadrant)	3,08%	96,92%
1M VMs	2^{46} (“I know better than IEEE”)	0,71%	99,29%

- Birthday paradox: https://en.wikipedia.org/wiki/Birthday_problem
- Roughly the same probability for IPv6 uniqueness, and IPv6 does DAD
- Calculator: https://instacalc.com/28845

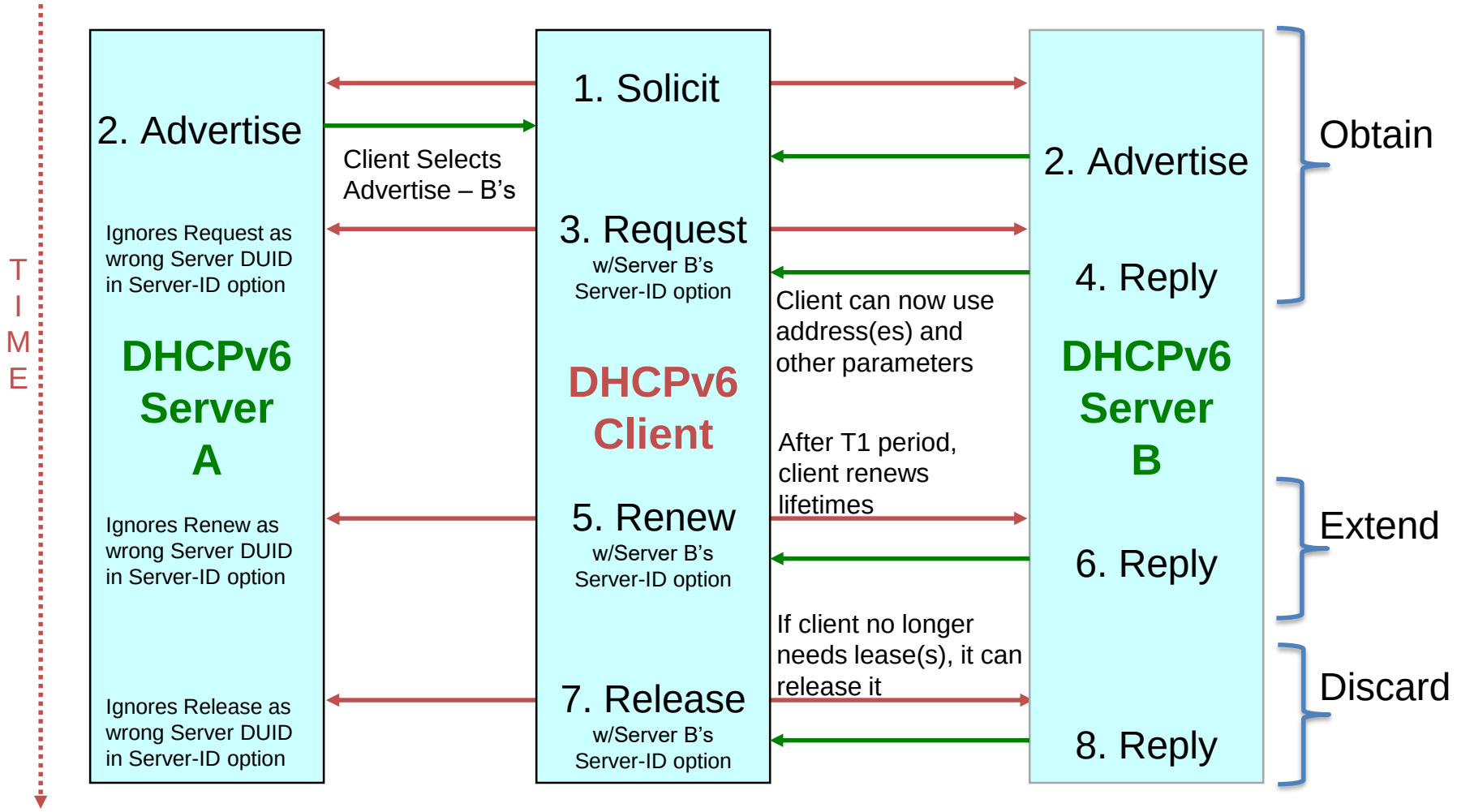
Use Cases for MAC addresses

- 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?

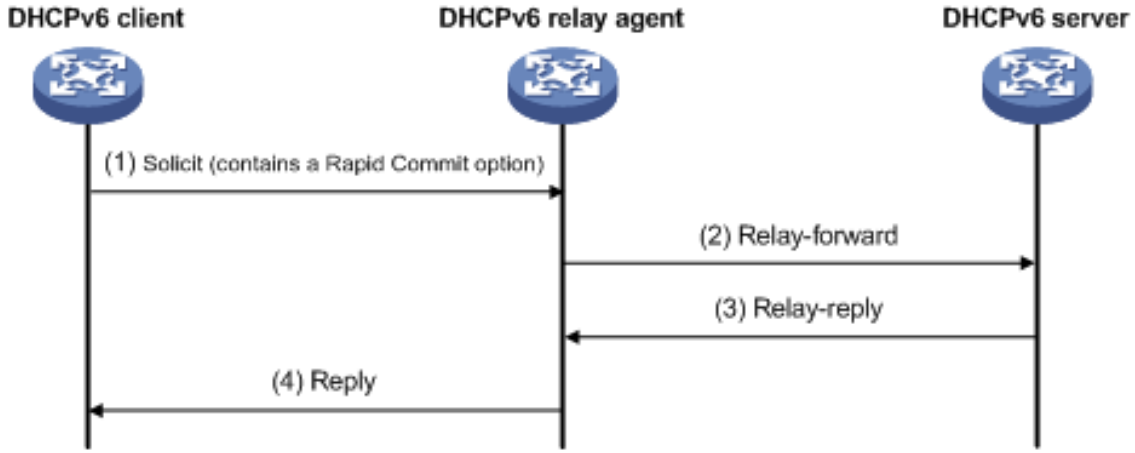
- Existing infrastructure: protocol, network, tools
- Servers already know how to manage and assign resources
- Protocol easily extensible
- Tomek and Bernie are in DHC WG co-chairs and …

DHCPv6 Overview (1/4)



Note: Client sends packets via link-local multicast (to FF02::1:2)
Servers unicast reply to client's link-local address

DHCPv6 Overview (2/4)



- Above adds relay agent and shows minimal 2 message exchange for initial assignment
- Client uses IPv6 link-local address
- Server or relay-agent must be present on link
- Relay-agents forward request to servers
- Return path is reverse of forward path
- **Client is not aware of relays**

DHCPv6 Overview (3/4)

- DHCPv6 uses minimal packet header with lots of possible options (TLV encoded)
- Options are IETF/IANA assigned
- Options can encapsulate other options
- Options for assigning resources
  - IA_NA & IAADDR for address assignment
  - IA_PD & IAPREFIX for prefix delegation
  - Assign multiple “resources” in single transaction

DHCPv6 Overview (4/4)

- Clients (and servers) identified by DUID
- 4 DUID types currently defined:
  - DUID-LL – based on link-layer address
  - DUID-LLT – based on link-layer address and time
  - DUID-EN – based on enterprise ID + data
  - DUID-UUID – based on UUID
- Server and clients must handle any DUID type

DHCPv6 Extensibility

- New options for DHCPv6 are relatively easily defined via IETF process
  - Write an individual submission draft
  - Get WG to adopt draft
  - Update draft to reach rough consenus
  - Get draft through WGLC & IESG review
  - IANA assigns option and RFC (Request for Comments) published

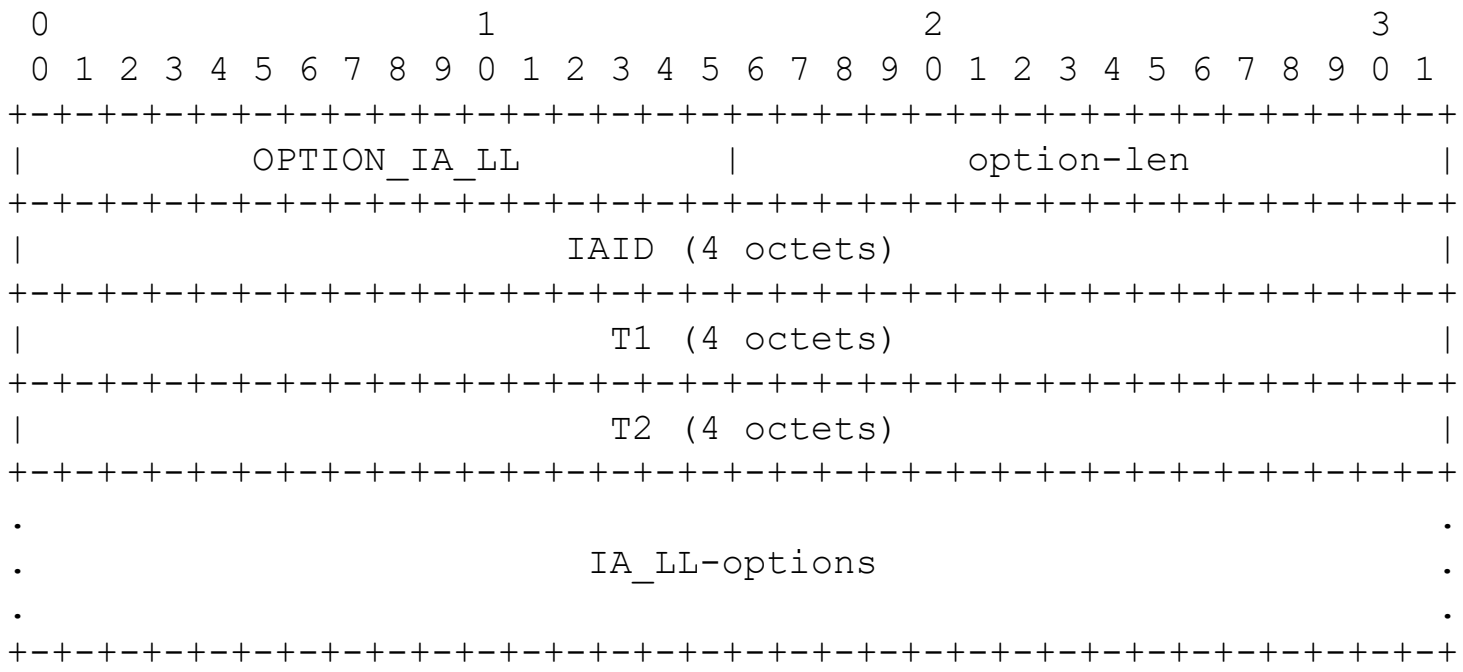
Hence the IETF Draft

- 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

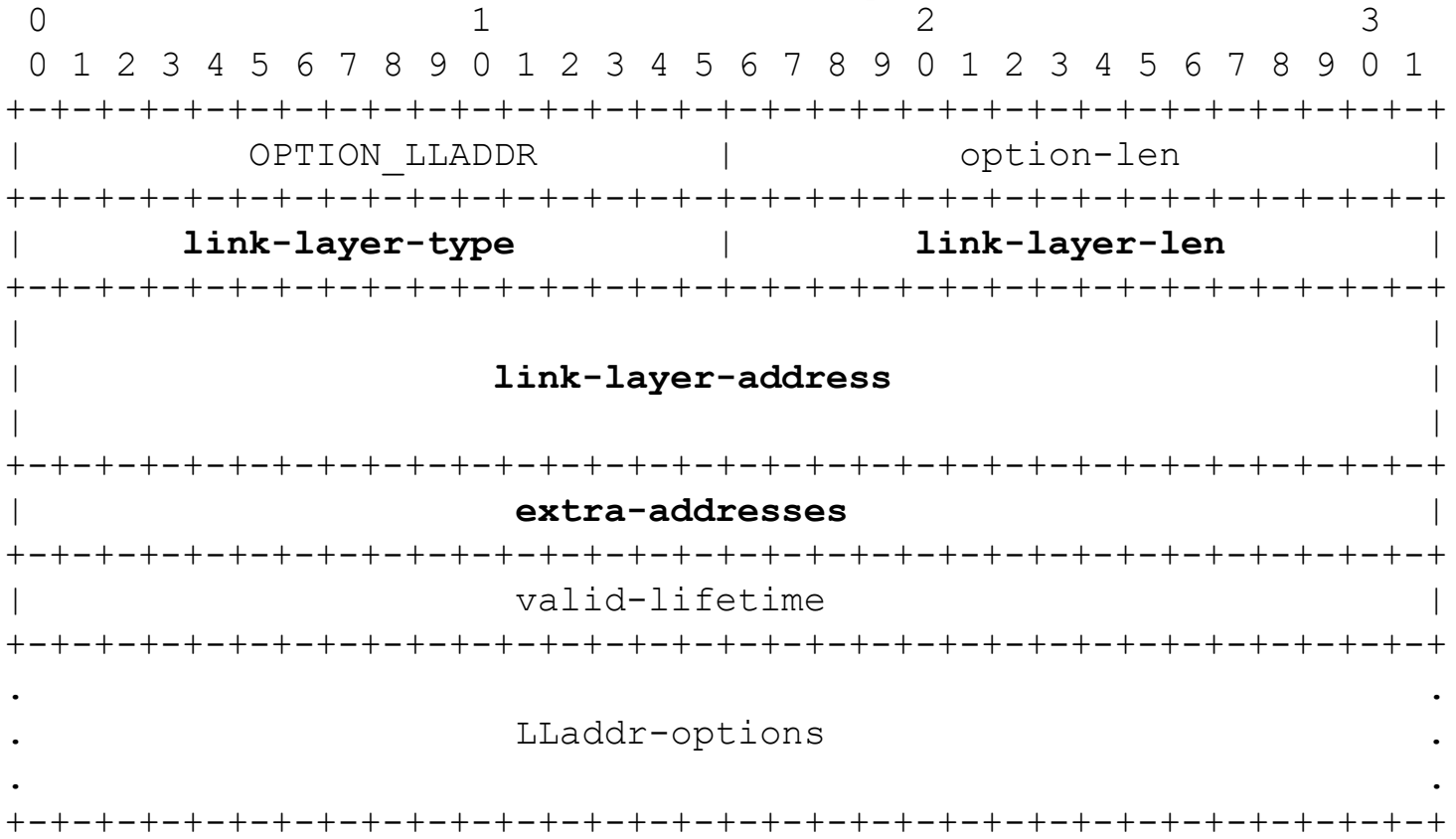
- IA_LL (Identity Association for Link Layer Addresses) Option
  - Similar to IA_NA and IA_PD
  - Used as container option for requested / assigned link-layer addresses
- LLADDR (Link Layer Addresses) Option
  - Similar to IAADDR and IAPREFIX
  - Used to request/assign link-layer addresses

IA_LL Option



- 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



- 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 “true” 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, …

draft-bvtm-dhc-mac-assign Status

- Currently an Individual Submission at IETF
  - Changes under author control
  - Targeted for DHC Working Group (WG)
- WG Adoption requires Adoption Call (authors will request shortly)
  - If adopted, becomes WG Draft
  - Changes under WG control (consensus)
- Currently no IPR claims against document

Next Steps

- Provide feedback to authors
  - On draft itself
    - send to the IETF DHC WG list (dhcwg@ietf.org) or email dhc-chairs@ietf.org
  - Help resolve open issues
    - Issue tracker
    - Also review closed issues to confirm action?
    - Raise new issues if needed
  - Indicate IEEE interest in this work
- Authors will request IETF DHC WG adoption of the draft before IETF-102 (Montreal, July 14-20, 2018)

Useful Links

- Draft text
  - https://datatracker.ietf.org/doc/draft-bvtm-dhc-mac-assign/
- Issue tracker for Draft
  - https://github.com/dhcwg/dhcp-mac/issues
- DHC mailing list (draft can be discussed here)
  - https://www.ietf.org/mailman/listinfo/dhcwg
- DHC WG
  - https://datatracker.ietf.org/wg/dhc

Question or comments …

&

THANKS