DHCPv6/SLAAC Address Configuration Interaction Problem Statement

(draft-liu-bonica-v6ops-dhcpv6-slaac-problem)

Bing Liu(speaker), Ronald Bonica
Xiangyang Gong, Wendong Wang
IETF 88@Vancouver, Nov 2013
Regarding the Mailing List Discussion

• ML discussions was regarding an overview of DHC/ND cooperation topic
  – Some people love DHC-only (follow the IPv4 style)
  – Some prefer ND (think about light bulbs, sensors...)
  – What is the boundary between DHC and ND
  – How they could serve respectively/efficiently
  – ...

• This is an important/fundamental topic might need to be worked out
  – [draft-yourtchenko-ra-dhcppv6-comparison]

• But for this draft, we’re currently focusing on an existing specific problem of address configuration
About this Draft

• We already have two (automatic) address configuration mechanisms
  – DHCPv6 and SLAAC
  – They would be probably co-exist in one network
• And they are correlated by several defined flags
  – A flag, M flag, O flag, defined in ND protocol
• But the behaviors of interpreting the flags are ambiguous
  – Ambiguity might be a problem for OAM.
  – 6man tried to clear the ambiguity before, but failed.
• This draft aims to:
  – identify operational problems caused by ambiguity; provide cautions to operators/administrators
  – might consequently promote re-work on standard revision to fix the problems
The Flags

• ND RA messages include the following flags
  - “Autonomous Flag”: indicates that a prefix can be used for SLAAC (included in the Prefix Information Option)
  - “Managed Flag”: indicates that addresses are available via DHCPv6
  - “OtherConfig Flag”: indicates that other configuration information (DNS .etc) is available via DHCPv6

• Neither [RFC4861] nor [RFC4862] completely specifies the host behavior when interpreting these flags
Hosts might confuse about...

• **Is there any dependency between the two mechanisms?**
  – E.g. Do I need to see M=1 to initiate DHCPv6? If there are no RAs at all, should I initiate DHCPv6 by myself?

• **Should I interpret these flags as advisory or prescriptive?**
  – E.g. when M flag set, MUST I initiate DHCPv6 or might be according to other factors?
  – Especially when flags are in transition
    • E.g. I’m already SLAAC-configured, should I still care about the M flag changed?

• **Relationship between “Address Configuring Method” and “Address Lifetime”**
  – When method changes, should I immediately release the addresses or just wait them expired

• **Is there any dependency between the flags?**
  – one flag is set or not, would behavior of other flags be impacted?
Testing

• We tested various operating system’s handling of these flags:
  – Windows 7
  – Linux (Ubuntu 12.10)
  – Mac OS X (10.7)
  – iOS (6.1.3, iPod Touch4)
  – *Android (4.0.4, HTC Incredible S)

(*Android lacks support of DHCPv6 so far.)
Important test results

A flag behaviors
- For SLAAC-configured hosts, when A changed from 1 to 0, Win7 deprecated SLAAC while Linux/MAC/iOS ignored the RA messages. ("Address Configuring Method" vs "Address Lifetime")

M flag behaviors
- Linux/MAC only start DHCPv6 until receive RA with M=1 (dependency between the two mechanisms)
- SLAAC-configured hosts receiving RA with M=1, Win7 does DHCPv6, Linux/MAC don’t (advisory or prescriptive)
- DHCPv6-configured hosts receiving RA with M=0, Win7 release DHCPv6 addresses, Linux/MAC doesn’t ("Address Configuring Method" vs "Address Lifetime")

O flag behaviors
- O is not independent with M. When M set, O is implicitly set as well (This is reasonable)
- Linux/MAC won't initiate stateless DHCPv6 when A flag is NOT set; Win7/iOS would (dependency between the flags)
- O=1, then M from 1 to 0 or vice versa, Win7 would switch to stateless DHCPv6 or statefull DHCPv6; Linux/MAC/iOS no action (advisory or prescriptive)
Operational Issues

• Given inconsistent host behavior, it is difficult for network managers to predict and control host addressing

• In the case of renumbering
  - Renumbering exercise may require transition from SLAAC to DHCP or vice versa. [RFC7010]

• In the case of cold start
  - “This makes it difficult for a site network manager to configure systems in such a way that all hosts boot in a consistent way.” [RFC5887]

• In the nominal cases
  - network wants hosts to do DHCPv6-only configuration
  - the hosts have been SLAAC-configured, then the network need the hosts to do DHCPv6 simultaneously (e.g. for multihoming)
  - the network wants the hosts to do statelessDHCPV6-only; for example, the hosts are configured with self-generated addresses (e.g. ULA), and they also need to contact the DHCPv6 server for info-configuration
Question

• Do operators/administrators care about the problems?
• Adopted by v6ops as a Problem Statement?
Comments?

Thank you

leo.liubing@huawei.com
rbonica@juniper.net
xygong@bupt.edu.cn
wdwang@bupt.edu.cn
IETF88, Vancouver
Backup Slides
Test Case 1: Initial behavior

- Host from non-configured to configured, we tested different A/M/O combinations in each OS platform. The configured states are enumerated as the following:
  - SLAAC only
  - SLAAC+Stateless DHCPv6 (info-request other-info than addresses)
  - SLAAC+Stateful DHCPv6 (address and other-info(if available) together)
  - Stateful DHCPv6 only (address and other-info(if available) together)
  - Stateless DHCPv6 only （only available in Windows 7）

- Following slides illustrate state graphs of the OS platforms respectively
Windows 7/iOS (6.1.3)

- SLAAC
- SLAAC+ DHCPv6
- DHCPv6
- DHCPv6 info-request only
- SLAAC+ DHCPv6 info-request only
- Non-config

A=0, M=0, O=0
A=1, M=1, O=0
A=0, M=1, O=1
A=0, M=0, O=1
A=1, M=0, O=1
Linux (Ubuntu 12.10) / Mac OS X (10.7)
Test Case1 Summary

- **A** is interpreted as prescript in each OS
- **M** is interpreted as prescript in each OS at the initial state, for controlling DHCPv6
- **A** and **M** are independent
- **A** and **O** are not totally independent. In Linux and Mac, A=1 is required for O=1 triggering DHCPv6 info-request; not applicable in Windows 7
- **M** and **O** are not totally independent. M=1 has priority than O=1 (when M=1 O=1, the system will configure adderess and other-info together, rather than respectively)
Test Case2: SLAAC/DHCPv6 Switching

**SLAAC only —> DHCPv6 only** *(SLAAC-only host receiving RA with A=0 M=1)*

- Windows 7: abandon SLAAC, initiate DHCPv6, successfully switched
- Linux /Mac OS X/iOS: keep SLAAC, don’t initiate DHCPv6 unless SLAAC is expired and no continuous RA

**DHCPv6 only —> SLAAC only** *(DHCPv6-only host receiving A=1 M=0)*

- Windows 7: config SLAAC, release DHCPv6,
- Linux: config SLAAC, keep DHCPv6 and keep renewing
- Mac OS X/iOS: config SLAAC, keep DHCPv6 and don’t renew
Test Case 3: Stateful/Stateless DHCPv6 Switching

**Stateless—> Stateful** (M changes from 0 to 1; keep A=O=1)

- Windows 7: initiates statefull DHCPv6, configures DHCPv6 address as well as re-configure other-info
- Linux/Mac OS X/iOS: no action

**Stateful—> Stateless** (M changes from 1 to 0; keep A=O=1)

- Windows 7: releases all DHCPv6 config including address and other-info, initiates stateless DHCPv6 information-request/reply
- Linux/Mac OS X/iOS: no action, renew when expired
END