SAVI
Internet Draft
Intended status: Standard Tracks
Expires: May 2011

Jun Bi
CERNET
Guang Yao
Tsinghua Univ.
Joel M. Halpern
Newbridge Networks Inc.
Eric Levy-Abegnoli
Cisco System

November 8, 2010

SAVI for Mixed Scenario
draft-bi-savi-mix-02.txt

Status of this Memo

Copyright Notice

Abstract

This document specifies the procedure a SAVI device resolves conflict from multiple co-existing SAVI solutions.

Table of Contents

1. Introduction

SAVI solutions are specified for scenarios allowing only single address assignment method without considering the co-existing of multiple address assignment methods. In practice, for both IPv4 and IPv6 network, generally multiple address assignment methods are allowed. Current SAVI solutions cannot be used directly in such scenarios, because collision between solutions may happen. This document specifies the possible collisions and proposes corresponding mechanism to solve the collisions.

2. Terminology

Address Assignment Source (AAS): The de facto entity type that assigns address.

3. Mixed Scenario

Currently, there are actually four SAVI solutions which cover different types of addresses:

(1) SAVI-FCFS: SLAAC

(2) SAVI-DHCP: stateful DHCP, static DHCP

(3) SAVI-SEND: CGA with certificate, CGA without certificate

(4) Manually configuration: static address manually configured by administrator on SAVI device statically. Note: address configured by user on host is treated as stateless address.

A practical network may enable any combination of address assignment methods, and all the corresponding solutions should be enabled to avoid legitimate packet filtering. If more than one SAVI solution is enabled on SAVI device, the scenario is named as mix scenario in this document.

4. Basic SAVI-MixMode Structure

Existing SAVI solutions are individual mechanism without considering inter-cooperation. To keep the independence and completeness of each solution, a SAVI solution is treated as a black box which snoops packet and generates/removes candidate binding, without concerning the inner structure of each solution.

Because the binding entry setup by each solution is ALLOW entry, thus a solution will reject any address not bound by it. However,

address bound by any solution must be allowed if no collision, thus, binding entry table should be shared by all the solutions. The main work of this document is handling the conflict resulted from solutions sharing the binding table.

If bindings on different addresses are set up by different solutions, no collisions can happen. Thus, a guideline here is to separate the address space of each type to avoid any kind of collision. However, if there is overlap between address spaces, bindings on the same address can be set up by different solutions, and collision can happen.

5. Problem Scope and Statement

5.1. Problem Scope

This document is specified for collision between SAVI solutions. The situation that collision happens in a single solution, for example, the same address is bound by the same solution on different binding anchors, is not in the scope of this document.

SAVI solutions mainly specify the setup and remove of bindings. Whenever a solution sets up a binding or removes an existing binding, it may violate the state of other solutions. In the mix mode, the SAVI device must decide whether to accept the operation request from each solution or not.

5.2. Collision in Binding Set-up Procedure

In binding set up, collision happens when:

(1) the same address

(2) different binding anchors on the SAVI perimeter and

(3) different binding solutions.

As an instance, after an address is bound on one binding anchor by DHCP solution, the FCFS solution requires to bind the address on another binding anchor. Both bindings are legitimate in corresponding solution; however, only one of the bindings should be allowed. Then the SAVI device must decide whom the address should be bound with.

NOTE: because a single SAVI device doesn't have the information of all bound addresses on the perimeter, a collision may not be explicit based only on local bindings. To make the perimeter-scope

collision explicit to each SAVI device, which means, a SAVI device must distinguish whether a local binding setup request violate a binding on other devices or not.

Following mechanism can be used:

(1) SAVI device performs DAD proxy for local manually configured address even if the node with static address is off-link;(Or to manually configure all the SAVI devices is also proposed.)

Then the collision that SAVI-FCFS request static address can be handled.

(2) Static address must be excluded from DHCP address pool;

Then the collision that SAVI-DHCP request static address can be handled.

(3) SAVI device performs DAD proxy for local DHCP address.

Then the collision that SAVI-FCFS request DHCP address on other SAVI devices can be handled.

5.2.1. Proposed solution

To make a choice between candidate bindings, a preference level based solution is thought to be efficient from the experience of similar implementations.

The essential problem is: 1. The granularity of preference level; 2. The basis of preference level (or at least the default level).

The preference level proposed in this document is an AAS (Address Assignment Source) granularity preference level. And preference level is assigned based on the trustworthy of AAS and the sequence of candidate bindings.

By now, there are 4 types of AAS:

(1) Node itself: SLAAC, CGA without certificate

(2) DHCP sever: stateful DHCP address

(3) PKI: CGA with certificate, plain address with certificate

(4) Administrator: static address, static DHCP address(may not be taken into consideration as no standard document)

Combined with binding sequence, there will be 16 scenarios:

FORMER	LARER	PREFERENCE
Node	Node	In the scope of SAVI-SLAAC
Node	DHCP	Switch here: either former or later
Node	PKI	Later
Node	Admin	Later
DHCP	Node	Former
DHCP	DHCP	In the scope of SAVI-DHCP
DHCP	PKI	Later
DHCP	Admin	Later
PKI	Node	Former
PKI	DHCP	Former
PKI	PKI	No definition
PKI	Admin	Later
Admin	Node	Former
Admin	DHCP	Former
Admin	PKI	Former
Admin	Admin	Later(Or not in scope of this document)

If ignoring the details, the basic preference level of AAS is simply node<DHCP<PKI<Admin, with only one exception in permutation (Node, DHCP).

DISSCUSSION:

We have considered some other possible granularities: (1) solution level; (2) binding parameter level.

Solution level granularity is most suitable based on the current structure of workings. The problem is the preference level of an assignment method may not be unique. For example, binding set up by SAVI-SEND may either with a certificate or not. Apparently, they should have different preference level.

Another measurement takes binding parameter into consideration, as proposed in [draft-levy-abegnoli-savi-plbt-02]. Other than the binding set up solution, also port type (access, trunk, trusted access, trusted trunk), link layer information, etc., are considered to affect the preference level. The problem is that how to compare the preference level of factors with different characteristics. This means it is hard to design a convincing preference level. Also, because bindings are not setup on trust port, trust port factors are not of value.

5.3. Collision in Binding Removal

A binding may be set up on the same binding anchor by multiple solutions. Generally, the binding lifetimes of different solutions are different. Potentially, if one solution requires to remove the binding, the node using the address may be taken the use right.

For example, a node performs DAD procedure after being assigned an address from DHCP, then the address will also be bound by SAVI-FCFS. If the SAVI-FCFS lifetime is shorter than DHCP lifetime, when the SAVI-FCFS lifetime expires, it will request to remove the binding. If the binding is removed, the node will not be able to use the address even the DHCP lease time doesn't expire.

The solution proposed is to keep a binding as long as possible. A binding is kept until it has been required to be removed by all the solutions that ever set up it.

6. Security Considerations

No security consideration currently.

7. IANA Considerations

No IANA consideration.

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9. Informative References

[savi-framework] Wu, J., Bi, J., Bagnulo, M., Baker, F., and Vogt, C., "Source Address Validation Improvement Framework", draft-ietf-savi-framework-01.

[savi-dhcp] Bi, J., Wu, J., Yao, G., and F. Baker, "SAVI Solution for DHCPv4/v6", draft-ietf-savi-dhcp-06.

[savi-fcfs] Nordmark, E., Bagnulo, M., and E. Levy-Abegnoli, "First-Come First-Serve Source-Address Validation Implementation", draft-ietf-savi-fcfs-05.

[savi-send]Bagnulo, M. and A. Garcia-Martinez, "SEND-based Source-Address Validation Implementation", draft-ietf-savi-send-04.

[eric-plbt] E. Levy-Abegnoli, "Preference Level based Binding Table", draft-levy-abegnoli-savi-plbt-02.txt.

10. Acknowledgments

Authors' Addresses

Jun Bi
CERNET
Network Research Center, Tsinghua University
Beijing 100084
China
Email: junbi@cernet.edu.cn

Guang Yao
Network Research Center, Tsinghua University
Beijing 100084, China
Email: yaog@netarchlab.tsinghua.edu.cn

Joel M. Halpern
Newbridge Networks Inc.
Email: jmh@joelhalpern.com

E. Levy-Abegnoli
Cisco System
Village d'Entreprises Green Side - 400, Avenue Roumanille
Biot-Sophia Antipolis - 06410
France
Email: elevyabe@cisco.com