SLAP quadrant selection options for DHCPv6
draft-bernardos-dhc-slap-quadrant-01

CJ. Bernardos, A. Mourad

Prague, DHC WG, 2019-03-27

Background

- (Bernie summarized it already on his slides)
- IEEE 802c has defined a new "optional Structured Local Address Plan" (SLAP)
  - 4 regions (quadrants) in the local MAC address space, with different assignment approaches
    - **Quadrant "Extended Local Identifier" (ELI):** addresses are assigned based on a Company ID (CID)
      - 24 bits available
    - **Quadrant "Standard Assigned Identifier" (SAI):** addresses are assigned based on a protocol specified in an IEEE 802 standard
      - 44 bits are available
    - **Quadrant "Administratively Assigned Identifier" (AAI):** addresses are assigned locally by an administrator
      - 44 bits are available
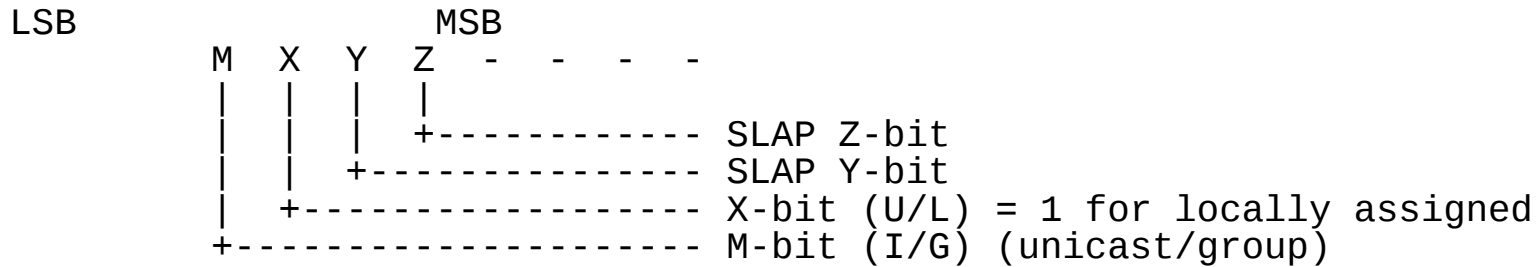   - **Quadrant "Reserved for future use"**

Background

```
LSB                    MSB
          M   X   Y   Z   -   -   -   -
          |   |   |   |
          |   |   |   +----------- SLAP Z-bit
          |   |   +--------------- SLAP Y-bit
          |   +------------------- X-bit (U/L) = 1 for locally assigned
          +----------------------- M-bit (I/G) (unicast/group)

            Figure 1: IEEE 48-bit MAC address structure
```

Quadrant	Y-bit	Z-bit	Local Identifier Type	Local Identifier
01	0	1	Extended Local	ELI
11	1	1	Standard Assigned	SAI
00	0	0	Administratively Assigned	AAI
10	1	0	Reserved	Reserved

Figure 2: SLAP quadrants

Motivation / Problem Statement

- The IEEE is working on mechanisms to allocate addresses in the SAI quadrant (IEEE 802.1CQ project)

- Some work also at the IETF: draft-bvtm-dhc-mac-assign specifies DHCPv6 extensions to handle the local MAC address assignments

- In this document, we complement ongoing IETF work with **mechanisms to allow choosing the SLAP quadrant** to use in the allocation of the MAC address to the requesting device/client
  - Why is this needed? Next slides…

SLAP quadrant selection: some scenarios

- WiFi terminals
 - Interfaces come with "burned in" MAC address (using OUI)
  - Now there is the need to assign local addresses:
    - IoT (Internet of Things): where there are a lot of cheap, sometimes short lived and disposable devices
      - Devices typically not moving
      - Any quadrant would be good but ELI/SAI quadrants might be more suitable in some scenarios (e.g. if it is needed that the addresses belong to the CID assigned to the IoT communication device vendor)
    - Privacy: issues can be mitigated by using & changing local random addresses
      - Devices typically mobile
      - AAI is probably the best quadrant, as it is best fit for randomization, and addresses are not required to survive after a change of network
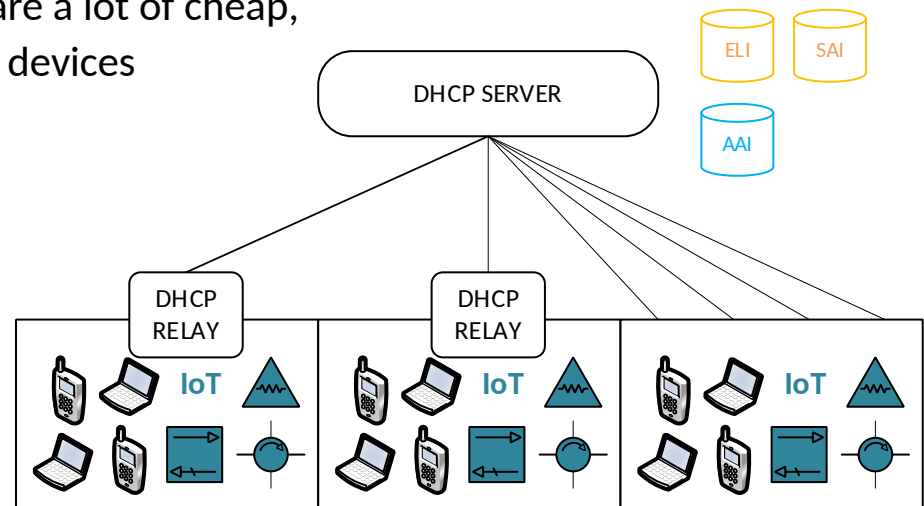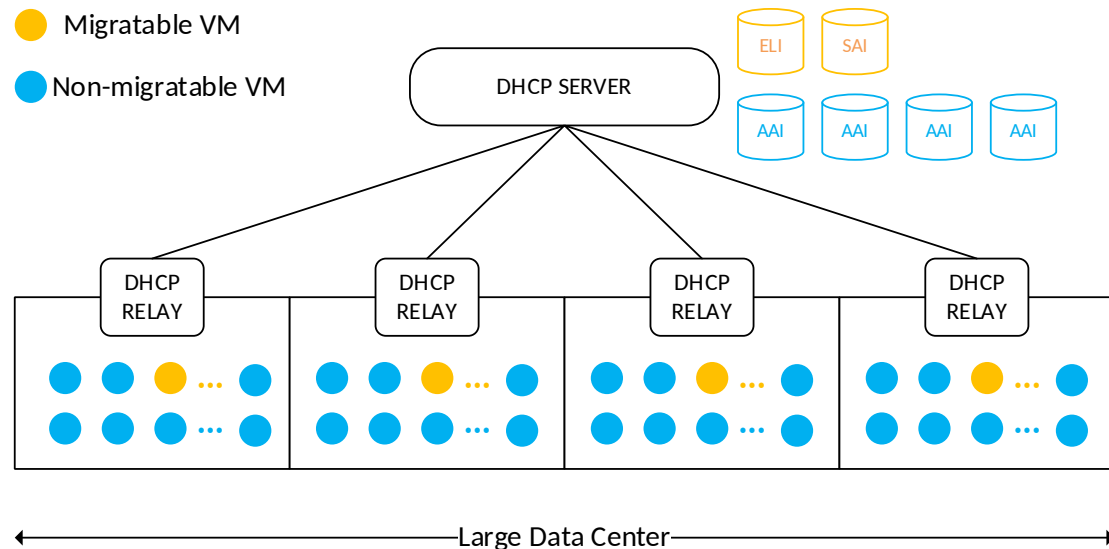
SLAP quadrant selection: some scenarios

- Hypervisor: migratable vs non-migratable functions
  - The hypervisor typically assigns addresses to VMs
  - Data centers may divide addressing space in regions. 2 situations:
    - Migratable functions:
      - VMs might be moved to a different data center
      - Context needs to be maintained
      - Devices typically not moving
      - ELI/SAI SLAP quadrants are more appropriate (can be centrally allocated by the server)
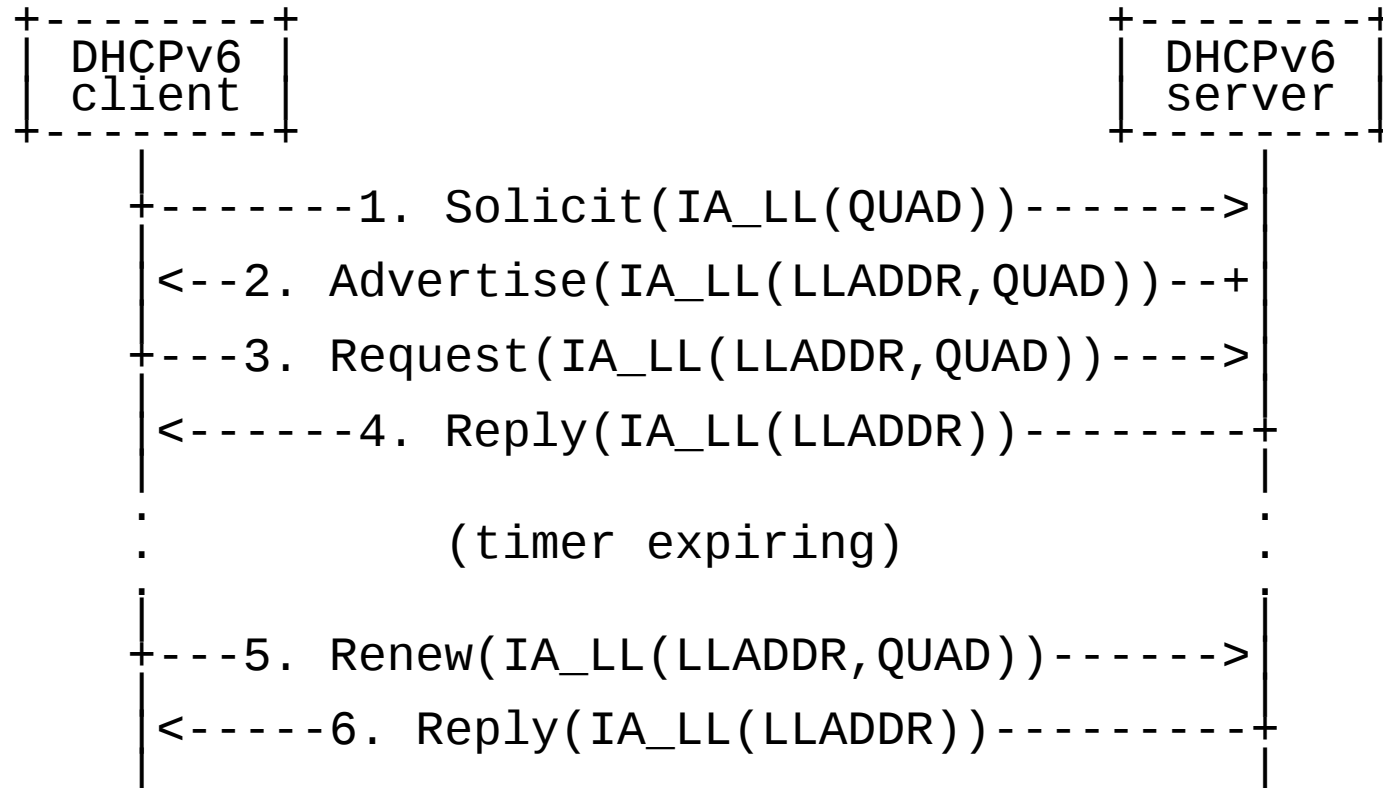   - Non-migratable functions:
      - VMs not moving
      - AAI SLAP quadrant is more suitable: each data center/region can use its own space without coordination to check for duplicates

SLAP quadrant selection mechanisms

- How to perform quadrant selection? Some examples
  - IoT scenario. Different parameters can be considered:
    - Type of IoT deployment: AAI for small deployments, ELI or SAI for large ones
    - Mobility: if mobile, SAI or AAI might be better to minimize address collisions
    - Managed/unmanaged
    - Operation/battery lifetime
  - WiFi device
    - Need for privacy based on context, profiles, app triggers, etc
  - Data center scenario
    - Hypervisor uses info from CMS/VIM: Migratable/Non-migratable VM, VM connectivity characteristics, etc.

SLAP quadrant selection: DHCPv6 extensions

```
+--------+                              +--------+
| DHCPv6 |                              | DHCPv6 |
| client |                              | server |
+--------+                              +--------+
    |                                       |
    +--------1. Solicit(IA_LL(QUAD))------->|
    |<--2. Advertise(IA_LL(LLADDR,QUAD))--+ |
    +---3. Request(IA_LL(LLADDR,QUAD))---->|
    |<------4. Reply(IA_LL(LLADDR))--------+
    |                                       |
    .             (timer expiring)          .
    .                                       .
    |                                       |
    +---5. Renew(IA_LL(LLADDR,QUAD))------>|
    |<-----6. Reply(IA_LL(LLADDR))--------+
    |                                       |
```
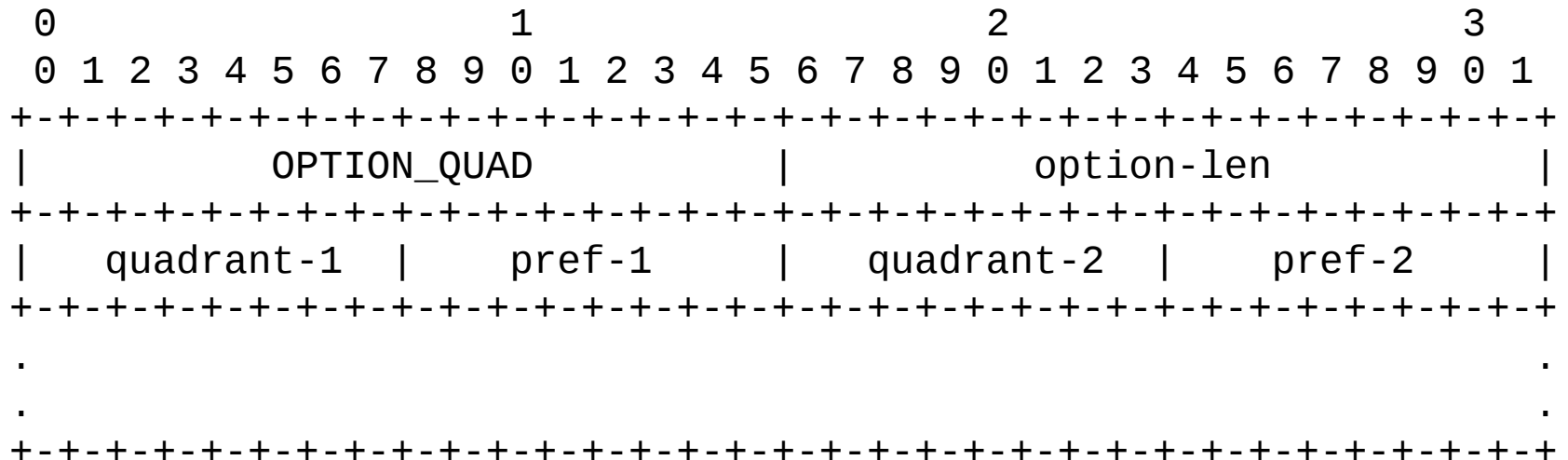
- SLAP quadrant indicated by the relay (client-relay-server signaling) also described in the draft

QUAD IA_LL option

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           OPTION_QUAD         |            option-len         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   quadrant-1  |    pref-1     |   quadrant-2  |    pref-2     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                                                               .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- option-len:2 * number of included (quadrant, preference)
- quadrant-n: Identifier of the quadrant
  (0: AAI, 1: ELI: 2, SAI: 3, 4: reserved)
- pref-n: Preference associated to quadrant-n

Next steps

- Is this work of interest to DHC?
  - Extensions to draft-bvtm-dhc-mac-assign
- Provide feedback to authors about draft