SLAAC (Stateless Address Autoconfiguration)

APNIC46

September, 2018

Noumea, New Caledonia

Jordi Palet

(jordi.palet@theipv6company.com)

Autoconfiguration (RFC4862)

- The document specifies the steps a host takes in deciding how to autoconfigure its interfaces in IPv6.
- The autoconfiguration process includes creating a link-local address and verifying its uniqueness on a link, determining what information should be autoconfigured (addresses, other information, or both), and in the case of addresses, whether they should be obtained through the stateless mechanism, the stateful mechanism, or both.
- IPv6 defines both a stateful and stateless address autoconfiguration mechanism.
- Stateless autoconfiguration requires no manual configuration of hosts, minimal (if any) configuration of routers, and no additional servers.

RA: Flags M y O

- RA M (Management) and O (Other) flags tell hosts how to configure network parameters
- M refers to the IP address configuration
- O stands for other parameters: DNS, etc.

Address / Others	M	O	Remarks
SLAAC / SLAAC	0	0	If dual-stack, IPv4 can be used for DNS
SLAAC / DHCPv6	0	1	DHCPv6 Stateless
DHCPv6 / SLAAC	1	0	If dual-stack, IPv4 can be used DNS
DHCPv6 / DHCPv6	1	1	Gateway always learnt by RA

Stateless or Serverless Autoconfiguration (RFC4862)

- Stateless mechanism allows a host to generate its own addresses using a combination of locally available information and information advertised by routers.
- Routers advertise prefixes that identify the subnet(s) associated with a link.
- Hosts generate an "interface identifier" that uniquely identifies an interface on a subnet, locally generated, e.g., using MAC address.
- An address is formed by combining the both.
- In the absence of routers, a host can only generate link-local addresses.
- Link-local addresses are sufficient for allowing communication among nodes attached to the same link.

Advantages of SLAAC

- Manual configuration of each host before getting connected, not required
- Small sites, with a single subnet, don’t need a DHCPv6 server, and even a router to get connected together (use link-local addresses)
- A bigger site with multiple subnets, don’t need a DHCPv6 for the address configuration
- Facilitates the prefix change (renumbering) by means of multiple addresses per interface and life time

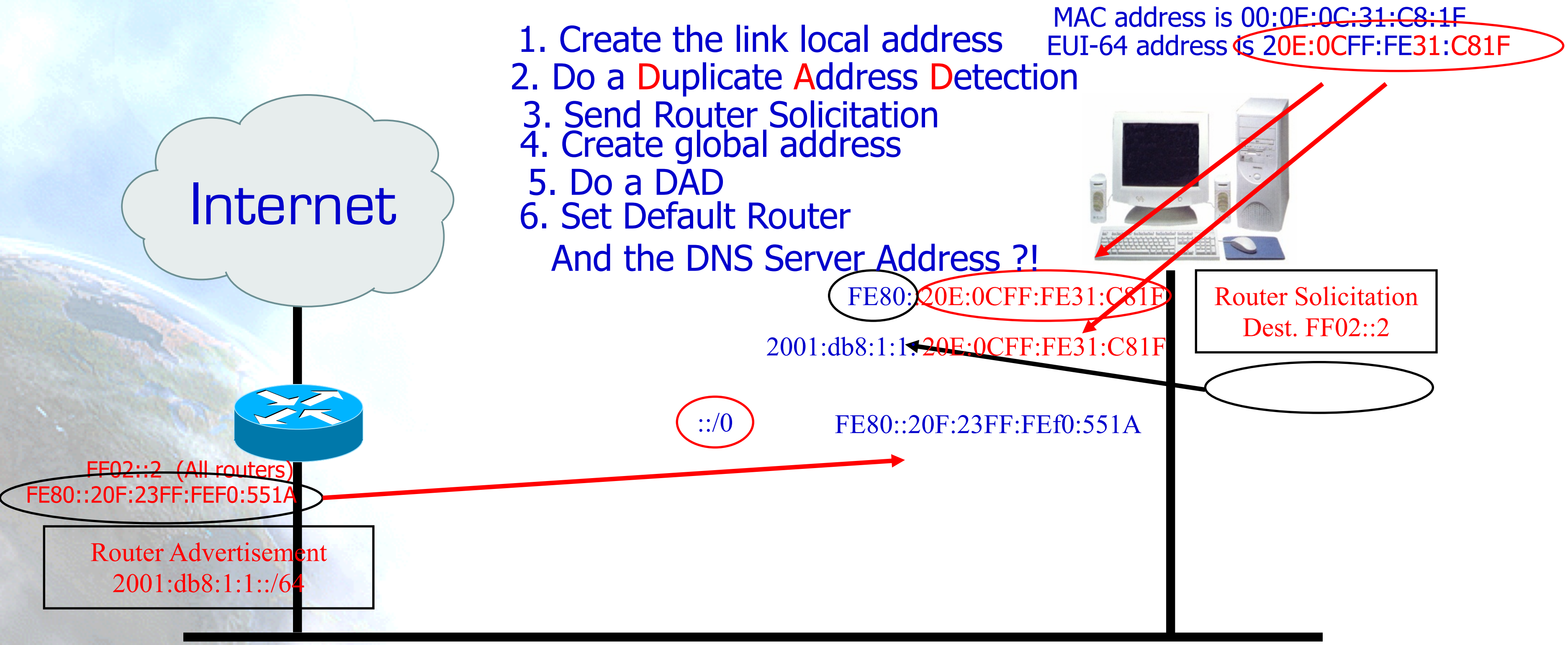
Address Life Time

- IPv6 addresses are leased to an interface for a fixed (possibly infinite) length of time, that indicates how long the address is bound to an interface.
- When a lifetime expires, the binding (and address) become invalid and the address may be reassigned to another interface elsewhere in the Internet.
- To handle the expiration of address bindings gracefully, an address goes through two distinct phases while assigned to an interface.
  - Initially, an address is "preferred", meaning that its use in arbitrary communication is unrestricted.
  - Later, an address becomes "deprecated" in anticipation that its current interface binding will become invalid.

Duplicate Address Detection

- To ensure that all configured addresses are likely to be unique on a given link, nodes run a "duplicate address detection" algorithm on addresses before assigning them to an interface.
- The Duplicate Address Detection algorithm is performed on all addresses, independent of whether they are obtained via stateless or stateful autoconfiguration.
- The procedure for detecting duplicate addresses uses Neighbor Solicitation and Advertisement messages.
- Since host autoconfiguration uses information advertised by routers, routers will need to be configured by some other means. However, it is expected that routers will generate link-local addresses using the same mechanism.
- Routers are expected to successfully pass the Duplicate Address Detection procedure on all addresses prior to assigning them to an interface.

SLAAC



1. Create the link local address
2. Do a Duplicate Address Detection
3. Send Router Solicitation
4. Create global address
5. Do a DAD
6. Set Default Router

And the DNS Server Address ?!

MAC address is 00:0E:0C:31:C8:1F
EUI-64 address is 20E:0CFF:FE31:C81F

DNS Server Configuration (1)

- Originally, two choices:
  - Manual configuration
  - Using DHCPv6 or DHCPv4 (if dual-stack)
- It may be a problem in big sites:
  - Need to use two protocols (SLAAC and DHCPv6)
  - Delay to get the DNS server if using DHCP
  - In wireless, the device may change network frequently, so manual configuration is not an option, or the DHCP delay is not acceptable
- A new mechanism was designed:
  - RFC8106, Recursive DNS Server (RDNSS), using RA
  - It can be used in parallel with DHCPv6

DNS Server Configuration (2)

- Same mechanism as prefixes/routers are learnt with ND: IPv6 Stateless Address Autoconfiguration [RFC4862]
- With RDNSS the node learns, with a single message:
  - Prefix for the autoconfiguration
  - Gateway
  - Recursive DNS Servers
- If, in addition of RDNSS, DHCPv6 is uses, the “O” flag must be set in the RA
- Two options to configure RDNSS option in routers:
  - Manually
  - Automatically, if they are DHCPv6 client

SLAAC Host Configuration

- Windows
  - GUI
  - netsh interface ipv6

- Linux
  - GUI
  - /etc/network/interfaces

    ```
    iface eth0 inet6 auto
    ```

  - CLI

    ```
    ip -6 addr add <ipv6address>/<prefixlength> dev <interface>
    ifconfig <interface> inet6 add <ipv6address>/<prefixlength>
    ```

SLAAC Router Configuration

```
interface FastEthernet0/0
        ! RA enabled by default with and IPv6 address
        ! To disable
        ipv6 nd ra supress
        ! Otros comands
        ipv6 nd ra (mtu | interval | lifetime)
        ! Flags M and O
        ipv6 nd managed-config-flag
        ipv6 nd other-config-flag
        ! Prefix announced
        ipv6 nd prefix X:X:X:X::X/64
        !RDNSS
        ipv6 nd ra dns server X:X:X:X::X
```

RADVD Linux Configuration

- Config file at /etc/radvd

```
interface br-lan {
    AdvSendAdvert on ;
    #UnicastOnly on ;
    # Advertise at least every 30 seconds
    MaxRtrAdvInterval 30;
    # in order to force non RFC 6106 compliant clients to get a dns address
    AdvOtherConfigFlag on ;
    AdvManagedFlag on;
    prefix 2001:470:68ee:30::/64 {
      AdvOnLink on;
      AdvAutonomous on;
    };
    RDNSS 2001:470:68ee:30::30 {
    };
};

service radvd start | restart | stop | status
```

Thanks !!

Contact:

- Jordi Palet (The IPv6 Company): jordi.palet@theipv6company.com