Internet Area Working Group (intarea)
Internet-Draft
Intended status: Standards Track
Expires: 29 November 2025

M. Xu
J. Wu
Tsinghua University
T. Lin
New H3C Technologies Co. Ltd
L. He
Y. Wang
Tsinghua University

28 May 2025

A SAVI Solution for WLAN
draft-bi-intarea-savi-wlan-05

Abstract

This document describes a source address validation solution for WLANs where 802.11i or other security mechanisms are enabled to secure MAC addresses. This mechanism snoops NDP and DHCP packets to bind IP addresses to MAC addresses, and relies on the security of MAC addresses guaranteed by 802.11i or other mechanisms to filter IP spoofing packets. It can work in the special situations described in the charter of SAVI (Source Address Validation Improvements) workgroup, such as multiple MAC addresses on one interface. This document describes three different deployment scenarios, with solutions for migration of binding entries when hosts move from one access point to another.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 29 November 2025.

Table of Contents

1. Introduction

Source spoofing poses a significant threat to the Internet. Therefore, source address validation is of great importance for enhancing network security. The IETF has standardized source address validation solutions for different address assignment scenarios, such as SLAAC [RFC6620], DHCP [RFC7513], SEND [RFC7219], and mixed scenarios [RFC8074].

However, the source address validation schemes standardized by the IETF primarily consider wired local area networks (LAN) scenarios and lack consideration for wireless LANs (WLANs) scenarios. In wired LANs, a host typically connect to a switch interface, and at this point, the switch interface corresponds to a binding anchor that can be used to verify the legitimacy of the host’s IP address. As discussed in [RFC7039], a "binding anchor" is an immutable or hard-to-change attribute that can be used to identify the system to which an IP address is assigned. In contrast, in wireless networks, all hosts connect to the wireless radio frequency of an access point (AP). Therefore, in wireless LANs, there is no binding anchor like a switch port that is naturally available. Moreover, hosts in wireless LANs can move frequently, and during movement, the host may maintain the same address, which means that the binding entries for the same host also need to be synchronized between different APs or even access controllers (ACs). This is a problem that existing solutions cannot solve.

This document describes a mechanism for performing per-packet IP source address validation in WLANs. This mechanism follows the source address validation improvement (SAVI) model proposed in [RFC7039] to implement source address validation. The mechanism first performs ND snooping or DHCP snooping to bind an assigned IP address to a verified MAC address (the binding anchor in this scenario). Therefore, the MAC address should be secured by 802.11i or other mechanisms. Static addresses are manually bound to the MAC address of the corresponding host. The mechanism can then check the validity of the source IP address in the local packet against the binding association. This fine-grained source address validation mechanism applies to both IPv4 and IPv6 packets.

This mechanism utilizes two important data structures, the IP-MAC mapping table on the control plane and the MAC-IP mapping table on the data plane, to implement source address validation, which is described in detail in this document.

The case of an interface with multiple MAC addresses is a special case mentioned in the SAVI charter and is the only special case that challenges MAC-IP binding. The mechanism to handle this case is specified in the document.

Three deployment scenarios for this mechanism are specified in this document, describing the devices and details of deployment in different scenarios.

When a host moves from one access point to another, the migration of binding entries can be triggered depending on the specific mobility scenario. The mechanism for handling host mobility is specified in the documentation based on different deployment scenarios.

1.1. Terminology

Access Point: The logical entity that provides access to the distribution services.

Lightweight Access Point: The access points used in centralized WLAN architecture.

Autonomous Access Point: The access points used in autonomous WLAN architecture.

Access Controller: The network entity in the centralized WLAN architecture that manages wireless network access points that allow wireless devices to connect to the network.

Binding anchor: A "binding anchor" is defined to be a physical and/or link-layer property of an attached device, as defined in [RFC7039]. In this document, the binding anchor refers to th MAC address.

Binding entry: A rule that associates an IP address with a binding anchor.

It is assumed to be familiar with SAVI-FCFS and SAVI-DHCP and their terms as defined in [RFC6620] and [RFC7513], respectively.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. IP-MAC Binding

This section specifies the operations for creating and clearing bindings between IP addresses/prefixes and MAC addresses. The reason for considering the binding of IP prefixes to MAC addresses is that there exists a need to issue prefixes for nodes in a broadcast network [I-D.draft-ietf-v6ops-dhcp-pd-per-device].

3.1. Data Structures

The bindings between IP addresses/prefixes and MAC address are stored using two data structures, i.e., the IP-MAC mapping table and MAC-IP mapping table.

3.1.1. IP-MAC Mapping Table

This table maps IP addresses/prefixes to their corresponding MAC addresses. The IP address/prefix is the index of the table. An IP address/prefix can have only one corresponding MAC address. Different IP addresses/prefixes can be mapped to the same MAC address.

This table is used in the control process. Before creating a new IP-MAC binding, this table must be queried to prevent conflicting binding entries. Also, this table must be queried before any packet filtering is performed. This table must be synchronized with the MAC-IP mapping table specified in Section 3.1.2.

Each entry in the IP-MAC mapping table must also record the binding method of the IP address/prefix. Addresses snooped in the DHCP address assignment procedure must have their binding method recorded as "DHCP", and addresses snooped in the Duplicate Address Detection procedure [RFC4862] must have their binding method recorded as "SLAAC". IPv6 prefixes snooped in the DHCP prefix delegation procedure must have their binding method recorded as "DHCP-PD".

3.1.2. MAC-IP Mapping Table

This table maps MAC addresses to the corresponding IP addresses/ prefixes. The MAC address is the index of the table. It is a one-to-many mapping table, which means a MAC address can be mapped to multiple IP addresses/prefixes. Although multiple MAC addresses may exist on one interface, these MAC addresses must be mapped to different IP addresses/prefixes.

This table is used for filtering. Different from wired networks, the MAC-IP mapping table and the IP-MAC mapping table can be maintained separately on different devices. A synchronization mechanism must be

used between these two tables to ensure the consistency of the bindings. We will explain the details in Section 5 for different deployment scenarios.

3.2. Pre-conditions for Binding

As specified in [RFC7039], in a binding-based mechanism, the security of IP address is dependent on the security of the binding anchor. In WLANs, 802.11i or other link-layer security mechanisms make MAC address a strong enough binding anchor.

If the MAC address is unprotected, an attacker can spoof the MAC address to pass validation successfully.

3.3. Binding IP addresses to MAC addresses

All the static IP-MAC address pairs are configured into the IP-MAC mapping table with the mechanism enabled.

A separate procedure handles the binding of addresses/prefixes assigned by DHCP to MAC addresses. This procedure snoops on the DHCP address assignment or prefix delegation process between the attached host and the DHCP server. DHCP snooping in WLANs is the same as that in wired networks specified in [RFC7513]. Note that if a client registers a self-generated address with a DHCPv6 server, the SAVI device also needs to establish a binding for that address by listening to the registration messages [I-D.draft-ietf-dhc-addr-notification]. A first come, first served model can be considered to establish a binding for the first registered address with its corresponding MAC.

A separate procedure handles the binding of stateless addresses to MAC addresses. This procedure snoops Duplicate Address Detection (DAD) procedure as described in [RFC4862] or Address Resolution procedure between attached hosts and neighbors as described in [RFC4861]. Based on the principle of roaming experience first in WLAN, the new binding anchor is selected in preference and triggers the deletion of the secure connection of the old binding anchor.

In some deployment scenarios, the functions of address snooping and IP-MAC mapping table maintenance may also be separated to different devices. Therefore, to prevent conflicting binding entries, the device for address snooping must interact with the device that maintains the IP-MAC mapping table. We will specify the details in Section 5.1.1.

Note that there are various ways to generate interface identifiers for IPv6 addresses [RFC7217][RFC8981][RFC8064], but there is no difference for creating binding tables and performing source address verification. As mentioned above, a SAVI device only cares about DHCP messages and ICMP messages associated with stateless addresses when establishing binding table entries. In addition, since an IPv6 interface may have multiple IPv6 addresses, it is necessary to bind each IP address of that IPv6 interface to the corresponding MAC address. However, this may also pose security issues, which we will discuss in Section 7.

3.4. Binding Migration

Different from wired networks, SAVI for WLAN must handle the migration of binding entries when a mobile host moves from one access point to another. After the move, the host will not perform another address configuration procedure to obtain new IP addresses but continue to use the existing IP address(es)/prefixes. Thus, binding entries in the foreign device accessed by mobile hosts cannot be established by snooping. A new mechanism is needed to correctly migrate the binding entry associated with the mobile host’s IP address/prefix from the home device to the foreign device. If the host is assigned multiple addresses, multiple binding entries will be generated, and these entries will be migrated. If the binding migration fails, it triggers the host to come back online, thus re-establishing the binding entries. We will specify the details in Section 5 depending on different deployment scenarios.

3.5. Binding Clearing

Three kinds of events will trigger binding clearing:

1. A host leaves explicitly this access point. All entries in the MAC-IP mapping table associated with this MAC address MUST be cleared.

2. A DHCP RELEASE message [RFC2131][RFC8415] is received from the owner of the corresponding IP address/prefix. This entry in the IP-MAC mapping table and the corresponding entries in the MAC-IP mapping table MUST be cleared.

3. A timeout message of the AC’s client idle-time is received. All entries in the MAC-IP mapping table related to the MAC address MUST be cleared.

4. Source Address Validation

This section describes source address validation procedure for packets. In this procedure, all the frames are considered to have passed the verification of 802.11i or other security mechanisms.

This procedure has the following steps:

1. Extract the IP source address and MAC source address from the frame. Look up the MAC address in the MAC-IP mapping table and check if the MAC-IP pair exists. If exists, forward the packet. Otherwise, go to step 2.

2. Look up the IP address in the IP-MAC mapping table and check if the IP address/prefix exists. If it does not exist, drop the packet. If it exists, check whether the MAC address in the entry is the same as that in the frame. If so, forward the packet. Otherwise, drop the packet.

In step 2, after the packet is judged to be valid and forwarded, synchronization between the MAC-IP and IP-MAC mapping tables SHOULD be triggered. The MAC-IP binding of the packet SHOULD be synchronized from the IP-MAC mapping table to the MAC-IP mapping table, and thus subsequent packets with the same MAC-IP pair will be forwarded without going to step 2.

5. Deployment Scenarios

This section specifies three deployment scenarios, including two under centralized WLAN and one under autonomous WLAN. The deployment details and solutions for host mobility between APs are described for each scenario, respectively.

5.1. Centralized WLAN

Centralized WLAN is comprised of lightweight access points (AP) and access controllers (AC). In this scenario, this document proposes the following two deployment solutions.

5.1.1. AP Filtering

With this deployment scheme, validated data packets received by an AP do not pass through the AC; only control packets and the questionable data packets pass through the AC. In this case, the AC maintains the IP-MAC mapping table, while the AP maintains the MAC-IP mapping table and performs address snooping.

5.1.1.1. Candidate Binding

An AP executes the procedure specified in Section 3.3. The candidate bindings are generated after the snooping procedure. Candidate bindings MUST be confirmed by the AC to be valid.

After a candidate binding is generated, the AC is notified and checks whether the binding is valid or not. If a candidate binding does not violate any existing binding in the IP-MAC mapping table, the validity of the binding is determined. Otherwise, if an address/prefix is not suitable for use by the host, the AC notifies the corresponding AP. If the candidate binding is valid, the AC adds an entry to the IP-MAC mapping table and notifies the AP. Afterwards, the AP also adds an entry to the local MAC-IP mapping table.

5.1.1.2. Packet Filtering

As specified in Section 4, for incoming data packets, an AP looks up the MAC address in the local MAC-IP mapping table and checks if the MAC-IP pair exists. If exists, the AP forwards the packet. Otherwise, the AP delivers the packet to the AC for further processing.

When receiving a data packet from the AP, the AC looks up the IP address or its prefix in the local IP-MAC mapping table and checks if the IP address/prefix exists. If it does not exist, the AC drops the packet. If it exists, the AC checks whether the MAC address in the entry is the same as that in the frame. If so, the AC forwards the packet. Otherwise, the AC drops the packet.

After the AC forwards a valid packet, it synchronizes the associated MAC-IP binding to the MAC-IP mapping table on the AP from which the packet comes. Subsequent packets with the same MAC-IP pair will be forwarded directly by the AP without going through the AC.

5.1.1.3. Negative Entries

In the AP filtering scenario, APs MAY drop packets directly without sending them to the AC by enabling the establishment of negative entries on APs. Specifically, APs may establish negative entries in the following circumstances.

1. When an AP receives a certain number of packets within a certain amount of time with the same MAC-IP pair that does not exist in the local MAC-IP mapping table, it establishes a negative entry for this MAC-IP pair. Then the AP drops all following packets that have the same MAC-IP pair as indicated in this negative entry without sending them to the AC for further processing.

2. When an AP receives a certain number of packets within a certain amount of time with the same MAC address but different MAC-IP pairs and none of these MAC-IP pairs exist in the local MAC-IP mapping table, it establishes a negative entry for this MAC address. Then the AP drops all the following packets that have the same MAC address as indicated in this negative entry without sending them to the AC for further processing.

Each negative entry has a limited lifetime. The number of packets and duration of time to trigger the establishment of the negative entry, and the lifetime of the negative entry are configurable.

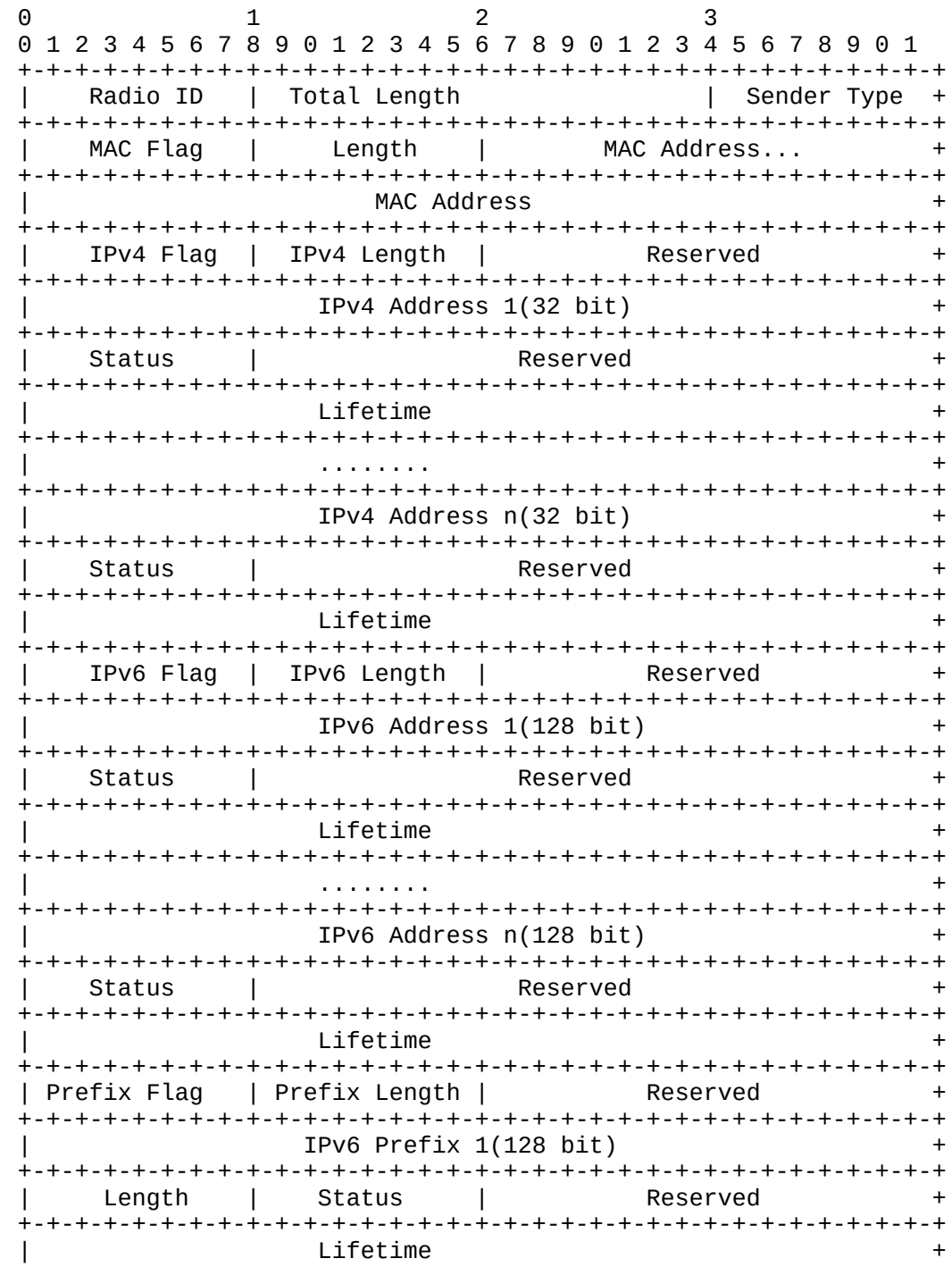
5.1.1.4. CAPWAP Extension

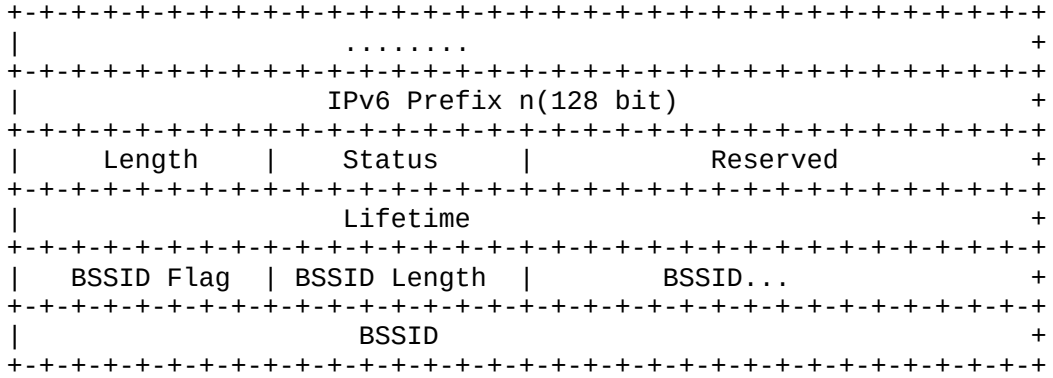
CAPWAP protocol is used for communication between the AP and the AC. A new CAPWAP protocol message element is introduced, which extends [RFC5415]. The host IP message element is used by both the AP and the AC to exchange the binding information of hosts.

The host IP message element can be used in the process of confirming candidate bindings. When the AP generates a candidate binding, it reports the MAC address and related IP addresses to the AC using this message, with suggestions of the status of each IP address/prefix (e.g., available, unavailable, candidate). After the AC checks the validity of the candidate binding, it replies using a message of the same format, informing the AP of the validation of each IP address/prefix with a suggested status.

The host IP message element can be used in the process of binding migration. When migration occurs, the source device uses this message to report the MAC address and related IP addresses to the destination device, with suggestions for the status of each IP address/prefix. After the destination device checks the validity of the candidate binding, it replies using a message of the same format to inform the source device of the validity of each IP address/prefix with a suggested status.

The host IP message element can also be used in other scenarios when the synchronization between MAC-IP and IP-MAC mapping tables is required as specified in Section 3.5 and Section 4. When the synchronization from IP-MAC mapping table to MAC-IP mapping table is triggered, the source device which holds the IP-MAC mapping table reports the MAC address and the related IP addresses/prefixes to the destination device which holds the MAC-IP mapping table using this message, with suggestions of the status of each IP address/prefix. The destination device replies using a message of the same format to acknowledge the source device.


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
Radio ID
Total Length
Sender Type
MAC Flag
Length
MAC Address...
MAC Address
IPv4 Flag
IPv4 Length
Reserved
IPv4 Address 1(32 bit)
Status
Reserved
Lifetime
.....
IPv4 Address n(32 bit)
Status
Reserved
Lifetime
IPv6 Flag
IPv6 Length
Reserved
IPv6 Address 1(128 bit)
Status
Reserved
Lifetime
.....
IPv6 Address n(128 bit)
Status
Reserved
Lifetime
Prefix Flag
Prefix Length
Reserved
IPv6 Prefix 1(128 bit)
Length
Status
Reserved
Lifetime



Radio ID: An 8-bit value representing the radio, whose value is between 1 and 31.

Total Length: Total length of the following fields.

Sender Type: An 8-bit value representing the sender of the message. AP is represented by value 1 and AC is represented by value 2.

MAC flag: An 8-bit value representing that the sub-field’s type is MAC address, whose value is 1.

MAC Length: The length of the MAC Address field. The formats and lengths specified in EUI-48 and EUI-64 [EUI] are supported.

MAC Address: A MAC address of the host. At least one MAC address block MUST appear in the message, otherwise the message is considered as invalid.

IPv4 Flag: An 8-bit value representing that the sub-field’s type is IPv4 address, whose value is 2.

IPv4 Length: The length of the IPv4 Address field.

IPv4 Address: An IPv4 address of the host. There may exist many entries, and each entry is comprised of an IPv4 address, an 8-bit value for address status (value 1 means available, value 0 means unavailable, value 255 means candidate), and a 32-bit value for lifetime. Lifetime refers to the valid time of the IPv4 Address. It is required to list all IPv4 addresses before IPv6 address blocks.

IPv6 Flag: An 8-bit value representing that the sub-field’s type is IPv6 address. A DHCPv6-assigned IP address is represented by the value 3, and a SLAAC-assigned IP address is represented by the value 4.

IPv6 Length: The length of the IPv6 Address field.

IPv6 Address: An IPv6 address of the host. There may exist many entries, and each entry is comprised of an IPv6 address, an 8-bit value of address status (value 1 means available, value 0 means unavailable, value 255 means candidate), and a 32-bit value lifetime. Lifetime is the lease time of the IPv6 address, or the valid time of the SLAAC-assigned IPv6 address. All IPv4 and IPv6 addresses bind to the MAC address that appears before them in the message.

Prefix Flag: An 8-bit value representing that the sub-field’s type is IPv6 prefix, whose value is 6.

Prefix Length: The length of the Prefix field.

IPv6 Prefix: An IPv6 address or a prefix of an IPv6 address. The Prefix Length field specifies the number of valid leading bits in the prefix. The bits in the prefix beyond the prefix length are reserved and MUST be ignored by the receiver. There may exist many entries, and each entry is comprised of an IPv6 prefix, an 8-bit value of prefix status (value 1 means available, value 0 means unavailable, and value 255 means candidate), and a 32-bit value lifetime. Lifetime is the valid time of the IPv6 prefix. All IPv4 and IPv6 addresses, as well as IPv6 prefixes, are bound to the MAC address that precedes them in the message.

Length: An 8-bit value that represents the number of leftmost contiguous bits of the address that form the IPv6 prefix. The value ranges from 0 to 128.

BSSID Flag: An 8-bit value representing that the sub-field’s type is BSSID, whose value is 5.

BSSID Length: The length of the BSSID field. The formats and lengths specified in EUI-48 and EUI-64 [EUI] are supported.

BSSID: A basic service set identifier representing the BSS.

5.1.1.5. Mobility Solution

When a host moves from one AP to another, Layer-2 association happens before IP packet forwarding begins. The home AP deletes the binding when the mobile host is disconnected, and the foreign AP immediately requests the bound addresses/prefixes with the associated MAC address from the AC. The AC returns the binding with a suggested status. Once the foreign AP gets the addresses/prefixes that should be bound, binding migration is completed. The protocol used for communication between the foreign AP and the AC is the same as described in Section 5.1.1.4, where the AC acts as the source device and the foreign AP as the destination device.

In WLAN deployments where a host moves between ACs while maintaining the same IP address, binding migration between ACs is also required. In this case, ACs must communicate using the same protocol as in Section 5.1.1.4. Here, the home AC functions as the source device, and the foreign AC as the destination device, enabling seamless IP continuity for the mobile host.

5.1.2. AC Filtering

In this scenario, an AC maintains both the MAC-IP and IP-MAC mapping tables and performs both address snooping and packet filtering. Therefore, all the packets must be forwarded to the AC first.

The AC executes the procedure specified in Section 3.3 and checks the validity of IP-MAC pairs by consulting the local IP-MAC mapping table. No extra procedure is needed to establish the IP-MAC bindings.

The AC executes the procedure specified in Section 4 for packet filtering, and no extra procedure is involved.

Host movement within an AC does not trigger any binding migration. Host movement between different ACs triggers binding migration. ACs must communicate to perform binding migration. The protocol used for communication between ACs is the same as described in Section 5.1.1.4, while in this scenario the home AC serves the role of the source device and the foreign AC serves the role of the destination device.

5.2. Autonomous WLAN

Autonomous WLAN is comprised of autonomous access points. In this scenario, an autonomous AP maintains both the MAC-IP and IP-MAC mapping tables and performs both address snooping and packet filtering.

The autonomous AP executes the procedure specified in Section 3.3 and checks the validity of IP-MAC pairs by consulting the local IP-MAC mapping table. No extra procedure is needed to establish the IP-MAC bindings.

The autonomous AP executes the procedure specified in Section 4 for packet filtering, and no extra procedure is involved.

Mobility between different autonomous APs will trigger binding migration. Autonomous APs must communicate to perform the binding migration. Considering that the communication protocol implementations among autonomous APs from different vendors may vary, this document does not specify the exact implementation method. Vendors only need to migrate the IP address/prefix and MAC address binding entries described in Section 3.1 between autonomous APs.

6. IANA Considerations

There is no IANA consideration currently.

7. Security Considerations

The security of address allocation methods matters the security of this mechanism. Thus, it is necessary to improve the security of stateless auto-configuration and DHCP first.

Section 3.3 mentions the possibility of an IPv6 interface having multiple addresses, which may in fact pose a risk. For example, if no restrictions are applied, an attacker could use the same MAC address for as many IP-MAC bindings as possible. In this case, other hosts may not be able to trigger the creation of any binding entries and therefore cannot get their packets through the SAVI device. To counteract potential security risks, additional mechanisms must be used, for example, to limit the maximum number of IPs that can be bound to a MAC address. However, considering that it is a reasonable requirement for a host to have many addresses in future use cases [RFC7934], it is RECOMMENDED that the maximum number of binding entries for the same MAC address not be set too small, and that the value SHOULD also be adjustable.

Moreover, a SAVI device MUST delete binding anchor information as soon as possible, except where there is an identified reason why that information is likely to be involved in the detection, prevention, or tracing of actual source-address spoofing. Information about hosts that never spoof (probably the majority of hosts) SHOULD NOT be logged.

8. Randomized MAC Address Considerations

The use of randomized MAC addresses is a common measure to protect user privacy [I-D.draft-ietf-madinas-mac-address-randomization]. It has become a common practice for mainstream operating systems such as Windows, iOS, and Android to use randomized MAC addresses to join networks. Hosts primarily utilize the PNGM, PPGM, and PSGM methods [I-D.draft-ietf-madinas-mac-address-randomization] to generate random MAC addresses. These methods ensure that the MAC address remains unchanged once the client is connected to the network. In [IEEE802.11-2020], to ensure continuous communication, hosts are not allowed to change their MAC addresses during transactional exchanges. Therefore, randomized MAC addresses do not affect the functionality of SAVI.

9. Acknowledgements

The authors would like to thank Jun Bi, Guang Yao, Yang Shi, and Hao Wang for their contributions to this document.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

10.2. Informative References

[EUI] IEEE Standards Association, "Guidelines for Use of Extended Unique Identifier (EUI), Organizationally Unique Identifier (OUI), and Company ID (CID)", 2017, <https://standards.ieee.org/content/dam/ieee-standards/standards/web/documents/tutorials/eui.pdf>.

[I-D.draft-ietf-dhc-addr-notification] Kumari, W., Krishnan, S., Asati, R., Colitti, L., Linkova, J., and S. Jiang, "Registering Self-generated IPv6 Addresses using DHCPv6", March 2024.

[I-D.draft-ietf-madinas-mac-address-randomization] Zuniga, JC., Bernardos, CJ., and A. Andersdotter, "Randomized and Changing MAC Address", March 2023.

[I-D.draft-ietf-v6ops-dhcp-pd-per-device]
Colitti, L., Linkova, J., and X. Ma, "Using DHCPv6-PD to Allocate Unique IPv6 Prefix per Client in Large Broadcast Networks", February 2024.

[IEEE802.11-2020]
IEEE Std 802.11-2020, "IEEE Standard for Information Technology--Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks--Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 2021.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <https://www.rfc-editor.org/info/rfc2131>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <https://www.rfc-editor.org/info/rfc4862>.

[RFC5415] Calhoun, P., Ed., Montemurro, M., Ed., and D. Stanley, Ed., "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", RFC 5415, DOI 10.17487/RFC5415, March 2009, <https://www.rfc-editor.org/info/rfc5415>.

[RFC6620] Nordmark, E., Bagnulo, M., and E. Levy-Abegnoli, "FCFS SAVI: First-Come, First-Served Source Address Validation Improvement for Locally Assigned IPv6 Addresses", RFC 6620, DOI 10.17487/RFC6620, May 2012, <https://www.rfc-editor.org/info/rfc6620>.

[RFC7039] Wu, J., Bi, J., Bagnulo, M., Baker, F., and C. Vogt, Ed., "Source Address Validation Improvement (SAVI) Framework", RFC 7039, DOI 10.17487/RFC7039, October 2013, <https://www.rfc-editor.org/info/rfc7039>.

[RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, DOI 10.17487/RFC7217, April 2014, <https://www.rfc-editor.org/info/rfc7217>.

[RFC7219] Bagnulo, M. and A. Garcia-Martinez, "SEcure Neighbor Discovery (SEND) Source Address Validation Improvement (SAVI)", RFC 7219, DOI 10.17487/RFC7219, May 2014, <https://www.rfc-editor.org/info/rfc7219>.

[RFC7513] Bi, J., Wu, J., Yao, G., and F. Baker, "Source Address Validation Improvement (SAVI) Solution for DHCP", RFC 7513, DOI 10.17487/RFC7513, May 2015, <https://www.rfc-editor.org/info/rfc7513>.

[RFC7934] Colitti, L., Cerf, V., Cheshire, S., and D. Schinazi, "Host Address Availability Recommendations", BCP 204, RFC 7934, DOI 10.17487/RFC7934, July 2016, <https://www.rfc-editor.org/info/rfc7934>.

[RFC8064] Gont, F., Cooper, A., Thaler, D., and W. Liu, "Recommendation on Stable IPv6 Interface Identifiers", RFC 8064, DOI 10.17487/RFC8064, February 2017, <https://www.rfc-editor.org/info/rfc8064>.

[RFC8074] Bi, J., Yao, G., Halpern, J., and E. Levy-Abegnoli, Ed., "Source Address Validation Improvement (SAVI) for Mixed Address Assignment Methods Scenario", RFC 8074, DOI 10.17487/RFC8074, February 2017, <https://www.rfc-editor.org/info/rfc8074>.

[RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <https://www.rfc-editor.org/info/rfc8415>.

[RFC8981] Gont, F., Krishnan, S., Narten, T., and R. Draves, "Temporary Address Extensions for Stateless Address Autoconfiguration in IPv6", RFC 8981, DOI 10.17487/RFC8981, February 2021, <https://www.rfc-editor.org/info/rfc8981>.

Authors' Addresses

Mingwei Xu
Tsinghua University
Beijing
100084
China
Email: xmw@cernet.edu.cn

Jianping Wu
Tsinghua University
Beijing
100084
China
Email: jianping@cernet.edu.cn

Tao Lin
New H3C Technologies Co. Ltd
466 Changhe Road, Binjiang District
Hangzhou
Zhejiang, 310052
China
Email: lintao@h3c.com

Lin He
Tsinghua University
Beijing
100084
China
Email: he-lin@tsinghua.edu.cn

You Wang
Tsinghua University
Beijing
100084
China
Email: wangyou10@mails.tsinghua.edu.cn