Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: 11 April 2025

J. Hui
Google LLC

8 October 2024

SNAC Router Flag in ICMPv6 Router Advertisement Messages
draft-ietf-6man-snac-router-ra-flag-02

Abstract

This document defines a new flag, the SNAC Router flag, in the Router Advertisement message that can be used to distinguish configuration information sent by SNAC routers from information sent by infrastructure routers. This flag is used only by SNAC routers and is ignored by all other devices.

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 11 April 2025.

Table of Contents

1. Introduction

A Stub Network Auto-Configuring Router (SNAC) router is an autonomously-configuring router that provides IP connectivity between one or more stub networks and one or more infrastructure networks. A common SNAC router example is a device that attaches a 6LoWPAN-based network to a home network, automatically providing IPv6 forwarding between the two networks without explicit operator configuration. SNAC routers are described in [I-D.ietf-snac-simple]. This document defines a new IPv6 ND Router Advertisement (RA) flag, the "SNAC router" flag, which SNAC routers use to identify RAs sent by other SNAC routers.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. SNAC Router Flag

The "SNAC Router" flag is router advertisement flag bit TBD.

The SNAC router flag is to be used by SNAC routers. The use of this flag is documented in [I-D.ietf-snac-simple]. Devices that do not operate as SNAC routers [I-D.ietf-snac-simple] MUST NOT set the SNAC router flag, and MUST silently ignore the SNAC router flag. This means that the presence or absence of the flag should not change the behavior of such devices in any way (other than that it is of course permissible to log and cache the value of the flag as part of normal router advertisement processing, where applicable).

In environments that implement RA guard in a way that filters RAs sent by SNAC routers, devices on the infrastructure network should never receive an RA with the SNAC router flag set.

4. IANA Considerations

IANA is requested to allocate a flag from the "Internet Control Message Protocol version 6 (ICMPv6) Parameters", "IPv6 ND Router Advertisement flags" registry [IANA-RA-FLAGS] as specified below:

RA Option Bit	Description	Reference
TBD	S - SNAC Router Flag	This Document

Table 1

5. Security Considerations

The security considerations of IPv6 ND are documented in the "Security Considerations" section of [RFC4861]. The addition of the SNAC router flag does not changes these considerations.

6. Normative References

[I-D.ietf-snac-simple]
Lemon, T. and J. Hui, "Automatically Connecting Stub Networks to Unmanaged Infrastructure", Work in Progress, Internet-Draft, draft-ietf-snac-simple-05, 8 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-snac-simple-05>.

[IANA-RA-FLAGS]
IANA, "IPv6 ND Router Advertisement flags", <https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml#icmpv6-parameters-11>.

[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>.

[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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Author's Address

Jonathan Hui
Google LLC
1600 Amphitheatre Parkway
Mountain View, California 940432
United States of America
Email: jonhui@google.com

anima Working Group
Internet-Draft
Intended status: Standards Track
Expires: 9 May 2025

T. Werner
Siemens AG
M. Richardson
Sandelman Software Works

5 November 2024

JWS signed Voucher Artifacts for Bootstrapping Protocols
draft-ietf-anima-jws-voucher-13

Abstract

I-D.ietf-anima-rfc8366bis defines a digital artifact (known as a voucher) as a YANG-defined JSON document that is signed using a Cryptographic Message Syntax (CMS) structure. This document introduces a variant of the voucher artifact in which CMS is replaced by the JSON Object Signing and Encryption (JOSE) mechanism described in RFC7515 to support deployments in which JOSE is preferred over CMS. In addition to specifying the format, the "application/voucher-jws+json" media type is registered and examples are provided.

Table of Contents

1. Introduction

"A Voucher Artifact for Bootstrapping Protocols" [I-D.ietf-anima-rfc8366bis] defines a YANG-based data structure used in "Bootstrapping Remote Secure Key Infrastructure" (BRSKI) [RFC8995] and "Secure Zero Touch Provisioning" (SZTP) [RFC8572] to transfer ownership of a device from a manufacturer to a new owner (customer or operational domain). That document provides a serialization of the voucher data to JSON [RFC8259] with cryptographic signing according to the Cryptographic Message Syntax (CMS) [RFC5652]. That resulting voucher artifact has the media type application/voucher-cms+json.

This document provides cryptographic signing of voucher data in form of JSON Web Signature (JWS) [RFC7515] and the media type application/voucher-jws+json to identify the voucher format. The encoding specified in this document is used by [I-D.ietf-anima-brski-prm] and may be more handy for use cases already using Javascript Object Signing and Encryption (JOSE).

This document should be considered as enhancement of [I-D.ietf-anima-rfc8366bis], as it provides a new voucher format. It is similar to [I-D.ietf-anima-constrained-voucher], which provides cryptographic signing according COSE [RFC8812] and the media type application/voucher-cose+cbor. These documents do not change nor extend the YANG definitions of [I-D.ietf-anima-rfc8366bis].

With the availability of different voucher formats, it is up to an industry-specific application statement to decide which format is to be used. The associated media types are used to distinguish different voucher formats.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

This document uses the following terms:

JSON Voucher Data: An unsigned JSON representation of the voucher data.

JWS Voucher: A JWS structure signing the JSON Voucher Data.

Voucher: A short form for voucher artifact and refers to the signed statement from Manufacturer Authorized Signing Authority (MASA) service that indicates to a Pledge the cryptographic identity of the domain it should trust, per [I-D.ietf-anima-rfc8366bis].

Voucher Data: The raw (serialized) representation of the ietf-voucher YANG module without any enclosing signature, per [I-D.ietf-anima-rfc8366bis].

MASA (Manufacturer Authorized Signing Authority): The entity that, for the purpose of this document, issues and signs the vouchers for manufacturer's pledges. In some onboarding protocols, the MASA may have an Internet presence and be integral to the onboarding process, whereas in other protocols the MASA may be an offline service that has no active role in the onboarding process, per [I-D.ietf-anima-rfc8366bis].

Pledge: The prospective component attempting to find and securely join a domain. When shipped or in factory reset mode, it only trusts authorized representatives of the manufacturer, per [I-D.ietf-anima-rfc8366bis].

Registrar: A representative of the domain that is configured, perhaps autonomically, to decide whether a new device is allowed to join the domain, per [I-D.ietf-anima-rfc8366bis].

This document uses the following encoding notations:

BASE64URL(OCTETS): Denotes the base64url encoding of OCTETS, per Section 2 of [RFC7515].

UTF8(STRING): Denotes the octets of the UTF-8 [RFC3629] representation of STRING, per Section 1 of [RFC7515].

3. Voucher Artifact with JSON Web Signature

JWS voucher artifacts MUST use the "General JWS JSON Serialization Syntax" defined in Section 7.2.1 of [RFC7515]. This syntax supports multiple signatures as already supported by [RFC8366] for CMS-signed vouchers. The following figure summarizes the serialization of JWS voucher artifacts:

```
{
  "payload": BASE64URL(UTF8(JSON Voucher Data)),
  "signatures": [
    {
      "protected": BASE64URL(UTF8(JWS Protected Header)),
      "signature": BASE64URL(JWS Signature)
    }
  ]
}
```

Figure 1: Voucher Representation in General JWS JSON Serialization Syntax (JWS Voucher)

The JSON Voucher Data MUST be UTF-8 encoded to become the octet-based JWS Payload defined in [RFC7515]. The JWS Payload is further base64url-encoded to become the string value of the payload member as described in Section 3.2 of [RFC7515]. The octets of the UTF-8 representation of the JWS Protected Header are base64url-encoded to become the string value of the protected member. The generated JWS Signature is base64url-encoded to become the string value of the signature member.

3.1. JSON Voucher Data

The JSON Voucher Data is an unsigned JSON document [RFC8259] that conforms with the data model described by the ietf-voucher YANG module [RFC7950] defined in Section 7.3 of [I-D.ietf-anima-rfc8366bis] and is encoded using the rules defined in [RFC7951]. The following figure provides an example of JSON Voucher Data:

```
{
  "ietf-voucher:voucher": {
    "assertion": "logged",
    "serial-number": "0123456789",
    "nonce": "5742698422680472",
    "created-on": "2022-07-08T03:01:24.618Z",
    "pinned-domain-cert": "base64encodedvalue=="
  }
}
```

Figure 2: JSON Voucher Data Example

3.2. JWS Protected Header

The JWS Protected Header defined in [RFC7515] uses the standard header parameters alg, typ, and x5c:

* The alg parameter MUST contain the algorithm type (e.g., ES256) used to create the signature as defined in Section 4.1.1 of [RFC7515].

* The typ parameter is optional and used when more than one kind of object could be present in an application data structure as described in Section 4.1.9 of [RFC7515]. If present, the typ parameter MUST contain the value voucher-jws+json.

* If X.509 (PKIX) certificates [RFC5280] are used, the x5c parameter MUST contain the base64-encoded (not base64url-encoded) X.509 v3 (DER) certificate as defined in Section 4.1.6 of [RFC7515] and SHOULD also contain the certificate chain.

Implementation Note: base64-encoded values, in contrast to base64url-encoded values, may contain slashes (/). JSON [RFC8259] optionally allows escaping these with backslashes (\\). Hence, depending on the JSON parser/serializer implementation used, they may or may not be included. JWS Voucher parsers need to be prepared accordingly to extract certificates correctly.

To validate voucher signatures, all certificates of the certificate chain are required up to the trust anchor. Note, to establish trust the trust anchor SHOULD be provided out-of-band up front.

The following figure gives an example of a JWS Protected Header:

```
{
  "alg": "ES256",
  "typ": "voucher-jws+json",
  "x5c": [
    "base64encodedvalue1==",
    "base64encodedvalue2=="
  ]
}
```

Figure 3: JWS Protected Header Example

3.3. JWS Signature

The JWS Signature is generated over the JWS Protected Header and the JWS Payload (= UTF-8 encoded JSON Voucher Data) as described in Section 5.1 of [RFC7515].

4. Privacy Considerations

The Pledge-Voucher-Request (PVR) reveals the IDevID of the component (Pledge) that is in the process of bootstrapping.

A PVR is transported via HTTP-over-TLS. However, for the Pledge-to-Registrar TLS connection a Pledge provisionally accepts the Registrar server certificate during the TLS server authentication. Hence, it is subject to disclosure by a Dolev-Yao attacker (a "malicious messenger") [ON-PATH], as explained in Section 10.2 of [RFC8995].

The use of a JWS header brings no new privacy considerations.

5. Security Considerations

The issues of how [I-D.ietf-anima-rfc8366bis] vouchers are used in a [BRSKI] system is addressed in Section 11 of [RFC8995]. This document does not change any of those issues, it just changes the signature technology used for voucher request and response artifacts.

Section 9 of [RFC8572] deals with voucher use in Secure Zero Touch Provisioning (SZTP), for which this document also makes no changes to security.

6. IANA Considerations

6.1. Media-Type Registry

This section registers application/voucher-jws+json in the "Media Types" registry.

6.1.1. application/voucher-jws+json

Type name: application
Subtype name: voucher-jws+json
Required parameters: none
Optional parameters: none
Encoding considerations: JWS+JSON vouchers are JOSE objects signed with one or multiple signers.
Security considerations: See section [Security Considerations]
Interoperability considerations: The format is designed to be broadly interoperable.
Published specification: [THIS RFC].
Applications that use this media type: ANIMA, 6tisch, and other zero-touch bootstrapping/provisioning solutions
Additional information:
Magic number(s): None
File extension(s): .vjj
Macintosh file type code(s): none
Person & email address to contact for further information: IETF ANIMA WG
Intended usage: LIMITED
Restrictions on usage: NONE
Author: ANIMA WG
Change controller: IETF
Provisional registration? (standards tree only): NO

7. Acknowledgments

We would like to thank the various reviewers for their input, in particular Steffen Fries, Ingo Wenda, Esko Dijk and Toerless Eckert. Thanks for the supporting PoC implementations to Hong Rui Li and He Peng Jia.

8. Examples

These examples are folded according to the [RFC8792] Single Backslash rule.

8.1. Example Pledge-Voucher-Request (PVR)

The following is an example of a Pledge-Voucher-Request (PVR) as JWS Voucher artifact, which would be sent from a Pledge to the Registrar:


```
{
  "payload": "eyJpZXRmLXZvdWNoZXItcmVxdWVzdDp2b3VjaGVyIjp7InNlcmlhbC\
1udW1iZXIiOiIwMTIzNDU2Nzg5Iiwibm9uY2UiOiI2R3RuK1pRS04ySHFERlZrQkV4Wk\
xRPT0iLCJjcmVhdGVkLW9uIjoiMjAyMi0wNy0wOFQwODo0MDo0Mi44MjBaIiwicHJveG\
ltaXR5LXJlZ2lzdHJhci1jZXJ0IjoiTUlJQjRqQ0NBWWlnQXdJQkFnSUdBWFk3MmJiWk\
1Bb0dDQ3FHU000OUJBTUNNRFV4RXpBUkJnTlZCQW9NQ2sxNVFuVnphVzVsYzNNeERUQU\
xCZ05WQkFjTUJGTnBkR1V4RHpBTkJnTlZCQU1NQmxSbGMzUkRRVEFlRncweU1ERXlNRG\
N3TmpFNE1USmFGdzB6TURFeU1EY3dOakU0TVRKYU1ENHhFekFSQmdOVkJBb01DazE1UW\
5WemFXNWxjM014RFRBTEJnTlZCQWNNQkZOcGRHVXhHREFXQmdOVkJBTU1EMFJ2YldGcG\
JsSmxaMmx6ZEhKaGNqQlpNQk1HQnlxR1NNNDlBZ0VHQ0NxR1NNNDlBd0VIQTBJQUJCaz\
E2Sy9pNzlvUmtLNVliZVBnOFVTUjgvdXMxZFBVaVpITXRva1NkcUtXNWZuV3NCZCtxUk\
w3V1JmZmVXa3lnZWJvSmZJbGx1cmNpMjV3bmhpT1ZDR2plekI1TUIwR0ExVWRKUVFXTU\
JRR0NDc0dBUVVGQndNQkJnZ3JCZ0VGQlFjREhEQU9CZ05WSFE4QkFmOEVCQU1DQjRBd1\
NBWURWUjBSQkVFd1A0SWRjbVZuYVhOMGNtRnlMWFJsYzNRdWMybGxiV1Z1Y3kxaWRDNX\
VaWFNDSG5KbFoybHpkSEpoY2kxMFpYTjBOaTV6YVdWdFpXNXpMV0owTG01bGREQUtCZ2\
dxaGtqT1BRUURBZ05JQURCRkFpQnhsZEJoWnEwRXY1SkwyUHJXQ3R5UzZoRFlXMXlDTy\
9SYXVicEM3TWFJRGdJaEFMU0piZ0xuZ2hiYkFnMGRjV0ZVVm8vZ0dOMC9qd3pKWjBTbD\
JoNHhJWGsxIn19",
  "signatures": [{
    "protected": "eyJ4NWMiOlsiTUlJQitUQ0NBYUNnQXdJQkFnSUdBWG5WanNVNU\
1Bb0dDQ3FHU000OUJBTUNNRDB4Q3pBSkJnTlZCQVlUQWtGUk1SVXdFd1lEVlFRS0RBeE\
thVzVuU21sdVowTnZjbkF4RnpBVkJnTlZCQU1NRGtwcGJtZEthVzVuVkdWemRFTkJNQ0\
FYRFRJeE1EWXdOREExTkRZeE5Gb1lEems1T1RreE1qTXhNak0xT1RVNVdqQlNNUXN3Q1\
FZRFZRUUdFd0pCVVRFVk1CTUdBMVVFQ2d3TVNtbHVaMHBwYm1kRGIzSndNUk13RVFZRF\
ZRUUZFd293TVRJek5EVTJOemc1TVJjd0ZRWURWUVFEREE1S2FXNW5TbWx1WjBSbGRtbG\
paVEJaTUJNR0J5cUdTTTQ5QWdFR0NDcUdTTTQ5QXdFSEEwSUFCQzc5bGlhUmNCalpjRU\
VYdzdyVWVhdnRHSkF1SDRwazRJNDJ2YUJNc1UxMWlMRENDTGtWaHRVVjIxbXZhS0N2TX\
gyWStTTWdROGZmd0wyM3ozVElWQldqZFRCek1Dc0dDQ3NHQVFVRkJ3RWdCQjhXSFcxaG\
MyRXRkR1Z6ZEM1emFXVnRaVzV6TFdKMExtNWxkRG81TkRRek1COEdBMVVkSXdRWU1CYU\
FGRlFMak56UFwvU1wva291alF3amc1RTVmdndjWWJNQk1HQTFVZEpRUU1NQW9HQ0NzR0\
FRVUZCd01DTUE0R0ExVWREd0VCXC93UUVBd0lIZ0RBS0JnZ3Foa2pPUFFRREFnTkhBRE\
JFQWlCdTN3UkJMc0pNUDVzTTA3MEgrVUZyeU5VNmdLekxPUmNGeVJST2xxcUhpZ0lnWE\
NtSkxUekVsdkQycG9LNmR4NmwxXC91eW1UbmJRRERmSmxhdHVYMlJvT0U9Il0sInR5cC\
I6InZvdWNoZXItandzK2pzb24iLCJhbGciOiJFUzI1NiJ9",
    "signature": "abVg4TDGzSTjVHkQlNeIW3ABu5ZXdMl1cEqwcIAlHFW4BrlGbO\
-DRTKfyCOGxSW49-ktJcrVlYgKqC4xmZoy0Q"
  }]
}
```


Figure 4: Example Pledge-Voucher-Request (PVR)

8.2. Example Parboiled Registrar-Voucher-Request (RVR)

The term parboiled refers to food which is partially cooked. In [BRSKI], the term refers to a Pledge-Voucher-Request (PVR) that was received by the Registrar, then has been processed by the Registrar ("cooked"), and is now being forwarded to the MASA.

The following is an example Registrar-Voucher-Request (RVR) as JWS Voucher artifact, which would be sent from the Registrar to the MASA. Note that the previous PVR can be seen in the payload in the field prior-signed-voucher-request.


```
{
  "payload": "eyJpZXRmLXZvdWNoZXItcmVxdWVzdDp2b3VjaGVyIjp7InNlcmlhbC\
1udW1iZXIiOiIwMTIzNDU2Nzg5IiwiaWRldmlkLWlzc3VlciI6IkJCZ3dGb0FVVkF1TT\
NNLzlMK1NpNk5EQ09Ea1RsKy9CeGhzPSIsIm5vbmNlIjoiNkd0bitaUUtOMkhxREZWa0\
JFeFpMUT09IiwicHJpb3Itc2lnbmVkLXZvdWNoZXItcmVxdWVzdCI6ImV5SndZWGxzYj\
JGa0lqb2laWGxLY0ZwWVVtMU1XRnAyWkZkT2IxcFlTWFJqYlZaNFpGZFdlbVJFY0RKaU\
0xWnFZVWRXZVVscWNEZEpiazVzWTIxc2FHSkRNWFZrVnpGcFdsaEphVTlwU1hkTlZFbD\
ZUa1JWTWs1Nlp6VkphWGRwWW0wNWRWa3lWV2xQYVVreVVqTlNkVXN4Y0ZKVE1EUjVVMG\
hHUlZKc1duSlJhMVkwVjJ0NFVsQlVNR2xNUTBwcVkyMVdhR1JIVm10TVZ6bDFTV3B2YV\
UxcVFYbE5hVEIzVG5rd2QwOUdVWGRQUkc4d1RVUnZNRTFwTkRSTmFrSmhTV2wzYVdOSV\
NuWmxSMngwWVZoU05VeFlTbXhhTW14NlpFaEthR05wTVdwYVdFb3dTV3B2YVZSVmJFcF\
JhbEp4VVRCT1FsZFhiRzVSV0dSS1VXdEdibE5WWkVKWFJtc3pUVzFLYVZkck1VSmlNR1\
JFVVROR1NGVXdNREJQVlVwQ1ZGVk9UbEpHVmpSU1dIQkNWV3RLYmxSc1drTlJWemxPVV\
RKemVFNVdSblZXYm5Cb1ZucFdjMWw2VGs1bFJWSlZVVlY0UTFvd05WZFJhMFpxVkZWS1\
IxUnVRbXRTVZZMFVraHdRbFJyU201VWJGcERVVlV4VGxGdGVGTmlSMDE2Vld0U1VsWk\
ZSbXhTYm1OM1pWVXhSVkpZYkU1U1IwNHpWRzF3Ums1Rk1WVlRiVVpIWkhwQ05sUlZVa1\
psVlRGRldUTmtUMkZyVlRCVVZsSkxXVlV4UlU1SWFFWmxhMFpUVVcxa1QxWnJTa0ppTU\
RGRVlYcEZNVlZYTlZkbGJVWllUbGQ0YWswd01UUlNSbEpDVkVWS2JsUnNXa05SVjA1T1\
VXdGFUMk5IVWtoV1dHaElVa1ZHV0ZGdFpFOVdhMHBDVkZVeFJVMUdTakpaYkdSSFkwZE\
tjMU50ZUdGTmJYZzJXa1ZvUzJGSFRuRlJiSEJPVVdzeFNGRnViSGhTTVU1T1RrUnNRbG\
93VmtoUk1FNTRVakZPVGs1RWJFSmtNRlpKVVZSQ1NsRlZTa05oZWtVeVUzazVjRTU2Yk\
haVmJYUk1UbFpzYVZwV1FtNVBSbFpVVldwbmRtUllUWGhhUmtKV1lWWndTVlJZVW5aaE\
1VNXJZMVYwV0U1WFduVldNMDVEV2tOMGVGVnJkek5XTVVwdFdtMVdXR0V6Ykc1YVYwcD\
JVMjFhU21KSGVERmpiVTV3VFdwV00ySnRhSEJVTVZwRVVqSndiR1ZyU1RGVVZVbDNVak\
JGZUZaWFVrdFZWa1pZVkZWS1VsSXdUa1JqTUdSQ1ZWWldSMUZ1WkU1UmEwcHVXak5LUT\
Fvd1ZrZFJiRVpxVWtWb1JWRlZPVU5hTURWWFUwWkZORkZyUm0xUFJWWkRVVlV4UkZGcV\
VrSmtNVTVDVjFWU1YxVnFRbE5SYTFaR1pERkJNRk5YVW1waVZscDFXVlpvVDAxSFRuUl\
NibXhOVjBaS2MxbDZUbEprVjAxNVlrZDRhVll4V2pGWk0ydDRZVmRTUkU1WVZtRlhSaz\
VFVTBjMVMySkdiM2xpU0hCclUwVndiMWt5YTNoTlJuQlpWR3BDVDJGVVZqWlpWbVJYWk\
Vad1dFNVljRTFXTUc5M1ZFY3dNV0pIVWtWUlZYUkRXakprZUdGSGRIRlVNVUpTVlZWU1\
Fsb3dOVXBSVlZKRFVtdEdjRkZ1YUhOYVJVcHZWMjVGZDFKWVdURlRhM2Q1VlVoS1dGRX\
pValZWZWxwdlVrWnNXRTFZYkVSVWVUbFRXVmhXYVdORlRUTlVWMFpLVWtka1NtRkZSaz\
FWTUhCcFdqQjRkVm95YUdsWmEwWnVUVWRTYWxZd1dsWldiVGgyV2pCa1QwMURPWEZrTT\
NCTFYycENWR0pFU205T1NHaEtWMGR6ZUVsdU1Ua2lMQ0p6YVdkdVlYUjFjbVZ6SWpwYm\
V5SndjbTkwWldOMFpXUWlPaUpsZVVvMFRsZE5hVTlzYzJsVVZXeEtVV2wwVlZFd1RrSl\
pWVTV1VVZoa1NsRnJSbTVUVldSQ1YwYzFWMkZ1VGxaT1ZURkNZakJrUkZFelJraFZNRE\
```

F3VDFWS1FsUlZUazVTUkVJMFVUTndRbE5yU201VWJGcERVVlpzVlZGWGRFZFZhekZUVm\
xoa1JtUXhiRVZXYkVaU1V6QlNRbVZGZEdoV2VsWjFWVEl4YzJSV2IzZFVibHBxWW10R0\
5GSnVjRUpXYTBwdVZHeGFRMUZWTVU1U1IzUjNZMGRLZEZwRmRHaFdlbFoxVm10a1YyVn\
RVa1pVYTBwT1VUQkdXVkpHVWtwbFJURkZWMWhrVDFKRlJYaFVhMUphWlVVMVIySXhiRV\
ZsYlhNeFZERlNjbVZGTVhGVVdHaE9ZV3N3ZUZReFVsWk9WbVJ4VVd4T1RsVllUak5STV\
VaYVVrWmFVbFZWWkVaa01IQkRWbFpTUmxack1VTlVWV1JDVFZaV1JsRXlaRE5VVms1MF\
lraFdZVTFJUW5kWmJURnJVa2RKZWxOdVpFNVZhekV6VWxaR1dsSkdXbEpWVlZwR1pEST\
VNMVJXVWtwbGF6VkZWbFJLVDJWdFl6RlVWa3BxWkRCYVVsZFZVbGRWVmtaRlVrVkZNVk\
15UmxoT1Z6VlVZbGQ0TVZkcVFsTmlSMUowWWtkd1lWWkZTbUZVVlVwT1VqQktOV05WWk\
ZSVVZGRTFVVmRrUmxJd1RrUmpWV1JVVkZSUk5WRllaRVpUUlVWM1UxVkdRMUY2WXpWaV\
IyeG9WVzFPUTJGc2NHcFNWVlpaWkhwa2VWWlhWbWhrYmxKSVUydEdNVk5FVW5kaGVsSk\
tUa1JLTWxsVlNrNWpNVlY0VFZkc1RWSkZUa1JVUjNSWFlVaFNWbFpxU1hoaVdGcG9Vek\
JPTWxSWVozbFhVM1JVVkZka1VrOUhXbTFrTUhkNVRUTnZlbFpGYkZkUmJHUnhXa1pTUT\
JWck1VUmpNR1JFVVROT1NGRldSbFpTYTBvelVsZGtRMUZxYUZoVFJtTjRZVWROZVZKWV\
VtdFNNVm8yV2tWTk1XVnRSbGhXYmxKaFZucFdObFJHWkV0TlJYaDBUbGQ0YTFKSE9ERl\
VhMUpTWldzeFEwOUZaRUpOVmxaclUxaGtVbGRWTVVOWlZVWkhVbXhHVFdGck5UWlZSbm\
QyVlRGM2RtRXlPVEZoYkVZellXMWpNVkpVVm0xa2JtUnFWMWRLVGxGck1VaFJWRVpXV2\
tWd1VsVlZNVTVSVnpsSVVUQk9lbEl3UmxKV1ZWcERaREF4UkZSVlJUQlNNRVY0VmxkU1\
JXUXdWa05ZUXprelZWVldRbVF3YkVsYU1GSkNVekJLYmxvelJtOWhNbkJRVlVaR1VsSk\
ZSbTVVYTJoQ1VrVktSbEZYYkVOa1ZFNHpWV3RLVFdNd2NFNVZSRlo2VkZSQk0wMUZaM0\
pXVlZwNVpWVTFWazV0WkV4bGEzaFFWVzFPUjJWV1NsTlVNbmg0WTFWb2NGb3diRzVYUl\
U1MFUydDRWV1ZyVm5Oa2ExRjVZMGM1VEU1dFVqUk9iWGQ0V0VNNU1XVlhNVlZpYlVwU1\
VrVlNiVk50ZUdoa1NGWlpUV3hLZGxRd1ZUbEpiREJ6U1c1U05XTkRTVFpKYmxwMlpGZE\
9iMXBZU1hSaGJtUjZTekp3ZW1JeU5HbE1RMHBvWWtkamFVOXBTa1pWZWtreFRtbEtPU0\
lzSW5OcFoyNWhkSFZ5WlNJNkltRmlWbWMwVkVSSGVsTlVhbFpJYTFGc1RtVkpWek5CUW\
5VMVdsaGtUV3d4WTBWeGQyTkpRV3hJUmxjMFFuSnNSMkpQTFVSU1ZFdG1lVU5QUjNoVF\
Z6UTVMV3QwU21OeVZteFpaMHR4UXpSNGJWcHZlVEJSSW4xZGZRPT0iLCJjcmVhdGVkLW\
9uIjoiMjAyMi0wNy0wOFQwODo0MDo0Mi44NDhaIn19",
  "signatures": [{
    "protected": "eyJ4NWMiOlsiTUlJQm96Q0NBVXFnQXdJQkFnSUdBVzBlTHVJRk\
1Bb0dDQ3FHU000OUJBTUNNRFV4RXpBUkJnTlZCQW9NQ2sxNVFuVnphVzVsYzNNeERUQU\
xCZ05WQkFjTUJGTnBkR1V4RHpBTkJnTlZCQU1NQmxSbGMzUkRRVEFlRncweE9UQTVNVE\
V3TWpNM016SmFGdzB5T1RBNU1URXdNak0zTXpKYU1GUXhFekFSQmdOVkJBb01DazE1UW\
5WemFXNWxjM014RFRBTEJnTlZCQWNNQkZOcGRHVXhMakFzQmdOVkJBTU1KVkpsWjJsem\
RISmhjaUJXYjNWamFHVnlJRkpsY1hWbGMzUWdVMmxuYm1sdVp5QkxaWGt3V1RBVEJnY3\
Foa2pPUFFJQkJnZ3Foa2pPUFFNQkJ3TkNBQVQ2eFZ2QXZxVHoxWlVpdU5XaFhwUXNrYV\
B5N0FISFFMd1hpSjBpRUx0NnVOUGFuQU4wUW5XTVlPXC8wQ0RFaklrQlFvYnc4WUtxan\
R4SkhWU0dUajlLT295Y3dKVEFUQmdOVkhTVUVEREFLQmdnckJnRUZCUWNESERBT0JnTl\
ZIUThCQWY4RUJBTUNCNEF3Q2dZSUtvWkl6ajBFQXdJRFJ3QXdSQUlnWXIyTGZxb2FDS0\
RGNFJBY01tSmkrTkNacWRTaXVWdWdJU0E3T2hLUnEzWUNJRHhuUE1NbnBYQU1UclBKdV\
BXeWNlRVIxMVB4SE9uKzBDcFNIaTJxZ3BXWCIsIk1JSUJwRENDQVVtZ0F3SUJBZ0lHQV\
cwZUx1SCtNQW9HQ0NxR1NNNDlCQU1DTURVeEV6QVJCZ05WQkFvTUNrMTVRblZ6YVc1bG\
MzTXhEVEFMQmdOVkJBY01CRk5wZEdVeER6QU5CZ05WQkFNTUJsUmxjM1JEUVRBZUZ3MH\
hPVEE1TVRFd01qTTNNekphRncweU9UQTVNVEV3TWpNM016SmFNRFV4RXpBUkJnTlZCQW\
9NQ2sxNVFuVnphVzVsYzNNeERUQUxCZ05WQkFjTUJGTnBkR1V4RHpBTkJnTlZCQU1NQm\
xSbGMzUkRRVEJaTUJNR0J5cUdTTTQ5QWdFR0NDcUdTTTQ5QXdFSEEwSUFCT2t2a1RIdT\
hRbFQzRkhKMVVhSTcrV3NIT2IwVVMzU0FMdEc1d3VLUURqaWV4MDZcL1NjWTVQSmlidm\
dIVEIrRlwvUVRqZ2VsSEd5MVlLcHdjTk1jc1N5YWpSVEJETUJJR0ExVWRFd0VCXC93UU\

```
lNQVlCQWY4Q0FRRXdEZ1lEVlIwUEFRSFwvQkFRREFnSUVNQjBHQTFVZERnUVdCQlRvWk\
lNelFkc0RcL2pcLytnWFwvN2NCSnVjSFwvWG1qQUtCZ2dxaGtqT1BRUURBZ05KQURCR0\
FpRUF0eFEzK0lMR0JQSXRTaDRiOVdYaFhOdWhxU1A2SCtiXC9MQ1wvZlZZRGpRNm9DSV\
FERzJ1UkNIbFZxM3loQjU4VFhNVWJ6SDgrT2xoV1V2T2xSRDNWRXFEZGNRdz09Il0sIn\
R5cCI6InZvdWNoZXItandzK2pzb24iLCJhbGciOiJFUzI1NiJ9",
    "signature": "0fzuqVdyhemWsu_HQeF-CmQwJeLp9IStNf-bWZwz6SojrEOR4a\
Dq6VStyG8eWXjGHNZiRyyLJo7RP1rKatuS2w"
  }]
}
```

Figure 5: Example Parboiled Registrar-Voucher-Request (RVR)

8.3. Example Voucher Response

The following is an example voucher response as JWS Voucher artifact, which would be sent from the MASA to the Pledge via Registrar.

```
{
  "payload": "eyJpZXRmLXZvdWNoZXI6dm91Y2hlciI6eyJhc3NlcnRpb24iOiJsb2\
dnZWQiLCJzZXJpYWwtbnVtYmVyIjoiMDEyMzQ1Njc4OSIsIm5vbmNlIjoiZGRoSGQ4Ml\
FpUGtzMDBTck1USTlEUT09IiwiY3JlYXRlZC1vbiI6IjIwMjItMDctMDdUMTc6NDc6MD\
EuODkwWiIsInBpbm5lZC1kb21haW4tY2VydCI6Ik1JSUJwRENDQVVtZ0F3SUJBZ0lHQV\
cwZUx1SCtNQW9HQ0NxR1NNNDlCQU1DTURVeEV6QVJCZ05WQkFvTUNrMTVRblZ6YVc1bG\
MzTXhEVEFMQmdOVkJBY01CRk5wZEdVeER6QU5CZ05WQkFNTUJsUmxjM1JEUVRBZUZ3MH\
hPVEE1TVRFd01qTTNNekphRncweU9UQTVNVEV3TWpNM016SmFNRFV4RXpBUkJnTlZCQW\
9NQ2sxNVFuVnphVzVsYzNNeERUQUxCZ05WQkFjTUJGTnBkR1V4RHpBTkJnTlZCQU1NQm\
xSbGMzUkRRVEJaTUJNR0J5cUdTTTQ5QWdFR0NDcUdTTTQ5QXdFSEEwSUFCT2t2a1RIdT\
hRbFQzRkhKMVVhSTcrV3NIT2IwVVMzU0FMdEc1d3VLUURqaWV4MDYvU2NZNVBKaWJ2Z0\
hUQitGL1FUamdlbEhHeTFZS3B3Y05NY3NTeWFqUlRCRE1CSUdBMVVkRXdFQi93UUlNQV\
lCQWY4Q0FRRXdEZ1lEVlIwUEFRSC9CQVFEQWdJRU1CMEdBMVVkRGdRV0JCVG9aSU16UW\
RzRC9qLytnWC83Y0JKdWNIL1htakFLQmdncWhrak9QUVFEQWdOSkFEQkdBaUVBdHhRMy\
tJTEdCUEl0U2g0YjlXWGhYTnVocVNQNkgrYi9MQy9mVllEalE2b0NJUURHMnVSQ0hsVn\
EzeWhCNThUWE1VYnpIOCtPbGhXVXZPbFJEM1ZFcURkY1F3PT0ifX0",
  "signatures": [{
    "protected": "eyJ4NWMiOlsiTUlJQmt6Q0NBVGlnQXdJQkFnSUdBV0ZCakNrWU\
1Bb0dDQ3FHU000OUJBTUNNRDB4Q3pBSkJnTlZCQVlUQWtGUk1SVXdFd1lEVlFRS0RBeE\
thVzVuU21sdVowTnZjbkF4RnpBVkJnTlZCQU1NRGtwcGJtZEthVzVuVkdWemRFTkJNQj\
RYRFRFNE1ERXlPVEV3TlRJME1Gb1hEVEk0TURFeU9URXdOVEkwTUZvd1R6RUxNQWtHQT\
FVRUJoTUNRVkV4RlRBVEJnTlZCQW9NREVwcGJtZEthVzVuUTI5eWNERXBNQ2NHQTFVRU\
F3d2dTbWx1WjBwcGJtZERiM0p3SUZadmRXTm9aWElnVTJsbmJtbHVaeUJMWlhrd1dUQV\
RCZ2NxaGtqT1BRSUJCZ2dxaGtqT1BRTUJCd05DQUFTQzZiZUxBbWVxMVZ3NmlRclJzOF\
IwWlcrNGIxR1d5ZG1XczJHQU1GV3diaXRmMm5JWEgzT3FIS1Z1OHMyUnZpQkdOaXZPS0\
dCSEh0QmRpRkVaWnZiN294SXdFREFPQmdOVkhROEJBZjhFQkFNQ0I0QXdDZ1lJS29aSX\
pqMEVBd0lEU1FBd1JnSWhBSTRQWWJ4dHNzSFAyVkh4XC90elVvUVwvU3N5ZEwzMERRSU\
5FdGNOOW1DVFhQQWlFQXZJYjNvK0ZPM0JUbmNMRnNhSlpSQWtkN3pPdXNuXC9cL1pLT2\
FFS2JzVkRpVT0iXSwidHlwIjoidm91Y2hlci1qd3MranNvbiIsImFsZyI6IkVTMjU2In\
0",
    "signature": "y1HLYBFlwouf42XWSKUWjeYQHnG2Q6A4bjA7hvTkB3z1dPwTUl\
jPHtuN2Qex6gDxTfaSiKeoXGsOD4JWOgQJPg"
  }]
}
```


Figure 6: Example Voucher Response

9. References

9.1. Normative References

[BRSKI] Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructure (BRSKI)", RFC 8995, DOI 10.17487/RFC8995, May 2021, <https://www.rfc-editor.org/rfc/rfc8995>.

[I-D.ietf-anima-rfc8366bis]
Watsen, K., Richardson, M., Pritikin, M., Eckert, T. T., and Q. Ma, "A Voucher Artifact for Bootstrapping Protocols", Work in Progress, Internet-Draft, draft-ietf-anima-rfc8366bis-12, 8 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-anima-rfc8366bis-12>.

[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/rfc/rfc2119>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC7515] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Signature (JWS)", RFC 7515, DOI 10.17487/RFC7515, May 2015, <https://www.rfc-editor.org/rfc/rfc7515>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/rfc/rfc8259>.

[RFC8995] Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructure (BRSKI)", RFC 8995, DOI 10.17487/RFC8995, May 2021, <https://www.rfc-editor.org/rfc/rfc8995>.

9.2. Informative References

[I-D.ietf-anima-brski-prm]
Fries, S., Werner, T., Lear, E., and M. Richardson, "BRSKI with Pledge in Responder Mode (BRSKI-PRM)", Work in Progress, Internet-Draft, draft-ietf-anima-brski-prm-15, 26 August 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-anima-brski-prm-15>.

[I-D.ietf-anima-constrained-voucher]
Richardson, M., Van der Stok, P., Kampanakis, P., and E. Dijk, "Constrained Bootstrapping Remote Secure Key Infrastructure (cBRSKI)", Work in Progress, Internet-Draft, draft-ietf-anima-constrained-voucher-25, 8 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-anima-constrained-voucher-25>.

[ON-PATH] "can an on-path attacker drop traffic?", n.d., <https://mailarchive.ietf.org/arch/msg/saag/m1r9uo4xYznOcf85Eyk0Rhut598/>.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <https://www.rfc-editor.org/rfc/rfc3629>.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/rfc/rfc5652>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/rfc/rfc7950>.

[RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, DOI 10.17487/RFC7951, August 2016, <https://www.rfc-editor.org/rfc/rfc7951>.

[RFC8366] Watsen, K., Richardson, M., Pritikin, M., and T. Eckert, "A Voucher Artifact for Bootstrapping Protocols", RFC 8366, DOI 10.17487/RFC8366, May 2018, <https://www.rfc-editor.org/rfc/rfc8366>.

[RFC8572] Watsen, K., Farrer, I., and M. Abrahamsson, "Secure Zero Touch Provisioning (SZTP)", RFC 8572, DOI 10.17487/RFC8572, April 2019, <https://www.rfc-editor.org/rfc/rfc8572>.

[RFC8792] Watsen, K., Auerswald, E., Farrel, A., and Q. Wu, "Handling Long Lines in Content of Internet-Drafts and RFCs", RFC 8792, DOI 10.17487/RFC8792, June 2020, <https://www.rfc-editor.org/rfc/rfc8792>.

[RFC8812] Jones, M., "CBOR Object Signing and Encryption (COSE) and JSON Object Signing and Encryption (JOSE) Registrations for Web Authentication (WebAuthn) Algorithms", RFC 8812, DOI 10.17487/RFC8812, August 2020, <https://www.rfc-editor.org/rfc/rfc8812>.

Contributors

Toerless Eckert
Futurewei Technologies Inc.
Email: tte+ietf@cs.fau.de

Esko Dijk
Email: esko.dijk@iotconsultancy.nl

Steffen Fries
Siemens AG
Email: steffen.fries@siemens.com

Authors' Addresses

Thomas Werner
Siemens AG
Email: thomas-werner@siemens.com

Michael Richardson
Sandelman Software Works
Email: mcr+ietf@sandelman.ca

BESS WorkGroup
Internet-Draft
Updates: 7432 (if approved)
Intended status: Standards Track
Expires: 19 April 2025

N. Malhotra, Ed.
A. Sajassi
A. Pattekar
Cisco Systems
J. Rabadan
Nokia
A. Lingala
AT&T
J. Drake
Juniper Networks
16 October 2024

Extended Mobility Procedures for EVPN-IRB
draft-ietf-bess-evpn-irb-extended-mobility-18

Abstract

This document specifies extensions to Ethernet VPN (EVPN) Integrated Routing and Bridging (IRB) procedures specified in RFC7432 and RFC9135 to enhance the mobility mechanisms for EVPN IRB-based networks. The proposed extensions improve the handling of host mobility and duplicate address detection in EVPN-IRB networks to cover a broader set of scenarios where host IP to MAC bindings may change across moves. These enhancements address limitations in the existing EVPN IRB mobility procedures by providing more efficient and scalable solutions. The extensions are backward compatible with existing EVPN IRB implementations and aim to optimize network performance in scenarios involving frequent IP address mobility.

NOTE TO IESG (TO BE DELETED BEFORE PUBLISHING): This draft lists six authors which is above the required limit of five. Given significant and active contributions to the draft from all six authors over the course of six years, we would like to request IESG to allow publication with six authors. Specifically, the three Cisco authors are the original inventors of these procedures and contributed heavily to rev 0 draft, most of which is still intact. AT&T is also a key contributor towards defining the use cases that this document addresses as well as the proposed solution. Authors from Nokia and Juniper have further contributed to revisions and discussions steadily over last six years to enable respective implementations and a wider adoption.

Status of This Memo

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

Copyright Notice

Table of Contents

1. Introduction

EVPN-IRB facilitates the advertisement of both MAC and IP routes via a single MAC+IP Route Type 2 (RT-2) advertisement. The MAC address is integrated into the local MAC-VRF bridge table, enabling Layer 2 (L2) bridged traffic across the network overlay. The IP address is incorporated into the local ARP table in an asymmetric IRB design, or into the IP-VRF routing table in a symmetric IRB design, facilitating routed traffic across the network overlay. For additional context on EVPN IRB forwarding modes, refer to [RFC9135].

To support the EVPN mobility procedure, a single sequence number mobility attribute is advertised with the combined MAC+IP route. This approach, which resolves both MAC and IP reachability with a single sequence number, inherently assumes a fixed 1:1 mapping between IP and MAC. While this fixed 1:1 mapping is a common use case and is addressed via the existing MAC mobility procedure defined in [RFC7432], there are additional IRB scenarios that do not adhere to this assumption. Such scenarios are prevalent in virtualized host environments where hosts connected to an EVPN network are virtual machines (VMs) or containerized workloads. The following IRB mobility scenarios are considered:

* A VM move results in the VM's IP and MAC moving together.

* A VM move results in the VM's IP moving to a new MAC association.

* A VM move results in the VM's MAC moving to a new IP association.

While the existing MAC mobility procedure can manage the MAC+IP move in the first scenario, the subsequent scenarios lead to new MAC-IP associations. Therefore, a single sequence number assigned independently per-{MAC, IP} is insufficient to determine the most recent reachability for both MAC and IP unless the sequence number assignment algorithm allows for changing MAC-IP bindings across moves.

This document updates the sequence number assignment procedures defined in [RFC7432] to adequately address mobility support across EVPN-IRB overlay use cases that permit MAC-IP bindings to change across VM moves and support mobility for both MAC and IP components carried in an EVPN RT-2 for these use cases.

Additionally, for hosts on an ESI multi-homed to multiple PE devices, additional procedures are specified to ensure synchronized sequence number assignments across the multi-homing devices.

This document addresses mobility for the following cases, independent of the overlay encapsulation (e.g., MPLS, SRv6, NVO Tunnel):

* Symmetric EVPN IRB overlay

* Asymmetric EVPN IRB overlay

* Routed EVPN overlay

1.1. Document Structure

Following sections of the document are informative:

* section 3 provides the necessary background and problem statement being addressed in this document.

* section 4 lists the resulting design considerations for the document.

* section 5 lists the main solution components that are foundational for the sepecifications that follow in subsequent sections.

Following sections of the document are normative:

* section 6 describes the mobility and sequence number assigment procedures in an EVPN-IRB overlay required to address the scenarios described in section 4.

* section 7 describes the mobility procedures for a routed overlay network as opposed to an IRB overlay.

* section 8 describes corresponding duplicate detection procedures for EVPN-IRB and routed overlays.

2. Requirements Language and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

* EVPN-IRB: A BGP-EVPN distributed control plane based integrated routing and bridging fabric overlay discussed in [RFC9135]

* Underlay: IP, MPLS, or SRv6 fabric core network that provides routed reachability between EVPN PEs.

* Overlay: L3 and L2 Virtual Private Network (VPN) enabled via NVO, SRv6, or MPLS service layer encapsulation.

* EVPN PE: A PE switch-router in a data-center fabric that runs overlay BGP-EVPN control plane and connects to overlay CE host devices. An EVPN PE may also be the first-hop layer-3 gateway for CE/host devices. This document refers to EVPN PE as a logical function in a data-center fabric. This EVPN PE function may be physically hosted on a top-of-rack switching device (ToR) OR at layer(s) above the ToR in the Clos fabric. An EVPN PE is typically also an IP or MPLS tunnel end-point for overlay VPN flow

* Symmetric EVPN-IRB: is a specific design approach used in EVPN-based networks [RFC9135] to handle both Layer 2 (L2) and Layer 3 (L3) forwarding within the same network infrastructure. The key characteristic of symmetric EVPN-IRB is that both ingress and egress PE routers perform routing for inter-subnet traffic.

* Asymmetric EVPN-IRB: is a design approach used in EVPN-based networks [RFC9135] to handle Layer 2 (L2) and Layer 3 (L3) forwarding. In this approach, only the ingress Provider Edge (PE) router performs routing for inter-subnet traffic, while the egress PE router performs bridging.

* ARP: Address Resolution Protocol [RFC826]. ARP references in this document are equally applicable to ND as well.

* ND: IPv6 Neighbor Discovery Protocol [RFC4861].

* Ethernet-Segment: Physical ethernet or LAG port that connects an access device to an EVPN PE, as defined in [RFC7432].

* EVPN all-active multi-homing: is a redundancy and load-sharing mechanism used in EVPN networks. This method allows multiple PE devices to simultaneously provide Layer 2 and Layer 3 connectivity to a single CE device or network segment.

* RT-2: EVPN route type 2 carrying both MAC and IP reachability as specied in [RFC7432].

* RT-5: EVPN route type 5 carrying IP prefix reachability as specified in [RFC7432].

* MAC-IP: IPv4 and/or IPv6 address and MAC binding for an overlay host.

* SYNC MAC route: In the context of EVPN multi-homing, this refers to a local MAC route SYNCed from another PE sharing the same ESI.

* SYNC MAC-IP route: In the context of EVPN multi-homing, this refers to a local MAC-IP route SYNCed from another PE sharing the same ESI.

* SYNC MAC sequence number: In the context of EVPN multi-homing, this refers to sequence number received with a SYNC MAC route.

* SYNC MAC-IP sequence number: In the context of EVPN multi-homing, this refers to sequence number received with a SYNC MAC-IP route.

* VM: Virtual Machine or containerized workloads

3. Background and Problem Statement

3.1. Optional MAC only RT-2

In an EVPN IRB scenario, where a single MAC+IP RT-2 advertisement carries both IP and MAC routes, a MAC-only RT-2 advertisement becomes redundant for host MACs already advertised via MAC+IP RT-2. Consequently, the advertisement of a local MAC-only RT-2 is optional at an EVPN PE. This consideration is important for mobility scenarios discussed in subsequent sections. It is noteworthy that a local MAC and its assigned sequence number are still maintained

locally on a PE, and only the advertisement of this route to other PEs is optional.

MAC-only RT-2 advertisements may still be issued for non-IP host MACs that are not included in MAC+IP RT-2 advertisements.

3.2. Mobility Use Cases

This section outlines the IRB mobility use cases addressed in this document. Detailed procedures to handle these scenarios are provided in Sections 6 and 7.

* A host move results in both the host's IP and MAC addresses moving together.

* A host move results in the host's IP address moving to a new MAC address association.

* A host move results in the host's MAC address moving to a new IP address association.

3.2.1. Host MAC+IP Move

This is the baseline scenario where a host move results in both the host's MAC and IP addresses moving together without altering the MAC-IP binding. The existing MAC mobility procedures defined in [RFC7432] can be leveraged to support this MAC+IP mobility scenario.

3.2.2. Host IP Move to new MAC

This scenario involves a host move where the host's IP address is reassigned to a new MAC address.

3.2.2.1. VM Reload

A host reload or orchestrated move may cause a host to be re-spawned at a new location, resulting in a new MAC assignment while retaining the existing IP address. This results in the host's IP moving to a new MAC binding, as shown below:

IP-a, MAC-a ---> IP-a, MAC-b

3.2.2.2. MAC Sharing

This scenario considers cases where multiple hosts, each with a unique IP address, share a common MAC address. A host move results in a new MAC binding for the host IP. For example, hosts running on a single physical server might share the same MAC. Alternatively, an L2 access network behind a firewall may have all host IPs learned with a common firewall MAC. In these "shared MAC" scenarios, multiple local MAC-IP ARP entries may be learned with the same MAC. A host IP move to a new physical server could result in a new MAC association for the host IP.

3.2.2.3. Problem

In the aforementioned scenarios, a combined MAC+IP EVPN RT-2 advertised with a single sequence number attribute assumes a fixed IP-to-MAC mapping. A host IP move to a new MAC breaks this assumption and results in a new MAC+IP route. If this new route is independently assigned a new sequence number, the sequence number can no longer determine the most recent host IP reachability in a symmetric EVPN-IRB design or the most recent IP-to-MAC binding in an asymmetric EVPN-IRB design.

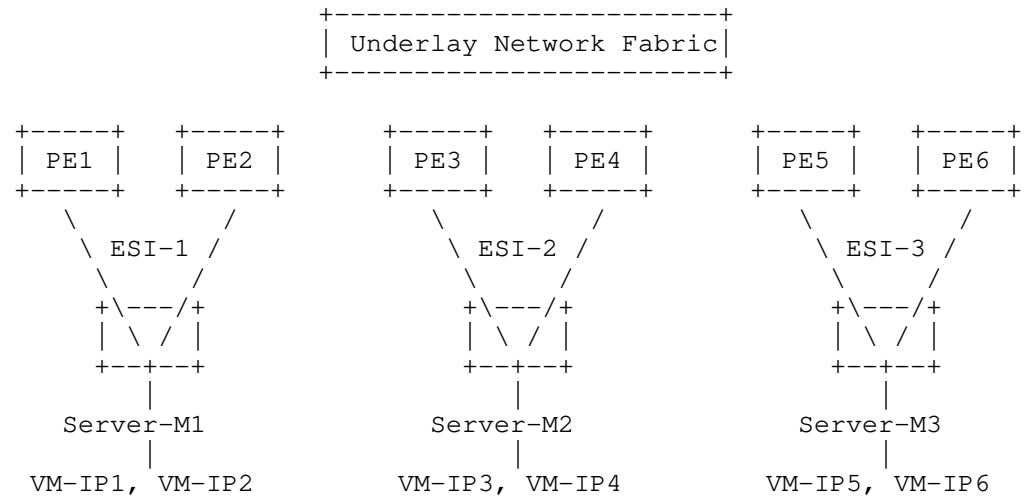


Figure 1

Figure 1 illustrates a topology with host VMs sharing the physical server MAC. In steady state, the IP1-M1 route is learned at PE1 and PE2 and advertised to remote PEs with a sequence number N. If VM-IP1 moves to Server-M2, ARP or ND-based local learning at PE3 and PE4

would result in a new IP1-M2 route. If this new route is assigned a sequence number of 0, the mobility procedure for VM-IP1 will not trigger across the overlay network.

A sequence number assignment procedure must be defined to unambiguously determine the most recent IP reachability, IP-to-MAC binding, and MAC reachability for such MAC sharing scenarios.

3.2.3. Host MAC move to new IP

This is a scenario where a host move or re-provisioning behind a new gateway location may result in the host getting a new IP address assigned, while keeping the same MAC.

3.2.3.1. Problem

The complication in this scenario arises because MAC reachability can be carried via a combined MAC+IP route, whereas a MAC-only route may not be advertised. Associating a single sequence number with the MAC+IP route implicitly assumes a fixed MAC-to-IP mapping. A MAC move that results in a new IP association breaks this assumption and creates a new MAC+IP route. If this new route independently receives a new sequence number, the sequence number can no longer reliably indicate the most recent host MAC reachability.

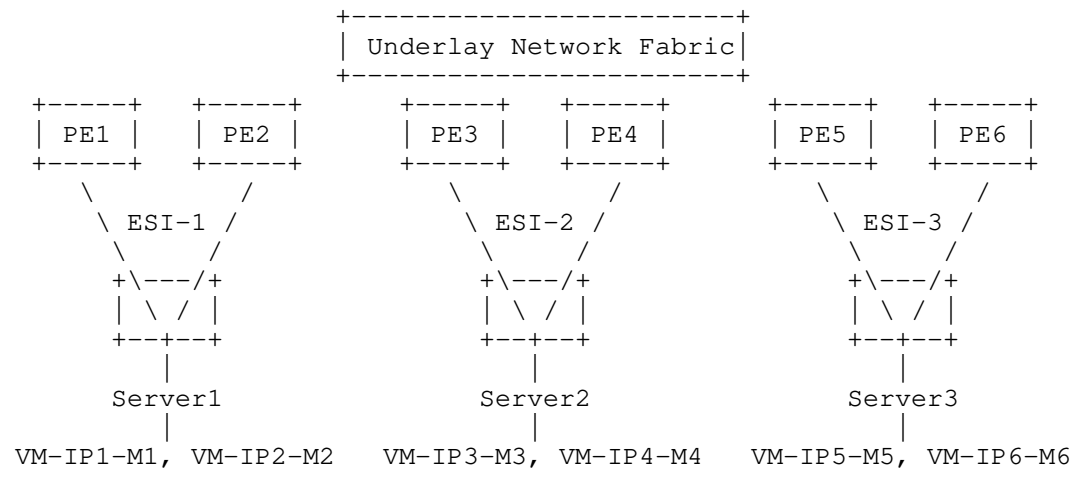


Figure 2

For instance, consider host VM IP1-M1 learned locally at PE1 and PE2 and advertised to remote hosts with sequence number N. If this VM with MAC M1 is re-provisioned at Server2 and assigned a different IP

address (e.g., IP7), the new IP7-M1 route learned at PE3 and PE4 would be advertised with sequence number 0. Consequently, L3 reachability to IP7 would be established across the overlay, but the MAC mobility procedure for M1 would not trigger due to the new MAC-IP route advertisement. Advertising an optional MAC-only route with its sequence number would trigger MAC mobility per [RFC7432]. However, without this additional advertisement, a single sequence number associated with a combined MAC+IP route may be insufficient to update MAC reachability across the overlay.

A MAC-IP sequence number assignment procedure is required to unambiguously determine the most recent MAC reachability in such scenarios without advertising a MAC-only route.

Furthermore, PE1 and PE2, upon learning new reachability for IP7-M1 via PE3 and PE4, must probe and delete any local IPs associated with MAC M1, such as IP1-M1.

It could be argued that the MAC mobility sequence number defined in [RFC7432] applies only to the MAC part of a MAC-IP route, thus covering this scenario. This interpretation could serve as a clarification to [RFC7432] and supports the need for a common sequence number assignment procedure across all MAC-IP mobility scenarios detailed in this document.

3.3. EVPN All Active multi-homed ES

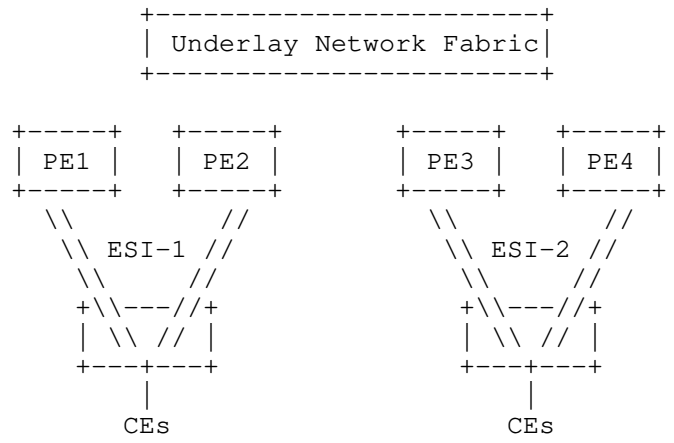


Figure 3

Consider an EVPN-IRB overlay network illustrated in Figure 3, where hosts are multi-homed to two or more PE devices via an all-active multi-homed ES. MAC and ARP entries learned on a local ES may also

be synchronized across the multi-homing PE devices sharing this ES. This synchronization enables local switching of intra- and inter-subnet ECMP traffic flows from remote hosts. Thus, local MAC and ARP entries on a given ES may be learned through local learning and/or synchronization from another PE device sharing the same ES.

For a host that is multi-homed to multiple PE devices via an all-active ES interface, the local learning of host MAC and MAC-IP at each PE device is an independent asynchronous event, dependent on traffic flow or ARP/ND response from the host hashing to a directly connected PE on the MC-LAG interface. Consequently, the sequence number mobility attribute value assigned to a locally learned MAC or MAC-IP route at each device may not always be the same, depending on transient states on the device at the time of local learning.

For example, consider a host VM that is deleted from ESI-2 and moved to ESI-1. It is possible for the host to be learned on PE1 following the deletion of the remote route from PE3 and PE4, while being learned on PE2 prior to the deletion of the remote route from PE3 and PE4. In this case, PE1 would process local host route learning as a new route and assign a sequence number of 0, while PE2 would process local host route learning as a remote-to-local move and assign a sequence number of N+1, where N is the existing sequence number assigned at PE3 and PE4.

Inconsistent sequence numbers advertised from multi-homing devices:

* Creates ambiguity regarding how remote PEs should handle paths with the same ESI but different sequence numbers. A remote PE might not program ECMP paths if it receives routes with different sequence numbers from a set of multi-homing PEs sharing the same ESI.

* Breaks consistent route versioning across the network overlay that is needed for EVPN mobility procedures to work.

For instance, in this inconsistent state, PE2 would drop a remote route received for the same host with sequence number N (since its local sequence number is N+1), while PE1 would install it as the best route (since its local sequence number is 0).

To support mobility for multi-homed hosts using the sequence number mobility attribute, local MAC and MAC-IP routes learned on a multi-homed ES must be advertised with the same sequence number by all PE devices to which the ES is multi-homed. There is a need for a mechanism to ensure the consistency of sequence numbers assigned across these PEs.

4. Design Considerations

To summarize, the sequence number assignment scheme and implementation must consider the following:

* Synchronization Across Multi-Homing PE Devices: MAC+IP may be learned on an ES multi-homed to multiple PE devices, requiring synchronized sequence numbers across these devices.

* Optional MAC-Only RT-2: In an IRB scenario, MAC-only RT-2 is optional and may not be advertised alongside MAC+IP RT-2.

* Multiple IPs Associated with a Single MAC: A single MAC may be linked to multiple IP addresses, indicating multiple host IPs sharing a common MAC.

* Host IP Movement: A host IP move may result in a new MAC association, necessitating a new IP to MAC association and a new MAC+IP route.

* Host MAC Movement: A host MAC move may result in a new IP association, requiring a new MAC to IP association and a new MAC+IP route.

* Local MAC-IP Learning via ARP: Local MAC-IP learning via ARP always accompanies a local MAC learning event resulting from the ARP packet. However, MAC and MAC-IP learning can occur in any order.

* Separate Sequence Numbers for MAC and IP: Use cases that do not maintain a constant 1:1 MAC-IP mapping across moves could potentially be addressed by using separate sequence numbers for MAC and IP components of the MAC+IP route. However, maintaining two separate sequence numbers adds significant complexity, debugging challenges, and backward compatibility issues. Therefore, this document addresses these requirements using a single sequence number attribute.

5. Solution Components

This section outlines the main components of the EVPN IRB mobility solution specified in this document. Subsequent sections will detail the exact sequence number assignment procedures based on the concepts described here.

5.1. Sequence Number Inheritance

The key concept presented here is to treat a local MAC-IP route as a child of the corresponding local MAC route within the local context of a PE. This ensures that the local MAC-IP route inherits the sequence number attribute from the parent local MAC-only route. In terms of object dependencies, this could be represented as MAC-IP route being a dependent child of the parent MAC:

Mx-IPx -----> Mx (seq# = N)

Thus, both the parent MAC and child MAC-IP routes share a common sequence number associated with the parent MAC route. This ensures that a single sequence number attribute carried in a combined MAC+IP route represents the sequence number for both a MAC-only route and a MAC+IP route, making the advertisement of the MAC-only route truly optional. This enables a MAC to assume a different IP address upon moving and still establish the most recent reachability to the MAC across the overlay network via the mobility attribute associated with the MAC+IP route advertisement. For instance, when Mx moves to a new location, it would be assigned a higher sequence number at its new location per [RFC7432]. If this move results in Mx assuming a different IP address, IPz, the local Mx+IPz route would inherit the new sequence number from Mx.

Local MAC and local MAC-IP routes are typically sourced from data plane learning and ARP learning, respectively, and can be learned in the control plane in any order. Implementation can either replicate the inherited sequence number in each MAC-IP entry or maintain a single attribute in the parent MAC by creating a forward reference local MAC object for cases where a local MAC-IP is learned before the local MAC.

5.2. MAC Sharing

For the shared MAC scenario, multiple local MAC-IP siblings inherit the sequence number attribute from the common parent MAC route:

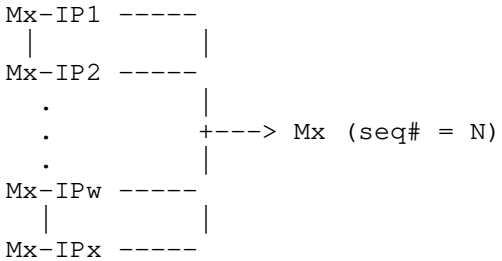


Figure 4

In such cases, a host-IP move to a different physical server results in the IP moving to a new MAC binding. A new MAC-IP route resulting from this move must be advertised with a sequence number higher than the previous MAC-IP route for this IP, advertised from the prior location. For example, consider a route Mx-IPx currently advertised with sequence number N from PE1. If IPx moves to a new physical server behind PE2 and is associated with MAC Mz, the new local Mz-IPx route must be advertised with a sequence number higher than N and the previous Mz sequence number M. This allows PE devices, including PE1, PE2, and other remote PE devices, to determine and program the most recent MAC binding and reachability for the IP. PE1, upon receiving this new Mz-IPx route with sequence number N+1, would update IPx reachability via PE2 for symmetric IRB and update IPx's ARP binding to Mz for asymmetric IRB, while clearing and withdrawing the stale Mx-IPx route with the lower sequence number.

This implies that the sequence number associated with local MAC Mz and all local MAC-IP children of Mz at PE2 must be incremented to N+1 or M+1 if the previous Mz sequence number M is greater than N and re-advertised across the overlay. While this re-advertisement of all local MAC-IP children routes affected by the parent MAC route adds overhead, it avoids the need for maintaining and advertising two separate sequence number attributes for IP and MAC components of MAC+IP RT-2. Implementation must be able to look up MAC-IP routes for a given IP and update the sequence number for its parent MAC and its MAC-IP children.

5.3. Multi-homing Mobility Synchronization

To support mobility for multi-homed hosts, local MAC and MAC-IP routes learned on a shared ES must be advertised with the same sequence number by all PE devices to which the ES is multi-homed. This applies to local MAC-only routes as well. Local MAC and MAC-IP may be learned natively via data plane and ARP/ND respectively, as well as via SYNC from another multi-homing PE to achieve local switching. Local and SYNC route learning can occur in any order. Local MAC-IP routes advertised by all multi-homing PE devices sharing the ES must carry the same sequence number, independent of the order in which they are learned. This implies:

* On local or SYNC MAC-IP route learning, the sequence number for the local MAC-IP route must be compared and updated to the higher value.

* On local or SYNC MAC route learning, the sequence number for the local MAC route must be compared and updated to the higher value.

If an update to the local MAC-IP sequence number is required as a result of the comparison with the SYNC MAC-IP route, it essentially amounts to a sequence number update on the parent local MAC, resulting in an inherited sequence number update on the MAC-IP route.

6. Requirements for Sequence Number Assignment

The following sections specify the sequence number assignment procedures required for local and SYNC MAC and MAC-IP route learning events to achieve the objectives outlined.

6.1. Local MAC-IP learning

A local Mx-IPx learning via ARP or ND should result in the computation or re-computation of the parent MAC Mx's sequence number, following which the MAC-IP route Mx-IPx inherits the parent MAC's sequence number. The parent MAC Mx sequence number MUST be computed as follows:

* MUST be higher than any existing remote MAC route for Mx, as per [RFC7432].

* MUST be at least equal to the corresponding SYNC MAC sequence number, if present.

* If the IP is also associated with a different remote MAC "Mz," it MUST be higher than the "Mz" sequence number.

Once the new sequence number for MAC route Mx is computed as per the above criteria, all local MAC-IPs associated with MAC Mx MUST inherit the updated sequence number.

6.2. Local MAC learning

The local MAC Mx Sequence number MUST be computed as follows:

* MUST be higher than any existing remote MAC route for Mx, as per [RFC7432].

* MUST be at least equal to the corresponding SYNC MAC sequence number if one is present. If the existing local MAC sequence number if less than the SYNC MAC sequence number, PE MUST update the local MAC sequence number to be equal to the SYNC MAC sequence number. If the existing local MAC sequence number is equal to or greater than the SYNC MAC sequence number, no update is required to the local MAC sequence number.

Once the new sequence number for MAC route Mx is computed as per the above criteria, all local MAC-IPs associated with MAC Mx MUST inherit the updated sequence number. Note that the local MAC sequence number might already be present if there was a local MAC-IP learned prior to the local MAC, in which case the above may not result in any change in the local MAC's sequence number.

6.3. Remote MAC or MAC-IP Update

Upon receiving a remote MAC or MAC-IP route update associated with a MAC Mx with a sequence number that is:

* Either higher than the sequence number assigned to a local route for MAC Mx,

* Or equal to the sequence number assigned to a local route for MAC Mx, but the remote route is selected as best due to a lower VTEP IP as per [RFC7432],

the following actions are REQUIRED on the receiving PE:

* The PE MUST trigger a probe and deletion procedure for all local IPs associated with MAC Mx.

* The PE MUST trigger a deletion procedure for the local MAC route for Mx.

6.4. REMOTE (SYNC) MAC update

Upon receiving a REMOTE SYNC, the corresponding local MAC Mx (if present) sequence number should be re-computed as follows:

* If the current sequence number is less than the received SYNC MAC sequence number, it MUST be increased to be equal to the received SYNC MAC sequence number.

* If a local MAC sequence number is updated as a result of the above, all local MAC-IPs associated with MAC Mx MUST inherit the updated sequence number.

6.5. REMOTE (SYNC) MAC-IP update

Receiving a SYNC MAC-IP for a locally attached host results in a derived SYNC MAC Mx route entry, as the MAC-only RT-2 advertisement is optional. The corresponding local MAC Mx (if present) sequence number should be re-computed as follows:

* If the current sequence number is less than the received SYNC MAC sequence number, it MUST be increased to be equal to the received SYNC MAC sequence number.

* If a local MAC sequence number is updated as a result of the above, all local MAC-IPs associated with MAC Mx MUST inherit the updated sequence number.

6.6. Interoperability

Generally, if all PE nodes in the overlay network follow the above sequence number assignment procedures and the PE is advertising both MAC+IP and MAC routes, the sequence numbers advertised with the MAC and MAC+IP routes with the same MAC would always be the same. However, an interoperability scenario with a different implementation could arise, where a non-compliant PE implementation assigns and advertises independent sequence numbers to MAC and MAC+IP routes. To handle this case, if different sequence numbers are received for remote MAC+IP and corresponding remote MAC routes from a remote PE, the sequence number associated with the remote MAC route MUST be computed and interpreted as:

* The highest of all received sequence numbers with remote MAC+IP and MAC routes with the same MAC.

* The MAC sequence number would be re-computed on a MAC or MAC+IP route withdraw as per the above.

A MAC and/or IP move to the local PE would then result in the MAC (and hence all MAC-IP) sequence numbers being incremented from the above computed remote MAC sequence number.

If MAC-only routes are not advertised at all, and different sequence numbers are received with multiple MAC+IP routes for a given MAC, the sequence number associated with the derived remote MAC route should still be computed as the highest of all received MAC+IP sequence numbers with the same MAC.

Note that it is not required for a PE to maintain explcit knowledge of a remote PE being compliant or non-compliant with this specification as long as it implements the above logic to handle remote sequence numbers that are not synchronized between MAC route and MAC-IP route(s) for the same remote MAC.

6.7. MAC Sharing Race Condition

In a MAC sharing use case described in section 5.2, a race condition is possible with simultaneous host moves between a pair of PEs. Example scenario below illustrates this race condition and its remediation:

* PE1 with locally attached host IPs I1 and I2 that share MAC M1. PE1 as a result has local MAC-IP routes I1-M1 and I2-M1.

* PE2 with locally attached host IPs I3 and I4 that share MAC M2. PE2 as a result has local MAC-IP routes I3-M2 and I4-M2.

* A simultaneous move of I1 from PE1 to PE2 and of I3 from PE2 to PE1 will cause I1's MAC to change from M1 to M2 and cause I3's MAC to change from M2 to M1.

* Route I3-M1 may be learnt on PE1 before I1's local entry I1-M1 has been probed out on PE1 and/or route I1-M2 may be learnt on PE2 before I3's local entry I3-M2 has been probed out on PE2.

* In such a scenario, MAC sequence number assignment rules defined in section 6.1 will cause new mac-ip routes I1-M2 and I3-M1 to bounce between PE1 and PE2 with seuence number increments until stale entries I1-M1 and I3-M2 have been probed out from PE1 and PE2 respectively.

An implementation MUST ensure proper probing procedures to remove stale ARP, ND, and local MAC entries, following a move, on learning remote routes as defined in section 6.3 (and as per [RFC9135]) to minimize exposure to this race condition.

6.8. Mobility Convergence

This section is optional and details ARP and ND probing procedures that MAY be implemented to achieve faster host re-learning and convergence on mobility events. PE1 and PE2 are used as two example PEs in the network to illustrate the mobility convergence scenarios in this section.

* Following a host move from PE1 to PE2, the host's MAC is discovered at PE2 as a local MAC via data frames received from the host. If PE2 has a prior remote MAC-IP host route for this MAC from PE1, an ARP/ND probe MAY be triggered at PE2 to learn the MAC-IP as a local adjacency and trigger EVPN RT-2 advertisement for this MAC-IP across the overlay with new reachability via PE2. This results in a reliable "event-based" host IP learning triggered by a "MAC learning event" across the overlay, and hence faster convergence of overlay routed flows to the host.

* Following a host move from PE1 to PE2, once PE1 receives a MAC or MAC-IP route from PE2 with a higher sequence number, an ARP/ND probe MAY be triggered at PE1 to clear the stale local MAC-IP neighbor adjacency or to re-learn the local MAC-IP in case the host has moved back or is duplicated.

* Following a local MAC age-out, if there is a local IP adjacency with this MAC, an ARP/ND probe MAY be triggered for this IP to either re-learn the local MAC and maintain local L3 and L2 reachability to this host or to clear the ARP/ND entry if the host is no longer local. This accomplishes the clearance of stale ARP entries triggered by a MAC age-out event even when the ARP refresh timer is longer than the MAC age-out timer. Clearing stale IP neighbor entries facilitates traffic convergence if the host was silent and not discovered at its new location. Once the stale neighbor entry for the host is cleared, routed traffic flow destined for the host can re-trigger ARP/ND discovery for this host at the new location.

6.8.1. Generalized Probing Logic

The above probing logic may be generalized as probing for an IP neighbor anytime a resolving parent MAC route is inconsistent with the MAC-IP neighbor route, where inconsistency is defined as being not present or conflicting in terms of the route source being local or remote. The MAC-IP to MAC parent relationship described in section 5.1 MAY be used to achieve this logic.

7. Routed Overlay

An additional use case involves traffic to an end host in the overlay being entirely IP routed. In such a purely routed overlay:

* A host MAC is never advertised in the EVPN overlay control plane.

* Host /32 or /128 IP reachability is distributed across the overlay via EVPN Route Type 5 (RT-5) along with a zero or non-zero ESI.

* An overlay IP subnet may still be stretched across the underlay fabric; however, intra-subnet traffic across the stretched overlay is never bridged.

* Both inter-subnet and intra-subnet traffic in the overlay is IP routed at the EVPN PE.

Please refer to [RFC7814] for more details.

Host mobility within the stretched subnet still needs support. In the absence of host MAC routes, the sequence number mobility Extended Community specified in [RFC7432], section 7.7, MAY be associated with a /32 or /128 host IP prefix advertised via EVPN Route Type 5. MAC mobility procedures defined in [RFC7432] can be applied to host IP prefixes as follows:

* On local learning of a host IP on a new ESI, the host IP MUST be advertised with a sequence number higher than what is currently advertised with the old ESI.

* On receiving a host IP route advertisement with a higher sequence number, a PE MUST trigger ARP/ND probe and deletion procedures on any local route for that IP with a lower sequence number. The PE will update the forwarding entry to point to the remote route with a higher sequence number and send an ARP/ND probe for the local IP route. If the IP has moved, the probe will time out, and the local IP host route will be deleted.

Note that there is only one sequence number associated with a host route at any time. For previous use cases where a host MAC is advertised along with the host IP, a sequence number is only associated with the MAC. If the MAC is not advertised, as in this use case, a sequence number is associated with the host IP.

This mobility procedure does not apply to "anycast IPv6" hosts advertised via NA messages with the Override Flag (O Flag) set to 0. Refer to [RFC9161] for more details.

8. Duplicate Host Detection

Duplicate host detection scenarios across EVPN IRB can be classified as follows:

* Scenario A: Two hosts have the same MAC address (host IPs may or may not be duplicates).

* Scenario B: Two hosts have the same IP address but different MAC addresses.

* Scenario C: Two hosts have the same IP address, and the host MAC is not advertised.

As specified in [RFC9161], Duplicate detection procedures for Scenarios B and C do not apply to "anycast IPv6" hosts advertised via NA messages with the Override Flag (O Flag) set to 0.

8.1. Scenario A

In cases where duplicate hosts share the same MAC address, the MAC is detected as duplicate using the duplicate MAC detection procedure described in [RFC7432]. Corresponding MAC-IP routes with the same MAC do not require separate duplicate detection and MUST inherit the duplicate property from the MAC route. If a MAC route is marked as duplicate, all associated MAC-IP routes MUST also be treated as duplicates. Duplicate detection procedures need only be applied to MAC routes.

8.2. Scenario B

Misconfigurations may lead to different MAC addresses being assigned the same IP address. This scenario is not detected by [RFC7432] duplicate MAC detection procedures and can result in incorrect routing of traffic destined for the IP address.

Such situations, when detected locally, are identified as a move scenario through the local MAC sequence number computation procedure described in section 6.1:

* If the IP is associated with a different remote MAC "Mz," the sequence number MUST be higher than the "Mz" sequence number.

This move results in a sequence number increment for the local MAC due to the remote MAC-IP route associated with a different MAC, counting as an "IP move" against the IP, independent of the MAC. The duplicate detection procedure described in [RFC7432] can then be applied to the IP entity independent of the MAC. Once an IP is detected as duplicate, the corresponding MAC-IP route should be treated as duplicate. Associated MAC routes and any other MAC-IP routes related to this MAC should not be affected.

8.2.1. Duplicate IP Detection Procedure for Scenario B

The duplicate IP detection procedure for this scenario is specified in [RFC9161]. An "IP move" is further clarified as follows:

* Upon learning a local MAC-IP route Mx-IPx, check for existing remote or local routes for IPx with a different MAC association (Mz-IPx). If found, count this as an "IP move" for IPx, independent of the MAC.

* Upon learning a remote MAC-IP route Mz-IPx, check for existing local routes for IPx with a different MAC association (Mx-IPx). If found, count this as an "IP move" for IPx, independent of the MAC.

A MAC-IP route SHOULD be treated as duplicate if either:

* The corresponding MAC route is marked as duplicate via the existing detection procedure.

* The corresponding IP is marked as duplicate via the extended procedure described above.

8.3. Scenario C

In a purely routed overlay scenario, as described in section 7, where only a host IP is advertised via EVPN RT-5 with a sequence number mobility attribute, duplicate MAC detection procedures specified in [RFC7432] can be applied intuitively to IP-only host routes for duplicate IP detection.

* Upon learning a local host IP route IPx, check for existing remote or local routes for IPx with a different ESI association. If found, count this as an "IP move" for IPx.

* Upon learning a remote host IP route IPx, check for existing local routes for IPx with a different ESI association. If found, count this as an "IP move" for IPx.

* Using configurable parameters "N" and "M," if "N" IP moves are detected within "M" seconds for IPx, IPx should be treated as duplicate.

8.4. Duplicate Host Recovery

Once a MAC or IP is marked as duplicate and frozen, corrective action must be taken to un-provision one of the duplicate MAC or IP addresses. Un-provisioning refers to corrective action taken on the host side. Following this correction, normal operation will not resume until the duplicate MAC or IP ages out unless additional action is taken to expedite recovery.

Possible additional corrective actions for faster recovery include:

8.4.1. Route Un-freezing Configuration

Unfreezing the duplicate or frozen MAC or IP via a CLI can be used to recover from the duplicate and frozen state following corrective un-provisioning of the duplicate MAC or IP. Unfreezing the MAC or IP should result in advertising it with a sequence number higher than that advertised from the other location.

Two scenarios exist:

* Scenario A: The duplicate MAC or IP is un-provisioned at the location where it was not marked as duplicate.

* Scenario B: The duplicate MAC or IP is un-provisioned at the location where it was marked as duplicate.

Unfreezing the duplicate and frozen MAC or IP will result in recovery to a steady state as follows:

* Scenario A: If the duplicate MAC or IP is un-provisioned at the non-duplicate location, unfreezing the route at the frozen location results in advertising with a higher sequence number, leading to automatic clearing of the local route at the un-provisioned location via ARP/ND PROBE and DELETE procedures.

* Scenario B: If the duplicate host is un-provisioned at the duplicate location, unfreezing the route triggers an advertisement with a higher sequence number to the other location, prompting re-learning and clearing of the local route at the original location upon receiving the remote route advertisement.

Probes referred to in these scenarios are event-driven probes resulting from receiving a route with a higher sequence number. Periodic probes resulting from refresh timers may also occur independently.

8.4.2. Route Clearing Configuration

In addition to the above, route clearing CLIs may be used to clear the local MAC or IP route after the duplicate host is un-provisioned:

* Clear MAC CLI: Used to clear a duplicate MAC route.

* Clear ARP/ND: Used to clear a duplicate IP route.

The route unfreeze CLI may still need to be executed if the route was un-provisioned and cleared from the non-duplicate location. Given that unfreezing the route via the CLI would result in auto-clearing from the un-provisioned location, as explained earlier, using a route clearing CLI for recovery from the duplicate state is optional.

9. Security Considerations

Security considerations discussed in [RFC7432] and [RFC9135] apply to this document. Methods described in this document further extend the consumption of sequence numbers for IRB deployments. They are hence subject to same considerations if the control plane or data plane was to be compromised. As an example, if host facing data plane is compromised, spoofing attempts could result in a legitimate host being perceived as moved, eventually resulting in the host being marked as duplicate. Considerations for protecting control and data plane described in [RFC7432] are equally applicable to such mobility spoofing use cases.

10. IANA Considerations

None.

11. Acknowledgements

Authors would like to thank Gunter van de Velde for significant contribution to improve the readability of this document. Authors would also like to thank Sonal Aggarwal and Larry Kreeger for multiple contributions through the implementation process. Authors would like to thank Vibov Bhan and Patrice Brissette for early feedback during implementation and testing of several procedures defined in this document. Authors would like to thank Wen Lin for a detailed review and valuable comments related to MAC sharing race conditions. Authors would also like to thank Saumya Dikshit for a detailed review and valuable comments across the document.

12. References

12.1. 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>.

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

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://datatracker.ietf.org/doc/html/rfc7432>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", RFC 8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC826] Plummer, D., "An Ethernet Address Resolution Protocol", RFC 826, November 1982, <https://www.rfc-editor.org/rfc/rfc826>.

[RFC9135] Sajassi, A., Salam, S., Thoria, S., Drake, J., and J. Rabadan, "Integrated Routing and Bridging in EVPN", RFC 9135, DOI 10.17487/RFC9135, October 2021, <https://www.rfc-editor.org/rfc/rfc9135>.

[RFC9161] Rabadan, J., Sathappan, S., Nagaraj, K., Hankins, G., and T. King, "Operational Aspects of Proxy-ARP/ND in EVPN Networks", RFC 9161, DOI 10.17487/RFC9161, January 2022, <https://www.rfc-editor.org/rfc/rfc9161>.

12.2. Informative References

[RFC7814] Xu, X., Jacquenet, C., Raszuk, R., Boyes, T., and B. Fee, "Virtual Subnet: A BGP/MPLS IP VPN-Based Subnet Extension Solution", RFC 7814, DOI 10.17487/RFC7814, March 2016, <https://tools.ietf.org/html/rfc7814>.

Authors' Addresses

Neeraj Malhotra (editor)
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
United States of America
Email: nmalhotr@cisco.com

Ali Sajassi
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
United States of America
Email: sajassi@cisco.com

Aparna Pattekar
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
United States of America
Email: apjoshi@cisco.com

Jorge Rabadan
Nokia
777 E. Middlefield Road
Mountain View, CA 94043
United States of America
Email: jorge.rabadan@nokia.com

Avinash Lingala
AT&T
3400 W Plano Pkwy
Plano, TX 75075
United States of America
Email: ar977m@att.com

John Drake
Juniper Networks
Email: jdrake@juniper.net

BESS Working Group
Internet-Draft
Intended status: Standards Track
Expires: 19 April 2025

P. Brissette
LA. Burdet, Ed.
Cisco Systems
B. Wen
Comcast
E. Leyton
Verizon Wireless
J. Rabadan
Nokia
16 October 2024

EVPN Port-Active Redundancy Mode
draft-ietf-bess-evpn-mh-pa-11

Abstract

The Multi-Chassis Link Aggregation Group (MC-LAG) technology enables establishing a logical link-aggregation connection with a redundant group of independent nodes. The objective of MC-LAG is to enhance both network availability and bandwidth utilization through various modes of traffic load-balancing. RFC7432 defines EVPN-based MC-LAG with Single-active and All-active multi-homing redundancy modes. This document builds on the existing redundancy mechanisms supported by EVPN and introduces a new Port-Active redundancy mode.

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 19 April 2025.

Copyright Notice

Table of Contents

1. Introduction

EVPN [RFC7432] defines the All-Active and Single-Active redundancy modes. All-Active redundancy provides per-flow load-balancing for multi-homing, while Single-Active redundancy ensures service carving where only one of the PEs in a redundancy relationship is active per service.

Although these two multi-homing scenarios are widely utilized in data center and service provider access networks, there are cases where active/standby multi-homing at the interface level is beneficial and necessary. The primary consideration for this new mode of load-

balancing is the determinism of traffic forwarding through a specific interface, rather than statistical per-flow load-balancing across multiple PEs providing multi-homing. This determinism is essential for certain QoS features to function correctly. Additionally, this mode ensures fast convergence during failure and recovery, which is expected by customers.

This document defines the Port-Active redundancy mode as a new type of multi-homing in EVPN and details how this mode operates and is supported via EVPN.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Multi-Chassis Link Aggregation (MC-LAG)

When a CE device is multi-homed to a set of PE nodes using the [IEEE_802.1AX_2014] Link Aggregation Control Protocol (LACP), the PEs must function as a single LACP entity for the Ethernet links to form and operate as a Link Aggregation Group (LAG). To achieve this, the PEs connected to the same multi-homed CE must synchronize LACP configuration and operational data among them. Historically, the Interchassis Communication Protocol (ICCP) [RFC7275] has been used for this synchronization. EVPN, as described in [RFC7432], covers the scenario where a CE is multi-homed to multiple PE nodes, using a LAG to simplify the procedure significantly. This simplification, however, comes with certain assumptions:

* a CE device connected to EVPN multi-homing PEs MUST have a single LAG with all its links connected to the EVPN multi-homing PEs in a redundancy group.

* identical LACP parameters MUST be configured on peering PEs, including system ID, port priority, and port key.

This document presumes proper LAG operation as specified in [RFC7432]. Issues resulting from deviations in the aforementioned assumptions, LAG misconfiguration, and miswiring detection across peering PEs are considered outside the scope of this document.

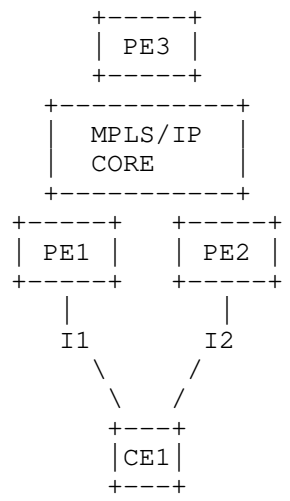


Figure 1: MC-LAG Topology

Figure 1 shows a MC-LAG multi-homing topology where PE1 and PE2 are part of the same redundancy group providing multi-homing to CE1 via interfaces I1 and I2. Interfaces I1 and I2 are members of a LAG running LACP. The core, shown as IP or MPLS enabled, provides a wide range of L2 and L3 services. MC-LAG multi-homing functionality is decoupled from those services in the core and it focuses on providing multi-homing to the CE. In Port-Active redundancy mode, only one of the two interfaces I1 or I2 would be in forwarding and the other interface will be in standby. This also implies that all services on the active interface are in active mode and all services on the standby interface operate in standby mode.

3. Port-Active Redundancy Mode

3.1. Overall Advantages

The use of Port-Active redundancy in EVPN networks provides the following benefits:

a. Port-Active redundancy offers open standards-based active/standby redundancy at the interface level, eliminating the need for ICCP and LDP (e.g., VXLAN or SRv6 may be used in the network).

b. This mode is agnostic of the underlying technology (MPLS, VXLAN, SRv6) and associated services (L2, L3, Bridging, E-LINE, etc.)

c. It enables deterministic QoS over MC-LAG attachment circuits.

d. Port-Active redundancy is fully compliant with [RFC7432] and does not require any new protocol enhancements to existing EVPN RFCs.

e. It can leverage various Designated Forwarder (DF) election algorithms, such as modulo ([RFC7432]), Highest Random Weight (HRW, [RFC8584]), etc.

f. Port-Active redundancy replaces legacy MC-LAG ICCP-based solutions and offers the following additional benefits:

   * Efficient support for 1+N redundancy mode (with EVPN using BGP RR), whereas ICCP requires a full mesh of LDP sessions among PEs in the redundancy group.

   * Fast convergence with mass-withdraw is possible with EVPN, which has no equivalent in ICCP.

3.2. Port-Active Redundancy Procedures

The following steps outline the proposed procedure for supporting Port-Active redundancy mode with EVPN LAG:

a. The Ethernet-Segment Identifier (ESI) MUST be assigned per access interface as described in [RFC7432]. The ESI can be auto-derived or manually assigned and the access interface MAY be a Layer-2 or Layer-3 interface.

b. The Ethernet-Segment (ES) MUST be configured in Port-Active redundancy mode on peering PEs for the specified access interface.

c. When ESI is configured on a Layer-3 interface, the Ethernet-Segment (ES) route (Route Type-4) MAY be the only route exchanged by PEs in the redundancy group.

d. PEs in the redundancy group leverage the DF election defined in [RFC8584] to determine which PE keeps the port in active mode and which one(s) keep it in standby mode. Although the DF election defined in [RFC8584] is per [ES, Ethernet Tag] granularity, the DF election is performed per [ES] in Port-Active redundancy mode. The details of this algorithm are described in Section 4.

e. The DF router MUST keep the corresponding access interface in an up and forwarding active state for that Ethernet-Segment.

f. Non-DF routers SHOULD implement a bidirectional blocking scheme for all traffic comparable to the Single-Active blocking scheme described in [RFC7432], albeit across all VLANs.

* Non-DF routers MAY bring and keep the peering access interface attached to them in an operational down state.

* If the interface is running the LACP protocol, the non-DF PE MAY set the LACP state to OOS (Out of Sync) instead of setting the interface to a down state. This approach allows for better convergence during the transition from standby to active mode.

g. The primary/backup bits of the EVPN Layer 2 Attributes Extended Community [RFC8214] SHOULD be used to achieve better convergence, as described in Section 5.1.

4. Designated Forwarder Algorithm to Elect per Port-Active PE

The ES routes operating in Port-Active redundancy mode are advertised with the new Port Mode Load-Balancing capability bit in the DF Election Extended Community as defined in [RFC8584]. Additionally, the ES associated with the port utilizes the existing Single-Active procedure and signals the Single-Active Multihomed site redundancy mode along with the Ethernet-AD per-ES route (refer to Section 7.5 of [RFC7432]). Finally, The ESI label-based split-horizon procedures specified in Section 8.3 of [RFC7432] SHOULD be employed to prevent transient echo packets when Layer-2 circuits are involved.

Various algorithms for DF Election are detailed in Sections 4.2 to 4.5 for comprehensive understanding, although the choice of algorithm in this solution does not significantly impact complexity or performance compared to other redundancy modes.

4.1. Capability Flag

[RFC8584] defines a DF Election extended community, and a Bitmap (2 octets) field to encode "capabilities" to use with the DF election algorithm in the DF algorithm field:

Bit 0: D bit or 'Don't Pre-empt' bit, as explained in [I-D.ietf-bess-evpn-pref-df].

Bit 1: AC-DF Capability (AC-Influenced DF election), as explained in [RFC8584].

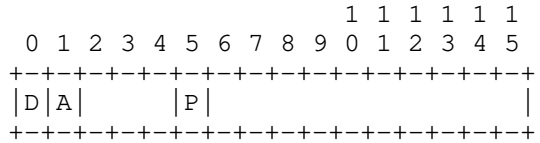


Figure 2: Amended Bitmap field in the DF Election Extended Community

This document defines the following value and extends the Bitmap field:

Bit 5: Port Mode Designated Forwarder Election (referred to as the P bit hereafter). This bit determines that the DF Election algorithm SHOULD be modified to consider the port ES only and not the Ethernet Tags.

4.2. Modulo-based Algorithm

The default DF Election algorithm, or modulo-based algorithm, as described in [RFC7432] and updated by [RFC8584], is applied here at the granularity of ES only. Given that the ES-Import Route Target extended community may be auto-derived and directly inherits its auto-derived value from ESI bytes 1-6, many operators differentiate ESIs primarily within these bytes. Consequently, bytes 3-6 are utilized to determine the designated forwarder using the modulo-based DF assignment, achieving good entropy during modulo calculation across ESIs.

Assuming a redundancy group of N PE nodes, the PE with ordinal i is designated as the DF for an <ES> when (Es mod N) = i, where Es represents bytes 3-6 of that ESI.

4.3. Highest Random Weight Algorithm

An application of Highest Random Weight (HRW) to EVPN DF Election is defined in [RFC8584] and MAY also be used and signaled. For Port-Active this is modified to operate at the granularity of <ES> rather than per <ES, VLAN>.

Section 3.2 of [RFC8584] describes computing a 32-bit CRC over the concatenation of Ethernet Tag (V) and ESI (Es). For Port-Active redundancy mode, the Ethernet Tag is omitted from the CRC computation and all references to (V, Es) are replaced by (Es).

The algorithm to detemine the DF Elected and Backup-DF Elected (BDF) at Section 3.2 of [RFC8584] is repeated and summarized below using only (Es) in the computation:

1. DF(Es) = Si| Weight(Es, Si) >= Weight(Es, Sj), for all j. In the case of a tie, choose the PE whose IP address is numerically the least. Note that 0 <= i,j < number of PEs in the redundancy group.

2. BDF(Es) = Sk| Weight(Es, Si) >= Weight(Es, Sk), and Weight(Es, Sk) >= Weight(Es, Sj). In the case of a tie, choose the PE whose IP address is numerically the least.

Where:

* DF(Es) is defined to be the address Si (index i) for which Weight(Es, Si) is the highest; 0 <= i < N-1.

* BDF(Es) is defined as that PE with address Sk for which the computed Weight is the next highest after the Weight of the DF. j is the running index from 0 to N-1; i and k are selected values.

4.4. Preference-based DF Election

When the new capability 'Port Mode' is signaled, the preference-based DF Election algorithm in [I-D.ietf-bess-evpn-pref-df] is modified to consider the port only and not any associated Ethernet Tags. The Port Mode capability is compatible with the 'Don't Pre-empt' bit and both may be signaled. When an interface recovers, a peering PE signaling D bit enables non-revertive behavior at the port level.

4.5. AC-Influenced DF Election

The AC-DF bit defined in [RFC8584] MUST be set to 0 when advertising Port Mode Designated Forwarder Election capability (P=1). When an AC (sub-interface) goes down, any resulting Ethernet A-D per EVI withdrawal does not influence the DF Election.

Upon receiving the AC-DF bit set (A=1) from a remote PE, it MUST be ignored when performing Port Mode DF Election.

5. Convergence considerations

To enhance convergence during failure and recovery when Port-Active redundancy mode is employed, advanced synchronization between peering PEs may be beneficial. The Port-Active mode poses a challenge since the "standby" port may be in a down state. Transitioning a "standby" port to an up state and stabilizing the network requires time. For Integrated Routing and Bridging (IRB) and Layer 3 services, synchronizing ARP / ND caches is recommended. Additionally, associated VRF tables may need to be synchronized. For Layer 2 services, synchronization of MAC tables may be considered.

Moreover, for members of a LAG running LACP, the ability to set the "standby" port to an "out-of-sync" state, also known as "warm-standby," can be utilized to improve convergence times.

5.1. Primary / Backup per Ethernet-Segment

The EVPN Layer 2 Attributes Extended Community ("L2-Attr") defined in [RFC8214] SHOULD be advertised in the Ethernet A-D per ES route to enable fast convergence.

Only the P and B bits of the Control Flags field in the L2-Attr Extended Community are relevant to this document, specifically in the context of Ethernet A-D per ES routes:

* When advertised, the L2-Attr Extended Community SHALL have only the P or B bits set in the Control Flags field, and all other bits and fields MUST be zero.

* A remote PE receiving the optional L2-Attr Extended Community in Ethernet A-D per ES routes SHALL consider only the P and B bits and ignore other values.

For L2-Attr Extended Community sent and received in Ethernet A-D per EVI routes used in [RFC8214], [RFC7432] and [I-D.ietf-bess-evpn-vpws-fxc]:

* P and B bits received SHOULD be considered overridden by "parent" bits when advertised in the Ethernet A-D per ES.

* Other fields and bits of the extended community are used according to the procedures outlined in the referenced documents.

By adhering to these procedures, the network ensures proper handling of the L2-Attr Extended Community to maintain robust and efficient convergence across Ethernet Segments.

5.2. Backward Compatibility

Implementations that comply with [RFC7432] or [RFC8214] only (i.e., implementations that predate this specification) will not advertise the EVPN Layer 2 Attributes Extended Community in Ethernet A-D per ES routes. That means that all remote PEs in the ES will not receive P and B bit per ES and will continue to receive and honour the P and B bits received in Ethernet A-D per EVI route(s). Similarly, an implementation that complies with [RFC7432] or [RFC8214] only and that receives an L2-Attr Extended Community in Ethernet A-D per ES routes will ignore it and continue to use the default path resolution algorithm:

* The remote ESI Label Extended Community ([RFC7432]) signals Single-Active (Section 4)

* the remote MAC and/or Ethernet A-D per EVI routes are unchanged, and since the L2-Attr Extended Community in Ethernet A-D per ES route is ignored, the P and B bits in the L2-Attr Extended Community in Ethernet A-D per EVI routes are used.

6. Applicability

A prevalent deployment scenario involves providing L2 or L3 services on PE devices that offer multi-homing capabilities. The services may include any L2 EVPN solutions such as EVPN VPWS or standard EVPN as defined in [RFC7432]. Additionally, L3 services may be provided within a VPN context, as specified in [RFC4364], or within a global routing context. When a PE provides first-hop routing, EVPN IRB may also be deployed on the PEs. The mechanism outlined in this document applies to PEs providing L2 and/or L3 services where active/standby redundancy at the interface level is required.

An alternative solution to the one described in this document is Multi-Chassis Link Aggregation Group (MC-LAG) with ICCP active-standby redundancy, as detailed in [RFC7275]. However, ICCP requires LDP to be enabled as a transport for ICCP messages. There are numerous scenarios where LDP is not necessary, such as deployments utilizing VXLAN or SRv6. The solution described in this document using EVPN does not mandate the use of LDP or ICCP and remains independent of the underlay encapsulation.

7. IANA Considerations

This document solicits the allocation of the following values from the "BGP Extended Communities" registry group :

* Bit 5 in the [RFC8584] DF Election Capabilities registry, "P bit - Port Mode Designated Forwarder Election".

8. Security Considerations

The Security Considerations described in [RFC7432] and [RFC8584] are applicable to this document.

Introducing a new capability necessitates unanimity among PEs. Without consensus on the new DF Election procedures and Port Mode, the DF Election algorithm defaults to the procedures outlined in [RFC8584] and [RFC7432].This fallback behavior could be exploited by an attacker who modifies the configuration of one PE within the Ethernet Segment (ES). Such manipulation could force all PEs in the ES to revert to the default DF Election algorithm and capabilities. In this scenario, the PEs may be subject to unfair load balancing, service disruption, and potential issues such as black-holing or duplicate traffic, as mentioned in the security sections of those documents.

9. Acknowledgements

The authors thank Anoop Ghanwani for his comments and suggestions and Stephane Litkowski for his careful review.

10. Contributors

In addition to the authors listed on the front page, the following coauthors have also contributed to this document:

Ali Sajassi
Cisco Systems
United States of America
Email: sajassi@cisco.com

Samir Thoria
Cisco Systems
United States of America
Email: sthoria@cisco.com

11. References

11.1. Normative References

[I-D.ietf-bess-evpn-pref-df]
Rabadan, J., Sathappan, S., Lin, W., Drake, J., and A. Sajassi, "Preference-based EVPN DF Election", Work in Progress, Internet-Draft, draft-ietf-bess-evpn-pref-df-13, 9 October 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-bess-evpn-pref-df-13>.

[IEEE_802.1AX_2014]
IEEE, "IEEE Standard for Local and metropolitan area networks -- Link Aggregation", IEEE 802-1ax-2014, DOI 10.1109/IEEESTD.2014.7055197, 5 March 2015, <https://ieeexplore.ieee.org/document/7055197>.

[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>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

[RFC8584] Rabadan, J., Ed., Mohanty, S., Ed., Sajassi, A., Drake, J., Nagaraj, K., and S. Sathappan, "Framework for Ethernet VPN Designated Forwarder Election Extensibility", RFC 8584, DOI 10.17487/RFC8584, April 2019, <https://www.rfc-editor.org/info/rfc8584>.

11.2. Informative References

[I-D.ietf-bess-evpn-vpws-fxc]
Sajassi, A., Brissette, P., Uttaro, J., Drake, J., Boutros, S., and J. Rabadan, "EVPN VPWS Flexible Cross-Connect Service", Work in Progress, Internet-Draft, draft-ietf-bess-evpn-vpws-fxc-08, 24 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-bess-evpn-vpws-fxc-08>.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC7275] Martini, L., Salam, S., Sajassi, A., Bocci, M., Matsushima, S., and T. Nadeau, "Inter-Chassis Communication Protocol for Layer 2 Virtual Private Network

(L2VPN) Provider Edge (PE) Redundancy", RFC 7275,
DOI 10.17487/RFC7275, June 2014,
<https://www.rfc-editor.org/info/rfc7275>.

Authors' Addresses

Patrice Brissette
Cisco Systems
Ottawa ON
Canada
Email: pbrisset@cisco.com

Luc Andre Burdet (editor)
Cisco Systems
Canada
Email: lburdet@cisco.com

Bin Wen
Comcast
United States of America
Email: Bin_Wen@comcast.com

Edward Leyton
Verizon Wireless
United States of America
Email: edward.leyton@verizonwireless.com

Jorge Rabadan
Nokia
United States of America
Email: jorge.rabadan@nokia.com

BESS WorkGroup
Internet-Draft
Intended status: Standards Track
Expires: 7 May 2025

A. Sajassi
P. Brissette
Cisco Systems
R. Schell
Verizon
J. Drake
Juniper
J. Rabadan
Nokia

3 November 2024

EVPN Virtual Ethernet Segment
draft-ietf-bess-evpn-virtual-eth-segment-16

Abstract

Ethernet VPN (EVPN) and Provider Backbone EVPN (PBB-EVPN) introduce a comprehensive suite of solutions for delivering Ethernet services over MPLS/IP networks. These solutions offer advanced features, including multi-homing capabilities. Specifically, they support Single-Active and All-Active redundancy modes for an Ethernet Segment (ES), which is defined as a collection of physical links connecting a multi-homed device or network to a set of Provider Edge (PE) devices. This document extends the concept of an Ethernet Segment by allowing an ES to be associated with a set of Ethernet Virtual Circuits (EVCs, such as VLANs) or other entities, including MPLS Label Switched Paths (LSPs) or Pseudowires (PWs). This extended concept is referred to as Virtual Ethernet Segments (vES). This draft outlines the requirements and necessary extensions to support vES in both EVPN and PBB-EVPN.

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] and [RFC8174].

Status of This Memo

This Internet-Draft will expire on 7 May 2025.

Table of Contents

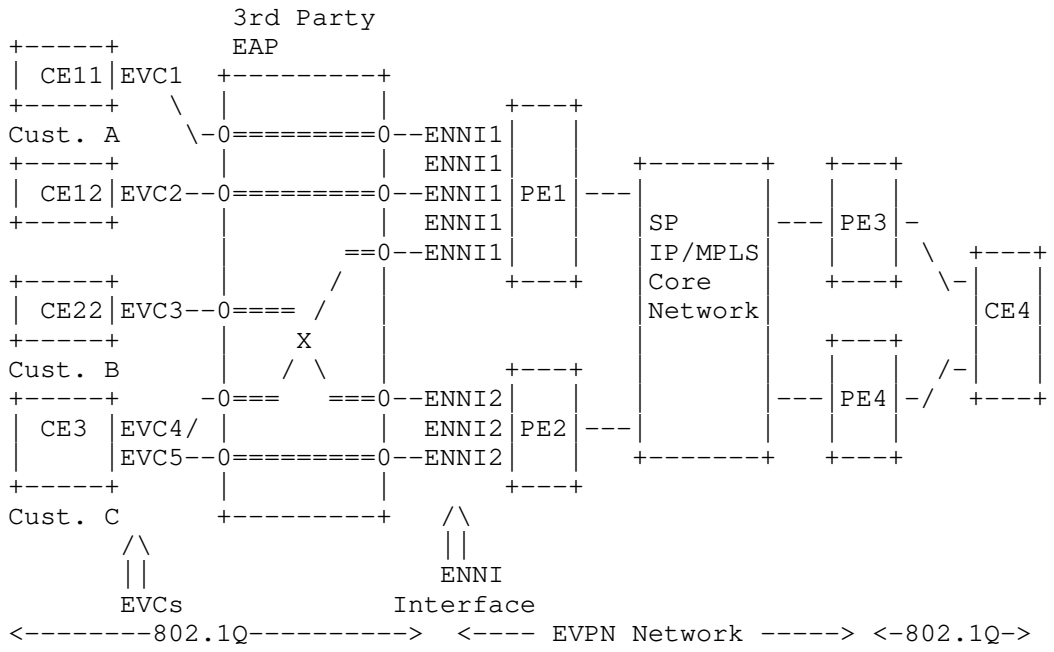
1. Introduction

Ethernet VPN (EVPN, [RFC7432]) and Provider Backbone EVPN (PBB-EVPN, [RFC7623])) introduce a comprehensive suite of solutions for delivering Ethernet services over MPLS/IP networks. These solutions offer advanced features, including multi-homing capabilities. Specifically, they support Single-Active and All-Active redundancy modes for an Ethernet Segment (ES). As defined in [RFC7432], an Ethernet Segment (ES) represents a collection of Ethernet links that connect a customer site to one ore more PEs devices.

This document extends the concept of an Ethernet Segment by allowing an ES to be associated with a set of Ethernet Virtual Circuits (EVCs, such as VLANs) or other entities, including MPLS Label Switched Paths (LSPs) or Pseudowires (PWs). This extended concept is referred to as Virtual Ethernet Segments (vES). This draft outlines the requirements and necessary extensions to support vES in both EVPN and PBB-EVPN. The scope of this document includes PBB-EVPN [RFC7623], EVPN over MPLS [RFC7432], and EVPN over IP [RFC8365]. However, it excludeds EVPN over SRv6 [RFC9252].

1.1. Virtual Ethernet Segments in Access Ethernet Networks

Some Service Providers (SPs) seek to extend the concept of physical Ethernet links in an ES to encompass Ethernet Virtual Circuits (EVCs), wherein multiple EVCs (such as VLANs) can be aggregated onto a single physical External Network-to-Network Interface (ENNI). An ES composed of a set of EVCs rather than physical links is referred to as a virtual ES (vES). Figure 1 illustrates two PE devices (PE1 and PE2), each with an ENNI aggregating several EVCs. Some of these EVCs on a given ENNI can be associated with vESes. For instance, the multi-homed vES depicted in Figure 1 consists of EVC4 on ENNI1 and EVC5 on ENNI2.



ENNIs are commonly used to reach remote customer sites via independent Ethernet access networks or third- party Ethernet Access Providers (EAP). ENNIs can aggregate traffic from many vESes (e.g., hundreds to thousands), where each vES is represented by its associated EVC on that ENNI. As a result, ENNIs and their associated EVCs are a key element of SP external boundaries that are carefully designed and closely monitored. As a reminder, the ENNI is the demarcation between the SP (IP/MPLS Core Network) and the third-party Ethernet Access Provider.

To meet customers' Service Level Agreements (SLA), SPs build redundancy via multiple EVPN PEs and across multiple ENNIs (as shown in Figure 1) where a given vES can be multi-homed to two or more EVPN PE devices (on two or more ENNIs) via their associated EVCs. Just like physical ESs in [RFC7432] and [RFC7623] solutions, these vESes can be single-homed or multi-homed ESs and when multi-homed, then can operate in either Single-Active or All-Active redundancy modes. In a typical SP external-boundary scenario (e.g., with an EAP), an ENNI can be associated with several thousands of single-homed vESes, several hundreds of Single- Active vESes and it may also be

associated with tens or hundreds of All-Active vESes. The specific figures (hundreds, thousands, etc.) used throughout this document reflect the relative quantities of various elements as understood at the time of writing.

1.2. Virtual Ethernet Segments in Access MPLS Networks

Other Service Providers (SPs) want to extend the concept of the physical links in an ES to individual Pseudowires (PWs) or to MPLS Label Switched Paths (LSPs) in Access MPLS networks - i.e., a vES consisting of a set of PWs or a set of LSPs. Figure 2 illustrates this concept.

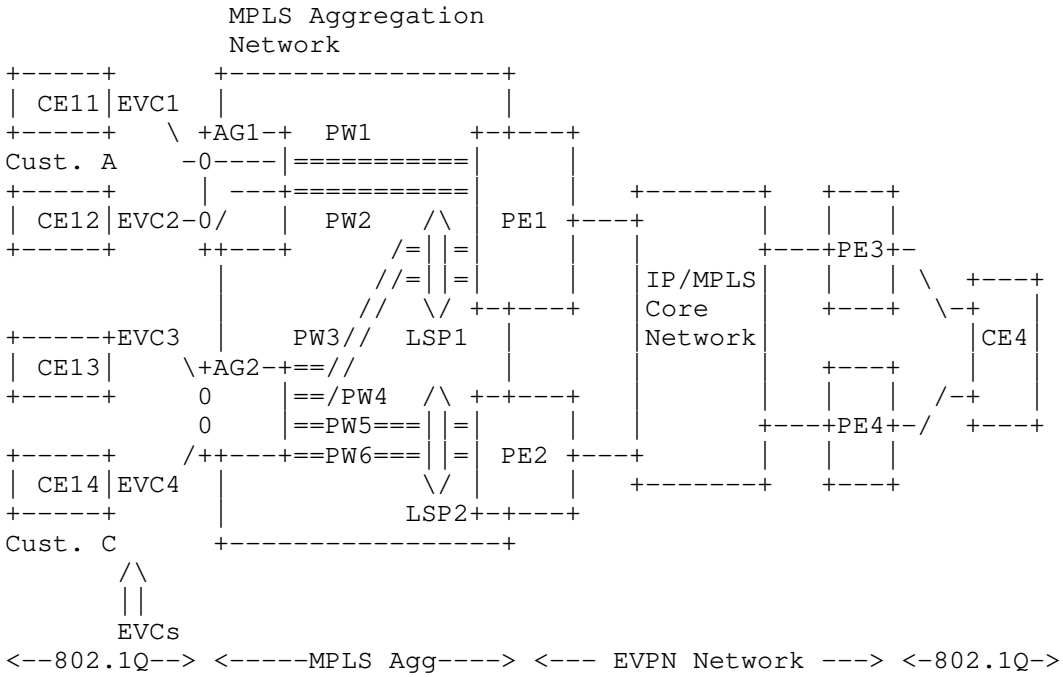


Figure 2: Dual-Homed and Single-homed Network on MPLS Aggregation networks

In certain scenarios, Service Providers utilize MPLS Aggregation Networks that are managed by separate administrative entities or third-party organizations to gain access to their own IP/MPLS core network infrastructure. This situation is depicted in Figure 2.

In such scenarios, a virtual ES (vES) is defined as a set of individual PWs when aggregation is not feasible. If aggregation is possible, the vES can be associated with a group of PWs that share the same unidirectional LSP pair, where the LSP pair consists of the ingress and egress LSPs between the same endpoints.

In the example of Figure 2, EVC3 is connected to a VPWS instance in AG2 that is connected to PE1 and PE2 via PW3 and PW5 respectively. EVC4 is connected to another VPWS instance on AG2 that is connected to PE1 and PE2 via PW4 and PW6, respectively. Since the PWs for the two VPWS instances can be aggregated into the same LSP pair going to and coming from the MPLS network, a common virtual ES (vES) can be defined for the four mentioned PWs. In Figure 2, LSP1 and LSP2 represent the two LSP pairs between PE1 and AG2, and between PE2 and AG2, respectively. The vES consists of these two LSP pairs (LSP1 and LSP2) and each LSP pair has two PWs. This vES will be shared by two separate EVPN instances (e.g., EVI-1 and EVI-2) in the EVPN network. PW3 and PW4 are associated with EVI-1 and EVI-2 respectively on PE1, and PW5 and PW6 are associated with EVI-1 and EVI-2 respectively on PE2.

In some cases, the aggregation of PWs that share the same LSP pair may not be possible. For instance, if PW3 were terminated into a third PE, e.g. PE3, instead of PE1, the vES would need to be defined on a per individual PW on each PE.

For MPLS/IP access networks where a virtual vES represents a set of LSP pairs or a set of PWs, this document extends the Single-Active multi-homing procedures defined in [RFC7432] and [RFC7623] to accommodate vES. The extension of vES to support All-Active multi-homing in MPLS/IP access networks is beyond the scope of this document.

This draft defines the concept of a vES and outlines the additional extensions necessary to support a vES in accordance with [RFC7432] and [RFC7623]. Section 3 enumerates the set of requirements for a vES. Section 4 details the extensions for a vES applicable to EVPN solutions, including those specified in [RFC7432] and [RFC7209]. These extensions are designed to meet the requirements outlined in Section 3. Section 4 also provides an overview of the solution, while Section 5 addresses failure handling, recovery, scalability, and fast convergence of [RFC7432] and [RFC7623] for vESes.

2. Terminology

AC: Attachment Circuit

B-MAC: Backbone MAC Address

CE: Customer Edge Device

C-MAC: Customer/Client MAC Address

DF: Designated Forwarder

ENNI: External Network-Network Interface

ES: Ethernet Segment

ESI: Ethernet Segment Identifier

Ethernet A-D: Ethernet Auto-Discovery Route

EVC: Ethernet Virtual Circuit, [MEF63]

EVI: EVPN Instance

EVPN: Ethernet VPN

I-SID: Service Instance Identifier (24 bits and global within a PBB network see [RFC7080])

PBB: Provider Backbone Bridge

PBB-EVPN: Provider Backbone Bridge EVPN

PE: Provider Edge Device

VPWS: Virtual Pseudowire Service

Single-Active Redundancy Mode (SA): When only a single PE, among a group of PEs attached to an Ethernet Segment, is allowed to forward traffic to/from that Ethernet Segment, then the Ethernet Segment is defined to be operating in Single-Active redundancy mode.

All-Active Redundancy Mode (AA): When all PEs attached to an Ethernet segment, are allowed to forward traffic to/from that Ethernet Segment, then the Ethernet Segment is defined to be operating in All-Active redundancy mode.

3. Requirements

This section describes the requirements specific to virtual Ethernet Segment (vES) for (PBB-)EVPN solutions. These requirements are in addition to the ones described in [RFC8214], [RFC7432], and [RFC7623].

3.1. Single-Homed and Multi-Homed vES

A PE device MUST support the following types of virtual Ethernet Segments (vES):

(R1a) The PE MUST handle single-homed vESes on a single physical port, such as a single ENNI.

(R1b) The PE MUST support a combination of single-homed vESes and Single-Active multi-homed vESes simultaneously on a single physical port, such as a single ENNI. Throughout this document, Single-Active multi-homed vESes will be referred to as Single-Active vESes.

(R1c) The PE MAY support All-Active multi-homed vESes on a single physical port. Throughout this document, All-Active multi-homed vESes will be referred to as All-Active vESes.

(R1d) The PE MAY support a combination of All-Active vESes along with other types of vESes on a single physical port.

(R1e) A Multi-Homed vES, whether Single-Active or All-Active, can span across two or more ENNIs on any two or more PEs.

3.2. Local Switching

Many vESes of different types can be aggregated on a single physical port on a PE device and some of these vESes can belong to the same service instance (e.g., EVI). This translates into the need for supporting local switching among the vESes for the same service instance on the same physical port (e.g., ENNI) of the PE.

(R3a) A PE device that supports the vES function MUST support local switching among different vESes associated with the same service instance on a single physical port. For instance, in Figure 1, PE1 must support local switching between CE11 and CE12, which are mapped to two single-homed vESes on ENNI1. In the case of Single-Active vESes, the local switching is performed among active EVCs associated with the same service instance on the same ENNI.

3.3. EVC Service Types

A physical port, such as an ENNI of a PE device, can aggregate numerous EVCs, each associated with a vES. An EVC may carry one or more VLANs. Typically, an EVC carries a single VLAN and is therefore associated with a single broadcast domain. However, there are no restrictions preventing an EVC from carrying multiple VLANs.

(R4a) An EVC can be associated with a single broadcast domain, such as in a VLAN-based service or a VLAN bundle service.

(R4b) An EVC MAY be associated with several broadcast domains, such as in a VLAN-aware bundle service.

Similarly, a PE can aggregate multiple LSPs and PWs. In the case of individual PWs per vES, typically, a PW is associated with a single broadcast domain, although there are no restrictions preventing a PW from carrying multiple VLANs if the PW is configured in Raw mode.

(R4c) A PW can be associated with a single broadcast domain, such as in a VLAN-based service or a VLAN bundle service.

(R4d) An PW MAY be associated with several broadcast domains, such as in a VLAN-aware bundle service.

3.4. Designated Forwarder (DF) Election

Section 8.5 of [RFC7432] outlines the default procedure for DF election in EVPN, which is also applied in [RFC7623] and [RFC8214]. [RFC8584] elaborates on additional procedures for DF election in EVPN. These DF election procedures are performed at the granularity of (ESI, Ethernet Tag). In the context of a vES, the same EVPN default procedure for DF election is applicable, but at the granularity of (vESI, Ethernet Tag). In this context, the Ethernet Tag is represented by an I-SID in PBB-EVPN and by a VLAN ID (VID) in EVPN. As described in [RFC7432], this default procedure for DF election at the granularity of (vESI, Ethernet Tag) is also known as "service carving." The goal of service carving is to evenly distribute the DFs for different vESes among various PEs, thereby ensuring an even distribution of traffic across the PEs. The following requirements are applicable to the DF election of vESes for (PBB-)EVPN.

(R5a) A PE that supports vES function, MUST support a vES with m EVCs among n ENNIs belonging to p PEs in any arbitrary order; where n >= p >= m >=2. For example, if there is a vES with 2 EVCs and there are 5 ENNIs on 5 PEs (PE1 through PE5), then vES can be dual homed to PE2 and PE4 and the DF election must be performed between PE2 and PE4.

(Rbc) Each vES MUST be identified by its own virtual ESI (vESI).

3.5. OAM

To detect the failure of an individual EVC and subsequently perform DF election for its associated vES as a result of this failure, each EVC should be monitored independently.

(R6a) Each EVC SHOULD be independently monitored for its operational health.

(R6b) A failure in a single EVC, among many aggregated on a single physical port or ENNI, MUST trigger a DF election for its associated vES.

3.6. Failure and Recovery

(R7a) Failure and failure recovery of an EVC for a Single-homed vES SHALL NOT impact any other EVCs within its service instance or any other service instances. In other words, for PBB-EVPN, it SHALL NOT trigger any MAC flushing both within its own I-SID as well as other I-SIDs.

(R7b) In case of All-Active vES, failure and failure recovery of an EVC for that vES SHALL NOT impact any other EVCs within its service instance or any other service instances. In other words, for PBB-EVPN, it SHALL NOT trigger any MAC flushing both within its own I-SID as well as other I-SIDs.

(R7c) Failure and failure recovery of an EVC for a Single-Active vES SHALL impact only its own service instance. In other words, for PBB-EVPN, MAC flushing SHALL be limited to the associated I-SID only and SHALL NOT impact any other I-SIDs.

(R7d) Failure and failure recovery of an EVC for a Single-Active vES MUST only impact C-MACs associated with multi-homed device/network for that service instance. In other words, MAC flushing MUST be limited to single service instance (I-SID in the case of PBB-EVPN) and only C-MACs for Single-Active multi-homed device/network.

3.7. Fast Convergence

Since many EVCs (and their associated vESes) are aggregated via a single physical port (e.g., ENNI), then the failure of that physical port impacts many vESes and triggers equally many ES route withdrawals. Formulating, sending, receiving, and processing such large number of BGP messages can introduce delay in DF election and convergence time. As such, it is highly desirable to have a mass-withdraw mechanism similar to the one in [RFC7432] for withdrawing many Ethernet A-D per ES routes.

(R8a) There SHOULD be a mechanism equivalent to EVPN mass-withdraw such that upon an ENNI failure, only a single BGP message is needed to indicate to the remote PEs to trigger DF election for all impacted vES associated with that ENNI.

4. Solution Overview

The solutions described in [RFC7432] and [RFC7623] are leveraged as-is with the modification that the ESI assignment is performed for an EVC or a group of EVCs or LSPs/PWs instead of a link or a group of physical links. In other words, the ESI is associated with a virtual ES (vES), hereby referred to as vESI.

In the EVPN solution, the overall procedures remain consistent, with the primary difference being the handling of physical port failures that can affect multiple vESes. Sections 5.1 and 5.3 describe the procedures for managing physical port or link failures in the context of EVPN. In a typical multi-homed setup, MAC addresses learned behind a vES are advertised using the ESI associated with the vES, referred to as the vESI. EVPN aliasing and mass-withdraw operations are conducted with respect to the vES identifier. Specifically, the Ethernet Auto-Discovery (A-D) routes for these operations are advertised using the vESI instead of the ESI.

For PBB-EVPN solution, the main change is with respect to the B-MAC address assignment which is performed similar to what is described in section 7.2.1.1 of [RFC7623] with the following refinements:

* One shared B-MAC address SHOULD be used per PE for the single-homed vESes. In other words, a single B-MAC is shared for all single-homed vESes on that PE.

* One shared B-MAC address SHOULD be used per PE per physical port (e.g., ENNI) for the Single-Active vESes. In other words, a single B-MAC is shared for all Single-Active vESes that share the same ENNI.

* One shared B-MAC address MAY be used for all Single-Active vESes on that PE.

* One B-MAC address SHOULD be used per set of EVCs representing an All-Active vES. In other words, a single B-MAC address is used per vES for All-Active scenarios.

* A single B-MAC address MAY also be used per vES per PE for Single-Active scenarios.

4.1. EVPN DF Election for vES

The procedure for service carving for virtual Ethernet Segments is almost the same as the ones outlined in section 8.5 of [RFC7432] and [RFC8584] except for the fact that ES is replaced with vES.

For the sake of clarity and completeness, the default DF election procedure of [RFC7432] is repeated below with the necessary changes:

1. When a PE discovers the vESI or is configured with the vESI associated with its attached vES, it advertises an Ethernet Segment route with the associated ES-Import extended community attribute.

2. The PE then starts a timer (default value = 3 seconds) to allow the reception of Ethernet Segment routes from other PE nodes connected to the same vES. This timer value MUST be same across all PEs connected to the same vES.

3. When the timer expires, each PE builds an ordered list of the IP addresses of all the PE nodes connected to the vES (including itself), in increasing numeric value. Each IP address in this list is extracted from the "Originator Router's IP address" field of the advertised Ethernet Segment route. Every PE is then given an ordinal indicating its position in the ordered list, starting with 0 as the ordinal for the PE with the numerically lowest IP address. The ordinals are used to determine which PE node will be the DF for a given EVPN instance on the vES using the following rule: Assuming a redundancy group of N PE nodes, the PE with ordinal i is the DF for an EVPN instance with an associated Ethernet Tag value of V when (V mod N) = i. It should be noted that using "Originator Router's IP address" field in the Ethernet Segment route to get the PE IP address needed for the ordered list, allows for a CE to be multi-homed across different ASes if such need ever arises.

4. The PE that is elected as a DF for a given EVPN instance will unblock traffic for that EVPN instance. Note that the DF PE unblocks all traffic in both ingress and egress directions for Single-Active vES and unblocks multi-destination in egress direction for All-Active Multi-homed vES. All non-DF PEs block all traffic in both ingress and egress directions for Single-Active vES and block multi-destination traffic in the egress direction for All-Active vES.

In case of an EVC failure, the affected PE withdraws its Virtual Ethernet Segment route if there are no more EVCs associated to the vES in the PE. This will re-trigger the DF Election procedure on all the PEs in the Redundancy Group. For PE node failure, or upon PE commissioning or decommissioning, the PEs re-trigger the DF Election procedure across all affected vESes. In case of a Single-Active, when a service moves from one PE in the Redundancy Group to another PE because of DF re-election, the PE, which ends up being the elected DF for the service, MUST trigger a MAC address flush notification towards the associated vES if the multi-homing device is a bridge or the multi-homing network is an Ethernet bridged network.

For LSP-based and PW-based vES, the non-DF PE SHOULD signal PW-status 'standby' to the Aggregation PE (e.g., AG1 and AG2 in Figure 2), and a new DF PE MAY send an LDP MAC withdraw message as a MAC address flush notification. It should be noted that the PW-status is signaled for the scenarios where there is a one-to-one mapping between EVI (EVPN instance) and the PW.

4.2. Grouping and Route Coloring for vES

Physical ports (e.g. ENNI) which aggregate many EVCs are 'colored' to enable the grouping schemes described below.

By default, the MAC address of the corresponding port (e.g. ENNI) is used to represent the 'color' of the port, and the EVPN Router's MAC Extended Community defined in [RFC9135] is used to signal this color.

The difference between coloring mechanism for EVPN and PBB-EVPN is that for EVPN, the extended community is advertised with the Ethernet A-D per ES route whereas for PBB-EVPN, the extended community is advertised with the B-MAC route.

The subsequent sections detailing Grouping of Ethernet Auto-Discovery (A-D) per ES and Grouping of B-MAC addresses will be essential for addressing port failure handling, as discussed in Sections Section 5.3, Section 5.4, and Section 5.5.

4.2.1. EVPN Route Coloring for vES

When a PE discovers the vESI or is configured with the vESI associated with its attached vES, an Ethernet-Segment route and Ethernet A-D per ES route are generated using the vESI identifier.

These Ethernet-Segment and Ethernet A-D per ES routes specific to each vES are colored with an attribute representing their association to a physical port (e.g. ENNI).

The corresponding port 'color' is encoded in the EVPN Router's MAC Extended Community defined in [RFC9135] and advertised along with the Ethernet Segment and Ethernet A-D per ES routes for this vES. The color (which is the MAC address of the port) MUST be unique.

The PE also constructs a special Grouping Ethernet A-D per ES route which represents all the vES associated with the port (e.g. ENNI). The corresponding port 'color' is encoded in the ESI field. For this encoding, Type 3 ESI (Section 5 of [RFC7432]) is used with the MAC field set to the color (MAC address) of the port and the 3-octet local discriminator field set to 0xFFFFFF.

The ESI label extended community (Section 7.5 of [RFC7432]) is not relevant to Grouping Ethernet A-D per ES route. The label value is not used for encapsulating BUM (Broadcast, Unknown-unicast, Multicast) packets for any split-horizon function. The ESI label extended community MUST NOT be added to Grouping Ethernet A-D per ES route and MUST be ignored on receiving PE.

The Grouping Ethernet Auto-Discovery (A-D) per ES route is advertised with a list of Route Targets corresponding to the affected service instances. If the number of associated Route Targets exceeds the capacity of a single route, multiple Grouping Ethernet A-D per ES routes are advertised accordingly.

4.2.2. PBB-EVPN Route Coloring for vES

In PBB-EVPN, particularly when there are a large number of service instances (i.e., I-SIDs) associated with each EVC, the PE device MAY assign a color attribute to each vES B-MAC route, indicating their association with a physical port (e.g., an ENNI).

The corresponding port 'color' is encoded in the EVPN Router's MAC Extended Community defined in [RFC9135] and advertised along with the B-MAC for this vES in PBB-EVPN.

The PE MAY then also construct a special Grouping B-MAC route which represents all the vES associated with the port (e.g. ENNI). The corresponding port 'color' is encoded directly into this special Grouping B-MAC route.

5. Failure Handling and Recovery

There are several failure scenarios to consider such as:

A: CE uplink port failure

B: Ethernet Access Network failure

C: PE access-facing port or link failure

D: PE node failure

E: PE isolation from IP/MPLS network

The solutions outlined in [RFC7432], [RFC7623], and [RFC8214] provide protection against failures as described in these respective references. In the context of these solutions, the presence of vESes introduces an additional failure scenario beyond those already considered, specifically the failure of individual EVCs. Addressing vES failure scenarios necessitates the independent monitoring of EVCs or PWs. Upon detection of failure or service restoration, appropriate DF election and failure recovery mechanisms must be executed.

[RFC7023] is used for monitoring EVCs and upon failure detection of a given EVC, DF election procedure per Section 4.1 is executed. For PBB-EVPN, some extensions are needed to handle the failure and recovery procedures of [RFC7623] to meet the above requirements. These extensions are described in the next section.

[RFC4377] and [RFC6310] are used for monitoring the status of LSPs and/or PWs associated to vES.

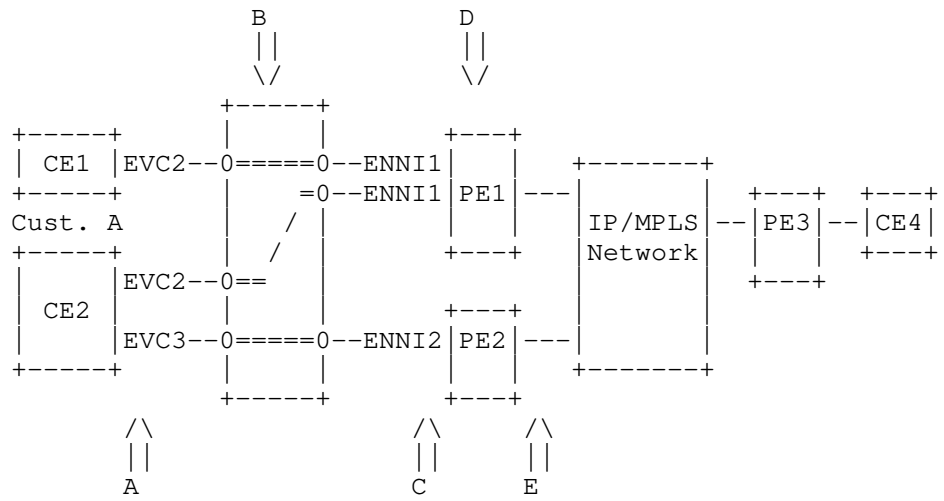


Figure 4: Failure Scenarios A,B,C,D and E

5.1. EVC Failure Handling for Single-Active vES in EVPN

In [RFC7432], when a DF PE connected to a Single-Active multi-homed Ethernet Segment loses connectivity to the segment, due to link or port failure, it signals to the remote PEs to invalidate all MAC addresses associated with that Ethernet Segment. This is done by means of a mass-withdraw message, by withdrawing the Ethernet A-D per ES route. It should be noted that for dual-homing use cases where there is only a single backup path, MAC invalidating can be avoided by the remote PEs as they can update their next hop associated with the affected MAC entries to the backup path per procedure described in section 8.2 of [RFC7432].

In case of an EVC failure which impacts a single vES, this same EVPN procedure is used. In this case, the mass-withdraw is conveyed by withdrawing the Ethernet A-D per vES route carrying the vESI representing the failed EVC. The remote PEs upon receiving this message perform the same procedures outlined in section 8.2 of [RFC7432].

5.2. EVC Failure Handling for Single-Active vES in PBB-EVPN

In [RFC7432] when a PE connected to a Single-Active Ethernet Segment loses connectivity to the segment, due to link or port failure, it signals the remote PE to flush all C-MAC addresses associated with that Ethernet Segment. This is done by updating the advertised a B-MAC route's MAC Mobility Extended community.

In case of an EVC failure that impacts a single vES, if the above PBB-EVPN procedure is used, it results in excessive C-MAC flushing because a single physical port can support large number of EVCs (and their associated vESes) and thus updating the advertised B-MAC corresponding to the physical port, with MAC mobility Extended community, will result in flushing C-MAC addresses not just for the impacted EVC but for all other EVCs on that port.

To reduce the scope of C-MAC flushing to only the impacted service instances (the service instance(s) impacted by the EVC failure), the PBB-EVPN C-MAC flushing needs to be adapted on a per service instance basis (i.e., per I-SID). [RFC9541] introduces B-MAC/I-SID route where existing PBB-EVPN B-MAC route is modified to carry an I-SID in the "Ethernet Tag ID" field instead of NULL value. This field indicates to the receiving PE, to flush all C-MAC addresses associated with that I-SID for that B-MAC. This C-MAC flushing mechanism per I-SID SHOULD be used in case of EVC failure impacting a vES. Since typically an EVC maps to a single broadcast domain and thus, a single service instance, the affected PE only needs to advertise a single B-MAC/I-SID route. However, if the failed EVC carries multiple VLANs each with its own broadcast domain, then the affected PE needs to advertise multiple B-MAC/I-SID routes - one for each VLAN (broadcast domain) - i.e., one for each I-SID. Each B-MAC/I-SID route basically instructs the remote PEs to perform flushing for C-MACs corresponding to the advertised B-MAC only for the advertised I-SID.

The C-MAC flushing based on B-MAC/I-SID route works fine when there are only a few VLANs (e.g., I-SIDs) per EVC. However if the number of I-SIDs associated with a failed EVC is large, then it is RECOMMENDED to assign a B-MAC per vES and upon EVC failure, the affected PE simply withdraws this B-MAC message to other PEs.

5.3. Port Failure Handling for Single-Active vESes in EVPN

When many EVCs are aggregated via a single physical port on a PE, where each EVC corresponds to a vES, then the port failure impacts all the associated EVCs and their corresponding vESes. If the number of EVCs corresponding to the Single-Active vESes for that physical port is in thousands, then thousands of service instances are impacted. Therefore, the propagation of failure in BGP needs to address all these impacted service instances. In order to achieve this, the following extensions are added to the baseline EVPN mechanism:

1. The PE MAY color each Ethernet A-D per ES route for a given vES, as described in Section 4.2.1. PE SHOULD use the physical port MAC by default. The receiving PEs take note of this color and create a list of vESes for this color.

2. The PE MAY advertises a special Grouping Ethernet A-D per ES route for that color, which represents all the vES associated with the port.

3. Upon a port failure (e.g., ENNI failure), the PE MAY send a mass-withdraw message by withdrawing the Grouping Ethernet A-D per ES route.

4. When this message is received, the remote PE MAY detect the special vES mass-withdraw message by identifying the Grouping Ethernet A-D per ES route. The remote PEs MAY then access the list created in (1) of the vESes for the specified color, and initiate locally MAC address invalidating procedures for each of the vESes in the list.

In scenarios where a logical ENNI is used the above procedure equally applies. The logical ENNI is represented by a Grouping Ethernet A-D per ES where the Type 3 ESI and the 6 bytes used in the ENNI's ESI MAC address field is used as a color for vESes as described above and in Section 4.2.1.

5.4. Port Failure Handling for Single-Active vESes in PBB-EVPN

When many EVCs are aggregated via a single physical port on a PE, where each EVC corresponds to a vES, then the port failure impacts all the associated EVCs and their corresponding vESes. If the number of EVCs corresponding to the Single-Active vESes for that physical port is in thousands, then thousands of service instances (I-SIDs) are impacted. In such failure scenarios, the following two MAC flushing mechanisms per [RFC7623] can be performed.

1. If the MAC address of the physical port is used for PBB encapsulation as B-MAC SA, then upon the port failure, the PE MUST use the EVPN MAC route withdrawal message to signal the flush.

2. If the PE shared MAC address is used for PBB encapsulation as B-MAC SA, then upon the port failure, the PE MUST re-advertise this MAC route with the MAC Mobility Extended Community to signal the flush.

The first method is recommended because it reduces the scope of flushing the most.

As noted above, the advertisement of the extended community along with B-MAC route for coloring purposes is optional and only recommended when there are many vESes per physical port and each vES is associated with very large number of service instances (i.e., large number of I-SIDs).

If there are large number of service instances (i.e., I-SIDs) associated with each EVC, and if there is a B-MAC assigned per vES as recommended in the above section, then to handle port failure efficiently, the following extensions are added to the baseline PBB-EVPN mechanism:

1. Each vES MAY be colored with a MAC address representing the physical port like the coloring mechanism for EVPN. In other words, each B-MAC representing a vES is advertised with the 'color' of the physical port per Section 4.2.2. The receiving PEs take note of this color being advertised along with the B-MAC route and for each such color, create a list of vESes associated with this color.

2. The PE MAY advertise a special Grouping B-MAC route for that color (consisting by default of port MAC address), which represents all the vES associated with the port.

3. Upon a port failure (e.g., ENNI failure), the PE MAY send a mass-withdraw message by withdrawing the Grouping B-MAC route.

4. When this message is received, the remote PE MAY detect the special vES mass-withdraw message by identifying the Grouping B-MAC route. The remote PEs MAY then access the list created in (1) for the specified color, and flush all C-MACs associated with the failed physical port.

5.5. Fast Convergence in (PBB-)EVPN

As described above, when many EVCs are aggregated via a physical port on a PE, and where each EVC corresponds to a vES, then the port failure impacts all the associated EVCs and their corresponding vESes. Two actions must be taken as the result of such port failure:

* For EVPN initiate mass-withdraw procedure for all vESes associated with the failed port to invalidate MACs and for PBB-EVPN flush all C-MACs associated with the failed port across all vESes and the impacted I-SIDs

* DF election for all impacted vESes associated with the failed port

Section 5.3 already describes how to perform mass-withdraw for all affected vESes and invalidating MACs using a single BGP withdrawal of the Grouping Ethernet A-D per ES route. Section 5.4 describes how to only flush C-MAC address associated with the failed physical port (e.g., optimum C-MAC flushing) as well as, optionally, the withdrawal of a Grouping B-MAC route.

This section describes how to perform DF election in the most optimal way - e.g., to trigger DF election for all impacted vESes (which can be very large) among the participating PEs via a single BGP message as opposed to sending large number of BGP messages (one per vES). This section assumes that the MAC flushing mechanism described in Section 5.4 is used and route coloring is used.

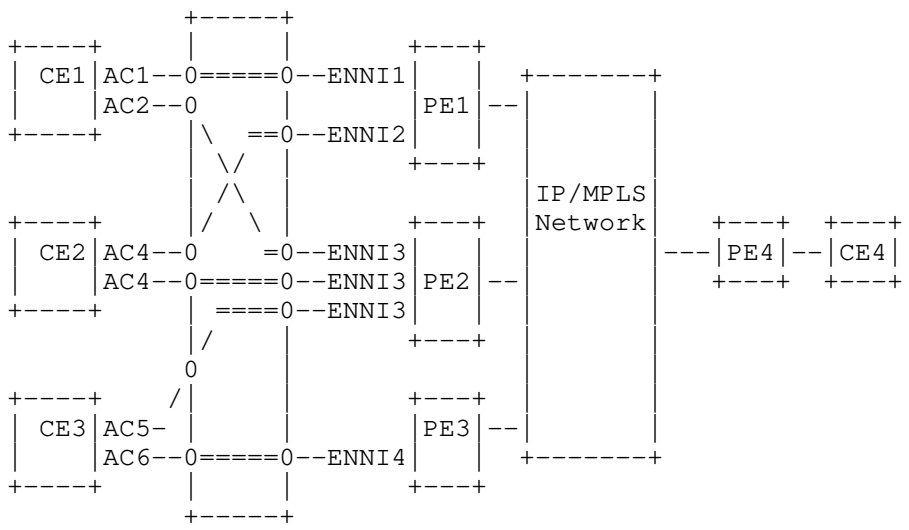


Figure 5: Fast Convergence Upon ENNI Failure

The procedure for coloring vES Ethernet Segment routes is described in Section 4.2. The following describes the procedure for fast convergence for DF election using these colored routes:

1. When a vES is configured, the PE SHOULD advertise the Ethernet Segment route for this vES with a color that corresponds to the associated physical port.

2. All receiving PEs within the redundancy group SHOULD record this color and compile a list of vESes associated with it.

3. Additionally, the PE SHOULD advertise a Grouping Ethernet A-D per ES for EVPN, and a Grouping B-MAC for PBB-EVPN, which corresponds to the color and vES grouping.

4. In the event of a port failure, such as an ENNI failure, the PE SHOULD withdraw the previously advertised Grouping Ethernet A-D per ES or Grouping B-MAC associated with the failed port. The PE SHOULD prioritize sending these Grouping route withdrawal messages over the withdrawal of individual vES routes affected by the failure. For instance, as depicted in Figure 5, when the physical port associated with ENNI3 fails on PE2, it withdraws the previously advertised Grouping Ethernet A-D per ES route. Upon receiving this withdrawal message, other multi-homing PEs

(such as PE1 and PE3) recognize that the vESes associated with CE1 and CE3 are impacted, based on the associated color, and thus initiate the DF election procedure for these vESes. Furthermore, remote PEs (such as PE4), upon receiving this withdrawal message, initiate the failover procedure for the vESes associated with CE1 and CE3, and switch to the other PE for each vES redundancy group.

5. On reception of Grouping Ethernet A-D per ES or Grouping B-MAC route withdrawal, other PEs in the redundancy group SHOULD initiate DF election procedures across all their affected vESes.

6. The PE with the physical port failure (ENNI failure), SHOULD send vES route withdrawal for every impacted vES. The other PEs upon receiving these messages, clear up their BGP tables. It should be noted the vES route withdrawal messages are not used for executing DF election procedures by the receiving PEs when Grouping Ethernet A-D per ES or Grouping B-MAC withdrawal has been previously received.

6. Acknowledgements

The authors would like to thank Mei Zhang, Jose Liste, and Luc Andre Burdet for their reviews of this document and feedback.

7. Security Considerations

All the security considerations in [RFC7432] and [RFC7623] apply directly to this document because this document leverages the control and data plane procedures described in those documents.

This document does not introduce any new security considerations beyond that of [RFC7432] and [RFC7623] because advertisements and processing of Ethernet Segment route for vES in this document follows that of physical ES in those RFCs.

8. IANA Considerations

This document requests no actions from IANA.

9. References

9.1. 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>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <https://www.rfc-editor.org/info/rfc7623>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

[RFC9135] Sajassi, A., Salam, S., Thoria, S., Drake, J., and J. Rabadan, "Integrated Routing and Bridging in Ethernet VPN (EVPN)", RFC 9135, DOI 10.17487/RFC9135, October 2021, <https://www.rfc-editor.org/info/rfc9135>.

[RFC9541] Rabadan, J., Ed., Sathappan, S., Nagaraj, K., Miyake, M., and T. Matsuda, "Flush Mechanism for Customer MAC Addresses Based on Service Instance Identifier (I-SID) in Provider Backbone Bridging EVPN (PBB-EVPN)", RFC 9541, DOI 10.17487/RFC9541, March 2024, <https://www.rfc-editor.org/info/rfc9541>.

9.2. Informative References

[MEF63] Metro Ethernet Forum, MEF., "[MEF6.3]: Subscriber Ethernet Services Definitions", 2019.

[RFC4377] Nadeau, T., Morrow, M., Swallow, G., Allan, D., and S. Matsushima, "Operations and Management (OAM) Requirements for Multi-Protocol Label Switched (MPLS) Networks", RFC 4377, DOI 10.17487/RFC4377, February 2006, <https://www.rfc-editor.org/info/rfc4377>.

[RFC6310] Aissaoui, M., Busschbach, P., Martini, L., Morrow, M., Nadeau, T., and Y. Stein, "Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping", RFC 6310, DOI 10.17487/RFC6310, July 2011, <https://www.rfc-editor.org/info/rfc6310>.

[RFC7023] Mohan, D., Ed., Bitar, N., Ed., Sajassi, A., Ed., DeLord, S., Niger, P., and R. Qiu, "MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking", RFC 7023, DOI 10.17487/RFC7023, October 2013, <https://www.rfc-editor.org/info/rfc7023>.

[RFC7080] Sajassi, A., Salam, S., Bitar, N., and F. Balus, "Virtual Private LAN Service (VPLS) Interoperability with Provider Backbone Bridges", RFC 7080, DOI 10.17487/RFC7080, December 2013, <https://www.rfc-editor.org/info/rfc7080>.

[RFC7209] Sajassi, A., Aggarwal, R., Uttaro, J., Bitar, N., Henderickx, W., and A. Isaac, "Requirements for Ethernet VPN (EVPN)", RFC 7209, DOI 10.17487/RFC7209, May 2014, <https://www.rfc-editor.org/info/rfc7209>.

[RFC8584] Rabadan, J., Ed., Mohanty, S., Ed., Sajassi, A., Drake, J., Nagaraj, K., and S. Sathappan, "Framework for Ethernet VPN Designated Forwarder Election Extensibility", RFC 8584, DOI 10.17487/RFC8584, April 2019, <https://www.rfc-editor.org/info/rfc8584>.

[RFC9252] Dawra, G., Ed., Talaulikar, K., Ed., Raszuk, R., Decraene, B., Zhuang, S., and J. Rabadan, "BGP Overlay Services Based on Segment Routing over IPv6 (SRv6)", RFC 9252, DOI 10.17487/RFC9252, July 2022, <https://www.rfc-editor.org/info/rfc9252>.

Authors' Addresses

Ali Sajassi
Cisco Systems
Email: sajassi@cisco.com

Patrice Brissette
Cisco Systems
Email: pbrisset@cisco.com

Rick Schell
Verizon

Email: richard.schell@verizon.com

John E Drake
Juniper
Email: jdrake@juniper.net

Jorge Rabadan
Nokia
Email: jorge.rabadan@nokia.com

BESS Working Group
Internet-Draft
Intended status: Standards Track
Expires: 23 March 2025

A. Sajassi, Ed.
P. Brissette
Cisco Systems
J. Uttaro
AT&T
J. Drake
Juniper Networks
S. Boutros
Ciena
J. Rabadan
Nokia

19 September 2024

EVPN VPWS Flexible Cross-Connect Service
draft-ietf-bess-evpn-vpws-fxc-09

Abstract

This document describes a new EVPN VPWS service type specifically for multiplexing multiple attachment circuits across different Ethernet Segments and physical interfaces into a single EVPN VPWS service tunnel and still providing Single-Active and All-Active multi-homing. This new service is referred to as flexible cross-connect service. After a description of the rationale for this new service type, the solution to deliver such service is detailed.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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/.

Copyright Notice

Table of Contents

1. Introduction

[RFC8214] describes a solution to deliver P2P services using BGP constructs defined in [RFC7432]. It delivers this P2P service between a pair of Attachment Circuits (ACs), where an AC can designate on a PE, a port, a VLAN on a port, or a group of VLANs on a port. It also leverages multi-homing and fast convergence capabilities of [RFC7432] in delivering these VPWS services. Multi-homing capabilities include the support of single-active and all-active redundancy mode and fast convergence is provided using "mass withdrawal" message in control-plane and fast protection switching using prefix independent convergence in data-plane upon node or link failure [I-D.ietf-rtgwg-bgp-pic]. Furthermore, the use of EVPN BGP constructs eliminates the need for multi-segment PW auto-discovery and signaling if the VPWS service need to span across multiple ASes [RFC5659].

Some service providers have very large number of ACs (in millions) that need to be back hauled across their MPLS/IP network. These ACs may or may not require tag manipulation (e.g., VLAN translation). These service providers want to multiplex a large number of ACs across several physical interfaces spread across one or more PEs (e.g., several Ethernet Segments) onto a single VPWS service tunnel in order to a) reduce number of EVPN service labels associated with EVPN-VPWS service tunnels and thus the associated OAM monitoring, and b) reduce EVPN BGP signaling (e.g., not to signal each AC as it is the case in [RFC8214]).

These service provider want the above functionality without scarifying any of the capabilities of [RFC8214] including single-active and all-active multi-homing, and fast convergence.

This document presents a solution based on extensions to [RFC8214] to meet the above requirements.

1.1. Terminology

AC: Attachment Circuit

CE: Customer Edge device e.g., host or router or switch

EPL: Ethernet Private Line

ES: Ethernet Segment

ESI: Ethernet Segment Identififer

EVI: EVPN Instance Identifier

EVPL: Ethernet Virtual Private Line

EVPN: Ethernet Virtual Private Network

Ethernet A-D: Ethernet Auto-Discovery (A-D) per EVI and Ethernet A-D per ESI routes, as defined in [RFC7432] and [RFC8214].

FXC: Flexible Cross Connect

L2: Layer 2

MAC: Media Access Control

MPLS: Multi Protocol Label Switching

MTU: Maximum Transmit Unit

OAM: Operations, Administration and Maintenance

PE: Provider Edge device

PW: Pseudowire

RT: Route Target

VCCV: Virtual circuit connection verification

VID: Vlan ID

VPWS: Virtual private wire service

VRF: Virtual Route Forwarding

VPWS Service Tunnel: It is represented by a pair of EVPN service labels associated with a pair of endpoints. Each label is downstream assigned and advertised by the disposition PE through an Ethernet Auto-Discovery (A-D) per EVI route. The downstream label identifies the endpoint on the disposition PE. A VPWS service tunnel can be associated with many VPWS service identifiers where each identifier is a normalized VID.

Single-Active Redundancy Mode: When a device or a network is multi-homed to two or more PEs and when only a single PE in such redundancy group can forward traffic to/from the multi-homed device or network for a given VLAN, then such multi-homing or redundancy is referred to as "Single-Active".

All-Active Redundancy Mode: When a device is multi-homed to two or

more PEs and when all PEs in such redundancy group can forward traffic to/from the multi-homed device for a given VLAN, then such multi-homing or redundancy is referred to as "All-Active".

2. Requirements

Two of the main motivations for service providers seeking a new solution are: 1) to reduce number of VPWS service tunnels by multiplexing large number of ACs across different physical interfaces instead of having one VPWS service tunnel per AC, and 2) to reduce the signaling of ACs as much as possible. Besides these two requirements, they also want multi-homing and fast convergence capabilities of [RFC8214].

In [RFC8214], a PE signals an AC indirectly by first associating that AC to a VPWS service tunnel (e.g., a VPWS service instance) and then signaling the VPWS service tunnel via a Ethernet A-D per EVI route with Ethernet Tag field set to a 24-bit VPWS service instance identifier (which is unique within the EVI) and ESI field set to a 10-octet identifier of the Ethernet Segment corresponding to that AC.

Therefore, a PE device that receives such EVPN routes, can associate the VPWS service tunnel to the remote Ethernet Segment using the ESI field, and when the remote ES fails and the PE receives the "mass withdrawal" message associated with the failed ES per [RFC7432], it can quickly update its BGP list of available remote entries to invalidate all VPWS service tunnels sharing the ESI field and achieve fast convergence for multi-homing scenarios. Even if fast convergence were not needed, there would still be a need for signaling each AC failure (via its corresponding VPWS service tunnel) associated with the failed ES, so that the BGP path list for each of them gets updated accordingly and the packets are sent to backup PE (in case of single- active multi-homing) or to other PEs in the redundancy group (in case of all-active multi-homing). In absence of updating the BGP path list, the traffic for that VPWS service tunnel will be black-holed.

When a single VPWS service tunnel carries multiple ACs across various Ethernet Segments (physical interfaces) without signaling the ACs via EVPN BGP to remote PE devices, those remote PE devices lack the information to associate the received Ethernet Segment with these ACs or with their local ACs. They also lack the association between the VPWS service tunnel (e.g., EVPN service label) and the far-end ACs. This means that while the remote PEs can associate their local ACs with the VPWS service tunnel, they cannot make similar associations for the far-end ACs.
Consequently, in case of a connectivity failure to the ES, the remote PEs are unable to redirect traffic via another multi-homing PE to

that ES. In other words, even if an ES failure is signaled via EVPN to the remote PE devices, they cannot effectively respond because they do not know the relationship between the remote ES, the remote ACs, and the VPWS service tunnel.

To address this issue when multiplexing a large number of ACs onto a single VPWS service tunnel, two mechanisms have been developed: one to support VPWS services between two single-homed endpoints, and another to support VPWS services where one of the endpoints is multi-homed.

For single-homed endpoints, it is acceptable not to signal each AC in BGP because, in the event of a connection failure to the ES, there is no alternative path to that endpoint. However, the implication of not signaling an AC failure is that the traffic destined for the failed AC is sent over the MPLS/IP core and then discarded at the destination PE, thereby potentially wasting network resources. This waste of network resources during a connection failure may be transient, as it can be detected and prevented at the application layer in certain cases. Section 3.2 outlines a solution for such single-homing VPWS services.

For VPWS services where one of the endpoints is multi-homed, there are two options:

1) to signal each AC via BGP, allowing the path list to be updated upon a failure affecting those ACs. This solution is described in Section 3.3 and is referred to as the VLAN-signaled flexible cross-connect service.

2) to bundle several ACs on an ES together per destination endpoint (e.g., ES, MAC-VRF, etc.) and associate such a bundle with a single VPWS service tunnel. This approach is similar to the VLAN-bundle service interface described in [RFC8214]. This solution is described in Section 3.2.1.

3. Solution

This section outlines a solution for providing a new VPWS service between two PE devices where a large number of ACs (such as VLANs) that span across multiple Ethernet Segments (physical interfaces) on each PE are multiplexed onto a single P2P EVPN service tunnel. Since the multiplexing involves several physical interfaces, there can be overlapping VLAN IDs across these interfaces. In such cases, the VLAN IDs (VIDs) must be translated into unique VIDs to prevent collisions. Furthermore, if the number of VLANs being multiplexed onto a single VPWS service tunnel exceeds 4095, then a single tag to double tag translation must be performed. This translation of VIDs

into unique VIDs (either single or double) is referred to as "VID normalization".

When a single normalized VID is used, the lower 12 bits of the Ethernet tag field in EVPN routes MUST be set to that VID. When a double normalized VID is used, the lower 12 bits of the Ethernet tag field MUST be set to the inner VID, while the higher 12 bits are set to the outer VID. As stated in [RFC8214], 12-bit and 24-bit VPWS service instance identifiers representing normalized VIDs MUST be right-aligned.

Since there is only a single EVPN VPWS service tunnel associated with many normalized VIDs (either single or double) across multiple physical interfaces, an MPLS lookup at the disposition PE is no longer sufficient to forward the packet to the correct egress endpoint or interface. Therefore, in addition to an EVPN label lookup corresponding to the VPWS service tunnel, a VID lookup (either single or double) is also required. At the disposition PE, the EVPN label lookup identifies a VID-VRF, and the lookup of the normalized VID(s) within that table identifies the appropriate egress endpoint or interface. The tag manipulation (translation from normalized VID(s) to the local VID) SHOULD be performed either as part of the VID table lookup or at the egress interface itself.

Since the VID lookup (single or double) needs to be performed at the disposition PE, VID normalization MUST be completed prior to MPLS encapsulation on the ingress PE. This requires that both the imposition and disposition PE devices be capable of VLAN tag manipulation, such as rewriting (single or double), addition, or deletion (single or double) at their endpoints (e.g., their ESs, MAC-VRFs, IP-VRFs, etc.). Operators should be informed of potential trade-offs from a performance standpoint, compared to typical PW processing.

3.1. VPWS Service Identifiers

In [RFC8214], a unique value identifying the service is signaled in the context of each PE's EVI. The 32-bit Ethernet Tag ID field MUST be set to this VPWS service instance identifier value. Translation at an ASBR is needed if re-advertising to another AS affects uniqueness.

For FXC, this same Ethernet Tag ID field value is an identifier which may represent:

* VLAN-Bundle : a unique value for a group of VLANs ;

* VLAN-Aware Bundle : a unique value for individual VLANs, and is considered same as the normalised VID.

Both the VPWS service instance identifier and normalised VID are carried in the Ethernet Tag ID field of the Ethernet A-D per EVI route. For FXC, in the case of a 12-bit ID the VPWS service instance identifier is the same as the single-tag normalised VID and will be the same on both VPWS service endpoints. Similarly in the case of a 24-bit ID, the VPWS service instance identifier is the same as the double-tag normalised VID.

3.2. Default Flexible Xconnect

In this mode of operation, many ACs across several Ethernet Segments are multiplexed into a single EVPN VPWS service tunnel represented by a single VPWS service ID. This is the default mode of operation for FXC and the participating PEs do not need to signal the VLANs (normalized VIDs) in EVPN BGP.

Regarding the data-plane aspects of this solution, both imposition and disposition Provider Edge (PE) devices MUST be aware of the VLANs as the imposition PE performs VID normalization and the disposition PE carries out VID lookup and translation. There SHOULD ideally be a single point-to-point (P2P) EVPN VPWS service tunnel between a pair of PEs for a specific set of Attachment Circuits (ACs).

As previously mentioned, because the EVPN VPWS service tunnel is employed to multiplex ACs across various Ethernet Segments (ESs) or physical interfaces, the EVPN label alone is not sufficient for accurate forwarding of the received packets over the MPLS/IP network to egress interfaces. Therefore, normalized VID lookup is REQUIRED in the disposition direction to forward packets to their proper egress end-points; the EVPN label lookup identifies a VID-VRF, and a subsequent normalized VID lookup in that table identifies the egress interface.

In this solution, for each PE, the single-homing ACs represented by their normalized VIDs are associated with a single VPWS service instance within a specific EVI. The generated EVPN route is an Ethernet A-D per EVI route with and ESI of 0, and Ethernet Tag field set to the VPWS service instance ID, and the MPLS label field set to a dynamically generated EVPN service label representing the EVPN VPWS service tunnel. This route is sent with a Route Target (RT) that represents the EVI, which can be auto-generated from the EVI according to Section 5.1.2.1 of [RFC8365]. Additionally, this route is sent with the EVPN Layer-2 Extended Community defined in Section 3.1 of [RFC8214] with two new flags (outlined in Section 4) that indicate: 1) this VPWS service tunnel is for the default

Flexible Cross-Connect, and 2) the normalized VID type (single versus double). The receiving PE uses these new flags for a consistency check and MAY generate an alarm if it detects inconsistencies, but it will not disrupt the VPWS service.

It should be noted that in this mode of operation, a single Ethernet A-D per EVI route is transmitted upon the configuration of the first Attachment Circuit (AC) with the normalized VID. As additional ACs are configured and associated with this EVPN VPWS service tunnel, the PE does not advertise any additional EVPN BGP routes and only associates locally these ACs with the pre-established VPWS service tunnel.

3.2.1. Multi-homing

The default FXC mode can also be used for multi-homing. In this mode, a group of normalized VIDs representing ACs on a single Ethernet Segment, all destined to a single endpoint, are multiplexed into a single EVPN VPWS service tunnel which is identified by a unique VPWS service ID. When employing the default FXC mode for multi-homing, rather than using a single EVPN VPWS service tunnel there may be multiple service tunnels per pair of PEs. Specifically, there is one tunnel for each group of VIDs per pair of PEs, and there can be many such groups between a pair of PEs, resulting in numerous EVPN service tunnels.

3.3. VLAN-Signaled Flexible Xconnect

In this mode of operation, similar to the default FXC mode described in Section 3.2, many normalized VIDs representing ACs across several Ethernet Segments/interfaces are multiplexed into a single EVPN VPWS service tunnel. However, this single tunnel is represented by multiple VPWS service IDs (one per normalized VID) and these normalized VIDs are signaled using EVPN BGP.

In this solution, on each Provider Edge (PE), the multi-homing ACs represented by their normalized VIDs are configured with a single EVI. There is no need to configure a separate VPWS service instance ID in here, as it corresponds to the normalized VID. For each normalized VID on each Ethernet Segment, the PE generates an Ethernet A-D per EVI route where the ESI field represents the ES ID, the Ethernet Tag field is set to the normalized VID, and the MPLS label field is set to a dynamically generated EVPN label representing the P2P EVPN service tunnel. This label is the same for all ACs multiplexed into a single EVPN VPWS service tunnel. This route is sent with a Route Target (RT) representing the EVI. As before, this RT can be auto-generated from the EVI per section Section 5.1.2.1 of [RFC8365]. Additionally, this route includes the EVPN Layer-2

Extended Community defined in Section 3.1 of [RFC8214] with two new flags (outlined in Section 4) that indicate: 1) this VPWS service tunnel is for VLAN-signaled Flexible Cross-Connect, and 2) the normalized VID type (single versus double). The receiving PE uses these new flags for a consistency check and may generate an alarm if it detects inconsistency, but it will not disrupt the VPWS service.

It should be noted that in this mode of operation, the PE sends a single Ethernet A-D per EVI route for each AC that is configured. Each normalized VID that is configured per ES results in generation of an Ethernet A-D per EVI.

This mode of operation enabled automatic cross-checking of normalized VIDs used for Ethernet Virtual Private Line (EVPL) services because these VIDs are signaled in EVPN BGP. For instance, if the same normalized VID is configured on three PE devices (instead of two) for the same EVI, then when a PE receives the second remote Ethernet A-D per EVI route, it generates an error message unless the two Ethernet A-D per EVI routes include the same ESI. Such cross-checking is not feasible in the default FXC mode because the normalized VIDs are not signaled.

3.3.1. Local Switching

When cross-connection occurs between two ACs belonging to two multi-homed Ethernet Segments on the same set of multi-homing PEs, the forwarding between the two ACs must be performed locally during normal operation (e.g., in absence of a local link failure). This means that traffic between the two ACs MUST be locally switched within the PE.

In terms of control plane processing, this means that when the receiving PE processes an Ethernet A-D per EVI route whose ESI is a local ESI, the PE does not modify its forwarding state based on the received route. This approach ensures that local switching takes precedence over forwarding via the MPLS/IP network. This method of prioritizing locally switched traffic aligns with the baseline EVPN principles described in [RFC7432], where locally switched preference is specified for MAC/IP routes.

In such scenarios, the Ethernet A-D per EVI route should be advertised with the MPLS label either associated with the destination Attachment Circuit or with the destination Ethernet Segment in order to avoid any ambiguity in forwarding. In other words, the MPLS label cannot represent the same VID-VRF outlined in Section 3.3, as the same normalized VID can be reachable via two Ethernet Segments. In the case of using an MPLS label per destination AC, this approach can also be applied to VLAN-based VPWS or VLAN-bundle VPWS services as per [RFC8214].

3.4. Service Instantiation

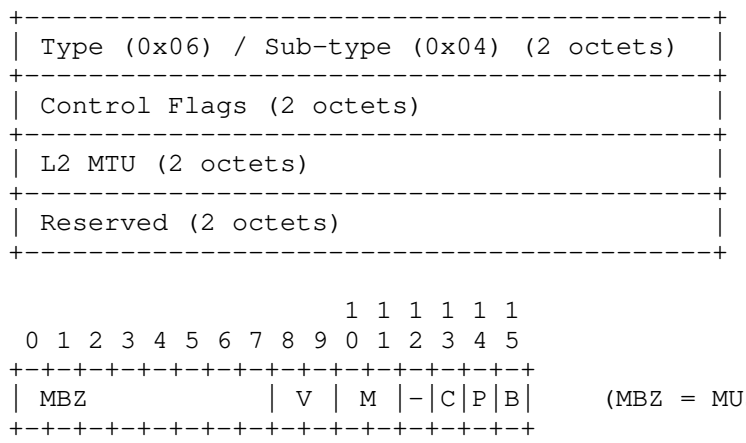
The V field defined in Section 4 is OPTIONAL. However, if transmitted, its value may indicate an error condition that could lead to operational issues. In such cases, merely notifying the operator of an error is insufficient; the VPWS service tunnel must not be established.

If both endpoints of a VPWS tunnel are signaling a matching Normalised VID in the control plane, but one is operating in single-tag mode and the other in double-tag mode, the signaling of the V-bit facilitates the detection and prevention of this tunnel's instantiation.

If single VID normalization is signaled in the Ethernet Tag ID field (12 bits) yet dataplane is operating based on double tags, the VID normalization applies only to outer tag. Conversely, if double VID normalization is signaled in the Ethernet Tag ID field (24 bits), VID normalization applies to both the inner and outer tags.

4. BGP Extensions

This draft uses the EVPN Layer-2 attribute extended community as defined in [RFC8214] with two additional flags incorporated into this Extended Community (EC) as detailed below. This EC is sent with Ethernet A-D per EVI route per Section 3, and SHOULD be sent for both Single-Active and All-Active redundancy modes.



The following bits in the Control Flags are defined; the remaining bits MUST be set to zero when sending and MUST be ignored when receiving this community.

Name	Meaning
B,P,C	per definition in [RFC8214]
-	reserved for Flow-label
M	00 mode of operation as defined in [RFC8214] 01 VLAN-Signaled FXC 10 Default FXC
V	00 operating per [RFC8214] 01 single-VID normalization 10 double-VID normalization

The M and V fields are OPTIONAL. The M field is ignored at reception for forwarding purposes and is used for error notifications.

5. Failure Scenarios

Two examples will be used as an example to analyze the failure scenarios.

The first scenario is a default Flexible Xconnect with Multi-Homing solution and it is depicted in Figure 1. In this case, VID Normalization is performed and a single Ethernet A-D per EVI route is

sent for the bundle of ACs on an ES. That is, PE1 will advertise two Ethernet A-D per EVI routes: the first one will identify the ACs on port p1's ES and the second one will identify the AC2 in port p2's ES. Similarly, PE2 will advertise two Ethernet A-D per EVI routes.

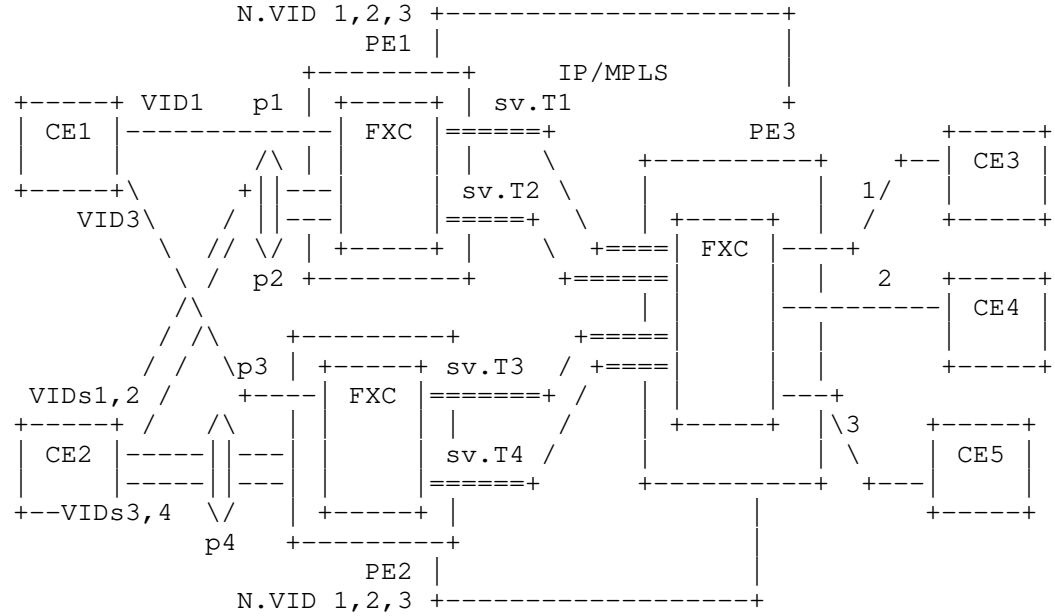


Figure 1: Default Flexible Xconnect

The second scenario, depicted in Figure 2, illustrates the VLAN-signaled FXC mode with Multi-Homing. In this example:

* CE1 is connected to PE1 and PE2 via (port,VID)=(p1,1) and (p3,3), respectively. CE1's VIDs are normalized to value 1 on both PEs, and CE1 is cross-connected to CE3's VID 1 at the remote end.

* CE2 is connected to PE1 and PE2 via ports p2 and p4 respectively:

  - The combinations (p2,1) and (p4,3) identify the ACs used to cross-connect CE2 to CE4's VID 2, and are normalized to value 2.

  - The combinations (p2,2) and (p4,4) identify the ACs used to cross-connect CE2 to CE5's VID 3, and are normalized to value 3.

In this scenario, PE1 and PE2 advertise an Ethernet A-D per EVI route for each normalized VID (values 1, 2 and 3). However, only two VPWS Service Tunnels are required: VPWS Service Tunnel 1 (sv.T1) between PE1's FXC service and PE3's FXC, and VPWS Service Tunnel 2 (sv.T2) between PE2's FXC and PE3's FXC.

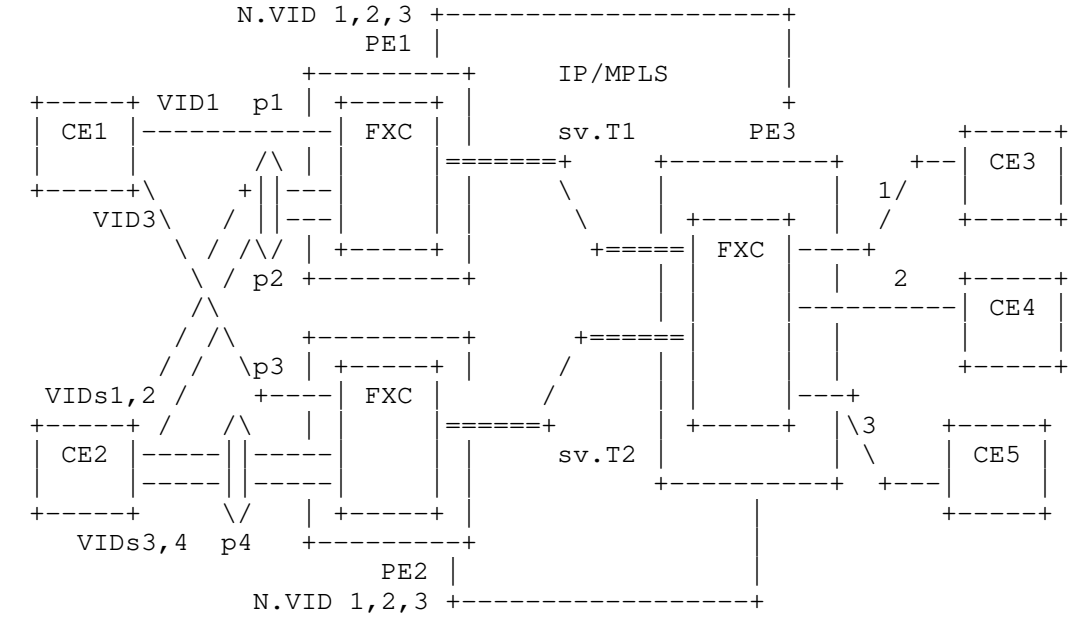


Figure 2: VLAN-Signaled Flexible Xconnect

5.1. EVPN VPWS Service Failure

The failure detection of an EVPN VPWS service can be performed via OAM mechanisms such as VCCV-BFD and upon such failure detection, the switch over procedure to the backup S-PE is the same as the one described above.

5.2. Attachment Circuit Failure

In the event of an AC failure, the VLAN-Signaled and default FXC modes exhibit distinct behaviors:

* Default FXC (Figure 1): in the default mode, a VLAN or AC failure is not signaled. Consequently, in case of an AC failure such as VID1 on CE2, there is nothing to prevent PE3 from directing traffic from CE4 to PE1, leading to a potential black hole. Application layer Operations, Administration, and Maintenance (OAM) may be utilized if per-VLAN fault propagation is necessary in this scenario.

* VLAN-Signaled FXC (Figure 2): in the case of a VLAN or AC failure such as VID1 on CE2, triggers the withdrawal of the Ethernet A-D per EVI route for the corresponding Normalized VID, specifically Ethernet-Tag 2. Upon receiving the route withdrawal, PE3 will remove PE1 from its outgoing path list for traffic originating from CE4.

5.3. PE Port Failure

In the event of a PE port failure, the failure will be signaled, and the other PE will assume forwarding in both scenarios:

* Default FXC (Figure 1): In the case of a port failure, such as p2, the route for Service Tunnel 2 (sv.T2) will be withdrawn. Upon receiving the fault notification, PE3 will remove PE1 from its path list for traffic originating from CE4 and CE5.

* VLAN-Signaled FXC (Figure 2): A port failure, such as p2, triggers the withdrawal of the Ethernet A-D per EVI routes for Normalized VIDs 2 and 3, along with the withdrawal of the Ethernet A-D per ES route for p2's ES. Upon receiving the fault notification, PE3 will remove PE1 from its path list for the traffic originating from CE4 and CE5.

5.4. PE Node Failure

In the case of PE node failure, the operation is similar to the steps described above, albeit that EVPN route withdrawals are performed by the Route Reflector instead of the PE.

6. Security Considerations

Since this document describes a muxing capability which leverages EVPN-VPWS signaling, no additional functionality beyond the muxing service is added and thus no additional security considerations are needed beyond what is already specified in [RFC8214].

7. IANA Considerations

This document requests allocation of bits 8-11 in the "EVPN Layer 2 Attributes Control Flags" registry with names M and V:

M	Signaling mode of operation (2 bits)
V	VLAN-ID normalization (2 bits)

8. References

8.1. 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>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

8.2. Informative References

[I-D.ietf-rtgwg-bgp-pic] Bashandy, A., Filsfils, C., and P. Mohapatra, "BGP Prefix Independent Convergence", Work in Progress, Internet-Draft, draft-ietf-rtgwg-bgp-pic-21, 7 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-rtgwg-bgp-pic-21>.

[RFC5659] Bocci, M. and S. Bryant, "An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge", RFC 5659, DOI 10.17487/RFC5659, October 2009, <https://www.rfc-editor.org/info/rfc5659>.

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

Appendix A. Contributors

In addition to the authors listed on the front page, the following co-authors have also contributed substantially to this document:

Wen Lin
Juniper Networks

EMail: wlin@juniper.net

Luc Andre Burdet
Cisco

EMail: lburdet@cisco.com

Authors' Addresses

Ali Sajassi (editor)
Cisco Systems
Email: sajassi@cisco.com

Patrice Brissette
Cisco Systems
Email: pbrisset@cisco.com

James Uttaro
AT&T
Email: uttaro@att.com

John Drake
Juniper Networks
Email: jdrake@juniper.net

Sami Boutros
Ciena
Email: sboutros@ciena.com

Jorge Rabadan
Nokia
Email: jorge.rabadan@nokia.com

BIER
Internet-Draft
Intended status: Standards Track
Expires: 23 May 2025

Z. Zhang
Juniper Networks

19 November 2024

BIER Penultimate Hop Popping
draft-ietf-bier-php-13

Abstract

This document specifies a mechanism for Penultimate Hop Popping (PHP) in the Bit Index Explicit Replication (BIER) architecture. PHP enables the removal of the BIER header by the penultimate router, thereby reducing the processing burden on the final router in the delivery path. This extension to BIER enhances operational efficiency by optimizing packet forwarding in scenarios where the final hop's capabilities or requirements necessitate such handling. The document details the necessary extensions to the BIER encapsulation and forwarding processes to support PHP, providing guidance for implementation and deployment within BIER-enabled networks.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 23 May 2025.

Table of Contents

1. Introduction

The Bit Index Explicit Replication (BIER) architecture [RFC8279] consists of three layers: the "routing underlay", the "BIER layer", and the "multicast flow overlay". The multicast flow overlay is responsible for allowing BIER Forwarding Egress Routers (BFERs) to signal to BIER Forwarding Ingress Routers (BFIRs) their interest in receiving specific multicast flows, enabling BFIRs to encode the appropriate bitstring for forwarding by the BIER layer.

Multicast Virtual Private Network (MVPN) [RFC6513] [RFC6514] and Ethernet VPN (EVPN) [RFC7432] are two analogous overlays in which BGP Auto-Discovery routes for MVPN/EVPN are exchanged among all Provider Edge (PE) routers to signal which PEs should receive multicast traffic for all or certain flows. Typically, a consistent provider tunnel type is used for traffic delivery to all receiving PEs.

In a deployment scenario where MVPN/EVPN is in use and a sufficient number of provider routers support BIER, BIER can become the preferred provider tunnel type [RFC8556] [RFC9624] . However, some PEs may lack the capability to support BIER forwarding. While it is possible for an ingress PE to send traffic to some PEs using one type of tunnel and to others using a different type, such a procedure can be complex and may result in suboptimal forwarding.

A potential solution to this issue is the use of Penultimate Hop Popping (PHP), whereby the upstream BFR pops the BIER header [RFC8296] and transmits the payload directly. This transmission can occur either directly or indirectly through any type of tunnel to the PE. This mechanism is analogous to Multi-Protocol Label Switching (MPLS) PHP, except that the BIER header is removed.

The transition from an existing MVPN/EVPN deployment with conventional provider tunnels to a BIER-based solution, where some PEs are not BIER-capable, can be incremental. Initially, all PEs are upgraded to support BIER in the control plane, with those unable to perform BIER forwarding requesting PHP. Subsequently, BIER-capable ingress PEs can independently and incrementally switch to BIER transport.

While MVPN/EVPN is used as an example in the above discussion, BIER PHP is applicable to any scenario where the multicast flow overlay edge router does not support BIER, provided that the edge router does not need to identify the transmitting BFIR or participate in BIER Operations, Administration, and Maintenance (OAM) procedures.

This approach is effective when a BIER-incapable PE only needs to receive multicast traffic. However, if the PE also needs to send multicast traffic, it must perform Ingress Replication to a BIER-capable helper PE, which will then relay the packet to other PEs. The helper PE may be a Virtual Hub as defined in [RFC7024] for MVPN and [I-D.ietf-bess-evpn-virtual-hub] for EVPN, or an AR-Replicator as defined in [RFC9574] for EVPN.

2. Specifications

The BIER Penultimate Hop Popping (PHP) mechanism is designed specifically for scenarios where a multicast flow overlay router within a BIER domain does not support BIER forwarding, either completely or for specific BitStringLengths (BSL). In the latter case, PHP applies only to BIER packets with those particular BSLs. If the flow overlay router were capable of BIER forwarding, it would function as a BFER, and PHP would not be performed by the penultimate hop.

The procedures outlined in this section are applicable only if, through means outside the scope of this document, it is established that all potential penultimate hop BFRs are capable of supporting PHP (i.e., able to remove the BIER header when forwarding to a requesting flow overlay router) and that the payload following the BIER header is one of the following:

* MPLS packets with a downstream-assigned label at the top of the stack (i.e., the Proto field in the BIER header is set to 1). For instance, a label from a Domain-wide Common Block (DCB) as specified in [RFC9573].

* IPv4/IPv6 multicast packets for which the Reverse Path Forwarding (RPF) check is disabled.

2.1. Signaling

In IS-IS signaling, a sub-TLV nested within another sub-TLV is referred to as a sub-sub-TLV (and further levels are possible, such as sub-sub-sub-TLV). In other signaling protocols, a sub-TLV nested within another sub-TLV is still referred to as a sub-TLV. For convenience, this document uses the term "sub-TLV" even when referring to a sub-sub-TLV in IS-IS, as there is no ambiguity in the terminology (e.g., MPLS Encapsulation).

A BIER-incapable router, when functioning as a multicast flow overlay router for BIER, MUST signal its BIER information as specified in [RFC8401], [RFC8444], [I-D.ietf-bier-ospfv3-extensions], or [I-D.ietf-bier-idr-extensions], including a PHP sub-TLV within the BIER sub-TLV (or TLV in the case of BGP) attached to the BIER-incapable router's BFR-prefix to request BIER PHP from other BFRs. The type of the sub-TLV or sub-sub-TLV is TBD, and the length is 0.

For MPLS encapsulation, the BIER-incapable multicast flow overlay router MAY omit the BIER MPLS Encapsulation sub-TLV, or it MUST set the Label field in the BIER MPLS Encapsulation sub-TLV to the Implicit Null Label [RFC3032].

In the case of MPLS encapsulation, if a BFER (which supports BIER but not a specific BSL) does not support a particular BSL, it MAY advertise a corresponding BIER MPLS Encapsulation sub-TLV with the Label field set to the Implicit Null Label to request PHP for that BSL. In this scenario, the PHP sub-TLV MUST NOT be included.

For non-MPLS encapsulation [I-D.ietf-bier-lsr-non-mpls-extensions], the BIER-incapable multicast flow overlay router MAY omit the BIER non-MPLS Encapsulation sub-TLV, or it MUST set the BIFT-id field in the BIER non-MPLS Encapsulation sub-TLV to 0.

Similarly, for non-MPLS encapsulation, if a BFER (which supports BIER but not a specific BSL) does not support a particular BSL, it MAY advertise a corresponding BIER non-MPLS Encapsulation sub-TLV but set the BIFT-id field to 0 to request PHP for that BSL. In this scenario, the PHP sub-TLV MUST NOT be included.

2.2. BIRT/BIFT Calculation

If a BFR adheres to Section 6.9 of [RFC8279] for handling BIER-incapable routers, it MUST treat a router as BIER-incapable for a specific BSL if the label in the corresponding MPLS Encapsulation sub-TLV advertised by the router is Implicit Null, or if the BIFT-id in the corresponding non-MPLS Encapsulation sub-TLV is 0. Additionally, the router MUST be treated as BIER-incapable for all BSLs if it advertises a PHP sub-TLV. Consequently, the router will not be utilized as a transit BFR for certain or all BSLs.

When the downstream neighbor (whether determined through IGP calculation or indicated in the BIER Nexthop sub-TLV in the case of BGP) for a BFR-prefix is the router that advertises the prefix with a PHP sub-TLV, an Implicit Null Label in its BIER MPLS Encapsulation sub-TLV, or a BIFT-id of 0 in its BIER non-MPLS Encapsulation sub-TLV, then, upon the creation or update of the corresponding BIRT or BIFT entry, the forwarding behavior MUST be that the BIER header is removed and the payload is forwarded to the downstream router without the BIER header, either directly or over any type of tunnel.

3. Security Considerations

This specification does not introduce additional security concerns beyond those already discussed in BIER architecture and OSPF/IS-IS/ BGP extensions for BIER signaling.

4. Operational Considerations

BIER PHP can only be used when the conditions specified in Section 2 are met. The BIER OAM functionality is not available on the BIER-incapable flow overlay routers, but using PHP when the conditions are met is simpler than the alternative of using BIER to send to some whereas using non-BIER tunnels to send to other flow overlay routers.

5. IANA Considerations

This document requests a new sub-sub-TLV type value from the "Sub-sub-TLVs for BIER Info Sub-TLV" registry within the "IS-IS TLV Codepoints" registry:

Type	Name
TBD	BIER PHP Request

This document requests a new sub-TLV type value from the OSPFv2 Extended Prefix TLV Sub-TLV registry:

Type	Name
TBD	BIER PHP Request

This document requests a new sub-TLV type value from the OSPFv3 Extended LSA Sub-TLVs registry:

Type	Name	L2BM
TBD	BIER PHP Request	X

This document requests a new sub-TLV type value from the BGP BIER TLV sub-TLV Types registry requested in [I-D.ietf-bier-idr-extensions]:

Type	Name
TBD	BIER PHP Request

6. Acknowledgements

The author wants to thank Eric Rosen and Antonie Przygienda for their review, comments and suggestions. The author also wants to thank Senthil Dhanaraj for his suggestion of requesting PHP if a BFER does not support certain BSL.

7. References

7.1. Normative References

[I-D.ietf-bier-idr-extensions]
Xu, X., Chen, M., Patel, K., Wijnands, I., Przygienda, T., and Z. J. Zhang, "BGP Extensions for BIER", Work in Progress, Internet-Draft, draft-ietf-bier-idr-extensions-14, 4 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-bier-idr-extensions-14>.

[I-D.ietf-bier-lsr-non-mpls-extensions]
Dhanaraj, S., Yan, G., Wijnands, I., Psenak, P., Zhang, Z. J., and J. Xie, "LSR Extensions for BIER non-MPLS Encapsulation", Work in Progress, Internet-Draft, draft-

ietf-bier-lsr-non-mpls-extensions-03, 7 February 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-bier-lsr-non-mpls-extensions-03>.

[I-D.ietf-bier-ospfv3-extensions] Psenak, P., Nainar, N. K., and I. Wijnands, "OSPFv3 Extensions for BIER", Work in Progress, Internet-Draft, draft-ietf-bier-ospfv3-extensions-07, 1 December 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-bier-ospfv3-extensions-07>.

[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>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <https://www.rfc-editor.org/info/rfc3032>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

[RFC8401] Ginsberg, L., Ed., Przygienda, T., Aldrin, S., and Z. Zhang, "Bit Index Explicit Replication (BIER) Support via IS-IS", RFC 8401, DOI 10.17487/RFC8401, June 2018, <https://www.rfc-editor.org/info/rfc8401>.

[RFC8444] Psenak, P., Ed., Kumar, N., Wijnands, IJ., Dolganow, A., Przygienda, T., Zhang, J., and S. Aldrin, "OSPFv2 Extensions for Bit Index Explicit Replication (BIER)", RFC 8444, DOI 10.17487/RFC8444, November 2018, <https://www.rfc-editor.org/info/rfc8444>.

[RFC8556] Rosen, E., Ed., Sivakumar, M., Przygienda, T., Aldrin, S., and A. Dolganow, "Multicast VPN Using Bit Index Explicit Replication (BIER)", RFC 8556, DOI 10.17487/RFC8556, April 2019, <https://www.rfc-editor.org/info/rfc8556>.

[RFC9573] Zhang, Z., Rosen, E., Lin, W., Li, Z., and IJ. Wijnands, "MVPN/EVPN Tunnel Aggregation with Common Labels", RFC 9573, DOI 10.17487/RFC9573, May 2024, <https://www.rfc-editor.org/info/rfc9573>.

[RFC9624] Zhang, Z., Przygienda, T., Sajassi, A., and J. Rabadan, "EVPN Broadcast, Unknown Unicast, or Multicast (BUM) Using Bit Index Explicit Replication (BIER)", RFC 9624, DOI 10.17487/RFC9624, August 2024, <https://www.rfc-editor.org/info/rfc9624>.

7.2. Informative References

[I-D.ietf-bess-evpn-virtual-hub] Patel, K., Sajassi, A., Drake, J., Zhang, Z. J., and W. Henderickx, "Virtual Hub-and-Spoke in BGP EVPNs", Work in Progress, Internet-Draft, draft-ietf-bess-evpn-virtual-hub-00, 26 January 2020, <https://datatracker.ietf.org/doc/html/draft-ietf-bess-evpn-virtual-hub-00>.

[RFC6513] Rosen, E., Ed. and R. Aggarwal, Ed., "Multicast in MPLS/BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <https://www.rfc-editor.org/info/rfc6513>.

[RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, DOI 10.17487/RFC6514, February 2012, <https://www.rfc-editor.org/info/rfc6514>.

[RFC7024] Jeng, H., Uttaro, J., Jalil, L., Decraene, B., Rekhter, Y., and R. Aggarwal, "Virtual Hub-and-Spoke in BGP/MPLS VPNs", RFC 7024, DOI 10.17487/RFC7024, October 2013, <https://www.rfc-editor.org/info/rfc7024>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC9574] Rabadan, J., Ed., Sathappan, S., Lin, W., Katiyar, M., and A. Sajassi, "Optimized Ingress Replication Solution for Ethernet VPNs (EVPNs)", RFC 9574, DOI 10.17487/RFC9574, May 2024, <https://www.rfc-editor.org/info/rfc9574>.

Author's Address

Zhaohui Zhang
Juniper Networks
Email: zzhang@juniper.net

Delay/Disruption Tolerant Networking
Internet-Draft
Updates: [9171, 7116, 6260] (if approved)
Intended status: Standards Track
Expires: 31 March 2025

R. Taylor
Aalyria Technologies
E. Birrane
JHU/APL

27 September 2024

Update to the ipn URI scheme
draft-ietf-dtn-ipn-update-14

Abstract

This document updates the specification of the ipn URI scheme previously defined in RFC 6260, the IANA registries established in RFC 7116, and the rules for the encoding and decoding of these URIs when used as an Endpoint Identifier (EID) in Bundle Protocol Version 7 (BPv7) as defined in RFC 9171. These updates clarify the structure and behavior of the ipn URI scheme, define new encodings of ipn scheme URIs, and establish the registries necessary to manage this scheme.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://ricktaylor.github.io/ipn2/draft-taylor-dtn-ipn-update.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-dtn-ipn-update/.

Discussion of this document takes place on the Delay/Disruption Tolerant Networking Working Group mailing list (mailto:dtn@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/dtn/. Subscribe at https://www.ietf.org/mailman/listinfo/dtn/.

Source for this draft and an issue tracker can be found at https://github.com/ricktaylor/ipn2.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The ipn URI scheme was originally defined in [RFC6260] and [RFC7116] as a way to identify network nodes and node services using concisely-encoded integers that can be processed faster and with fewer resources than other verbose identifier schemes. The scheme was designed for use with the experimental Bundle Protocol version 6 (BPv6, [RFC5050]) and IPN was defined as an acronym for the term "InterPlanetary Network" in reference to its intended use for deep-space networking. Since then, the efficiency benefit of integer identifiers makes ipn scheme URIs useful for any networks operating with limited power, bandwidth, and/or compute budget. Therefore, the term IPN is now used as a non-acronymous name.

Similar to the experimental BPv6, the standardized Bundle Protocol version 7 (BPv7, [RFC9171]) codifies support for the use of the ipn URI scheme for the specification of bundle Endpoint Identifiers (EIDs). The publication of BPv7 has resulted in operational deployments of BPv7 nodes for both terrestrial and non-terrestrial use cases. This includes BPv7 networks operating over the terrestrial Internet and BPv7 networks operating in self-contained environments behind a shared administrative domain. The growth in the number and scale of deployments of BPv7 has been accompanied by a growth in the usage of the ipn URI scheme which has highlighted areas to improve the structure, moderation, and management of this scheme.

By updating [RFC7116] and [RFC9171], this document updates the specification of the ipn URI scheme, in a backwards-compatible way, to provide needed improvements both in the scheme itself and its usage to specify EIDs with BPv7. Specifically, this document introduces a hierarchical structure for the assignment of ipn scheme URIs, clarifies the behavior and interpretation of ipn scheme URIs, defines efficient encodings of ipn scheme URIs, and updates/defines the registries associated for this scheme.

Although originally developed by the deep space community for use with Bundle Protocol, the ipn URI scheme is sufficiently generic to be used in other environments where a concise unique representation of a resource on a particular node is required.

It is important to remember that, like most other URI schemes, the ipn URI scheme defines a unique identifier of a resource, and does not include any topological information describing how to route messages to that resource.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

For the remainder of this document, the term "ipn URI" is used to refer to a URI that uses the ipn scheme.

3. Core Concepts

Every ipn URI, no matter whether it is expressed with the textual representation or a binary encoding, MUST be considered as a tuple of the following three components:

* The Allocator Identifier
* The Node Number
* The Service Number

The Allocator Identifier indicates the entity responsible for assigning Node Numbers to individual resource nodes, maintaining uniqueness whilst avoiding the need for a single registry for all assigned Node Numbers. See Allocator Identifiers (Section 3.2).

The Node Number is a shared identifier assigned to all ipn URIs for resources co-located on a single node. See Node Numbers (Section 3.3).

The Service Number is an identifier to distinguish between resources on a given node. See Service Numbers (Section 3.5).

The combination of these three components guarantees that every correctly constructed ipn URI uniquely identifies a single resource. Additionally, the combination of the Allocator Identifier and the Node Number provides a mechanism to uniquely identify the node on which a particular resource is expected to exist. See Fully-qualified Node Number (Section 3.3.1).

3.1. The Null ipn URI

It has been found that there is value in defining a unique ’null’ ipn URI to indicate "nowhere". This ipn URI is termed the "Null ipn URI", and has all three components: Allocator Identifier, Node Number, and Service Number, set to the value zero (0). No resource identified by Null ipn URI exists, and any destination identified by such a resource is therefore by definition unreachable.

3.2. Allocator Identifiers

An Allocator is any organization that wishes to assign Node Numbers for use with the ipn URI scheme, and has the facilities and governance to manage a public registry of assigned Node Numbers. The authorization to assign these numbers is provided through the assignment of an Allocator Identifier by IANA. Regardless of other attributes of an Allocator, such as a name, point of contact, or other identifying information, Allocators are identified by Allocator Identifiers: a unique, unsigned integer, in the range 0 to 2^32-1.

The Allocator Identifier MUST be the sole mechanism used to identify the Allocator that has assigned the Node Number in an ipn URI. An Allocator may have multiple assigned Allocator Identifiers, but a given Allocator Identifier MUST only be associated with a single Allocator.

A new IANA "’ipn’ Scheme URI Allocator Identifiers" registry is defined for the registration of Allocator Identifiers, see ’ipn’ Scheme URI Allocator Identifiers registry (Section 9.1). Although the uniqueness of Allocator Identifiers is required to enforce uniqueness of ipn URIs, some identifiers are explicitly reserved for experimentation or future use.

Each Allocator assigns Node Numbers according to its own policies, without risk of creating an identical ipn URI, as permitted by the rules in the Node Numbers (Section 3.3) section of this document. Other than ensuring that any Node Numbers it allocates are unique amongst all Node Numbers it assigns, an Allocator does not need to coordinate its allocations with other Allocators.

If a system does not require interoperable deployment of ipn scheme URIs, then the Private Use Node Numbers (Section 3.4.3) range, reserved by the Default Allocator (Section 3.2.2) for this purpose, are to be used.

3.2.1. Allocator Identifier Ranges

Some organizations with internal hierarchies may wish to delegate the allocation of Node Numbers to one or more of their sub-organizations. Rather than assigning unique Allocator Identifiers to each sub-organization on a first-come first-served basis, there are operational benefits in assigning Allocator Identifiers to such an organization in a structured way so that an external observer can detect that a group of Allocator Identifiers are organizationally associated.

An Allocator Identifier range is a set of consecutive Allocator Identifiers associated with the same Allocator. Each individual Allocator Identifier in a given range SHOULD be assigned to a distinct sub-organization of the Allocator. Assigning identifiers in this way allows external observers both to associate individual Allocator Identifiers with a single organization and to usefully differentiate amongst sub-organizations.

The practice of associating a consecutive range of numbers with a single organization is inspired by the Classless Inter-domain Routing assignment of Internet Addresses described in [RFC4632]. In that assignment scheme, an organization (such as an Internet Service Provider) is assigned a network prefix such that all addresses sharing that same prefix are considered to be associated with that organization.

Each Allocator Identifier range is identified by the first Allocator Identifier in the range and the number of consecutive identifiers in the range.

Allocator Identifier ranges differ from CIDR addresses in two important ways.

1. Allocator Identifiers are used to identify organizations and are not, themselves, addresses.

2. Allocator Identifiers may be less than 32 bits in length.

In order to differentiate between Allocator Identifier ranges using efficient bitwise operations, all ranges MUST be of a size S that is a power of 2, and for a given range of length N bits, with S = 2^N, the least-significant N bits of the first Allocator Identifier MUST be all 0.

An example of the use of Allocator Identifier ranges for four organizations (A, B, C, and D) is as follows:

Organization	Range (dec)	Range (hex)	Range Length (Bits)
Org A	974848 .. 974975	0xEE000 .. 0xEE07F	7 bits
Org B	974976 .. 974991	0xEE080 .. 0xEE08F	4 bits
Org C	974992 .. 974993	0xEE090 .. 0xEE091	1 bit
Org D	974994	0xEE092	0 bits

Table 1: Allocator Identifier Range Assignment Example

With these assignments, any Allocator Identifier whose most-significant 25 bits match 0xEE000 belong to organization A. Similarly, any Allocator Identifier whose most-significant 28 bits match 0xEE080 belong to organization B, and any Allocator Identifier whose most-significant 31 bits are 0xEE090 belong to organization C. Organization D has a single Allocator Identifier, and hence a range of bit-length 0.

3.2.2. The Default Allocator

As of the publication of [RFC7116], the only organization permitted to assign Node Numbers was the Internet Assigned Numbers Authority (IANA) which assigned Node Numbers via the IANA "CBHE Node Numbers" registry. This means that all ipn URIs created prior to the addition of Allocator Identifiers are assumed to have Node Number allocations that comply with the IANA "CBHE Node Numbers" registry.

The presumption that, unless otherwise specified, Node Numbers are allocated by IANA from a specific registry is formalized in this update to the ipn URI scheme by designating IANA as the Default Allocator, and by assigning the Allocator Identifier zero (0) in the 'ipn' Scheme URI Allocator Identifiers registry (Section 9.1) to the Default Allocator. In any case where an encoded ipn URI does not explicitly include an Allocator Identifier, an implementation MUST assume that the Node Number has been allocated by the Default Allocator.

A new IANA "'ipn' Scheme URI Default Allocator Node Numbers" registry is defined to control the allocation of Node Numbers values by the Default Allocator. This new registry inherits behaviors and existing assignments from the IANA "CBHE Node Numbers" registry, and reserves some other values as defined in the Special Node Numbers (Section 3.4) section below.

3.3. Node Numbers

A Node Number identifies a node that hosts a resource in the context of an Allocator. A Node Number is an unsigned integer. A single Node Number assigned by a single Allocator MUST refer to a single node.

All Node Number assignments, by all Allocators, MUST be in the range 0 to 2^32-1.

It is RECOMMENDED that Node Number zero (0) not be assigned by an Allocator to avoid confusion with the Null ipn URI (Section 3.1).

3.3.1. Fully-qualified Node Numbers

One of the advantages of ipn URIs is the ability to easily split the identity of a particular service from the node upon which the service exists. For example a message received from one particular ipn URI may require a response to be sent to a different service on the same node that sent the original message. Historically the identifier of the sending node has been colloquially referred to as the "node number" or "node identifier".

To avoid future confusion, when referring to the identifier of a particular node the term "Fully-qualified Node Number" (FQNN) MUST be used to refer to the combination of the Node Number component and Allocator Identifier component of an ipn URI that uniquely identifies a particular node. In other words, an FQNN is the unique identifier of a particular node that supports services identified by ipn URIs.

In the examples in this document, FQNNs are written as (Allocator Identifier, Node Number), e.g., (977000,100) is the FQNN for a node assigned Node Number 100 by an Allocator with Allocator Identifier 977000.

3.4. Special Node Numbers

Some special-case Node Numbers are defined by the Default Allocator, see 'ipn' Scheme URI Default Allocator Node Numbers registry (Section 9.2).

3.4.1. The Zero Node Number

The Default Allocator assigns the use of Node Number zero (0) solely for identifying the Null ipn URI (Section 3.1).

This means that any ipn URI with a zero (0) Allocator Identifier and a zero (0) Node Number, but a non-zero Service Number component is invalid. Such ipn URIs MUST NOT be composed, and processors of such ipn URIs MUST consider them as the Null ipn URI.

3.4.2. LocalNode ipn URIs

The Default Allocator reserves Node Number 2^32-1 (0xFFFFFFFF) to specify resources on the local node, rather than on any specific individual node.

This means that any ipn URI with a zero (0) Allocator Identifier and a Node Number of 2^32-1 refers to a service on the local bundle node. This form of ipn URI is termed a "LocalNode ipn URI".

3.4.3. Private Use Node Numbers

The Default Allocator provides a range of Node Numbers that are reserved for "Private Use", as defined in [RFC8126].

Any ipn URI with a zero (0) Allocator Identifier and a Node Number reserved for "Private Use" is not guaranteed to be unique beyond a single administrative domain. An administrative domain, as used here, is defined as the set of nodes that share a unique allocation of FQNNs from the "Private Use" range. These FQNNs can be considered to be functionally similar to "Private Address Space" IPv4 addresses, as defined in [RFC1918].

Because of this lack of uniqueness, any implementation of a protocol using ipn URIs that resides on the border between administrative domains MUST have suitable mechanisms in place to prevent protocol units using such "Private Use" Node Numbers to cross between different administrative domains.

3.5. Service Numbers

A Service Number is an unsigned integer that identifies a particular service operating on a node. A service in this case is some logical function that requires its own resource identifier to distinguish it from other functions operating on the same node.

4. Textual Representation of ipn URIs

All ipn scheme URIs comply with [RFC3986], and are therefore represented by scheme identifier, and a scheme-specific part. The scheme identifier is: ipn, and the scheme-specific parts are represented as a sequence of numeric components separated with the '.' character. A formal definition is provided below, see ipn URI Scheme Text Syntax (Section 4.1), and can be informally considered as:

ipn:[<allocator-identifier>.]<node-number>.<service-number>

To keep the text representation concise, the following rules apply:

1. All leading 0 characters MUST be omitted. A single '0' is valid.

2. If the Allocator Identifier is zero (0), then the <allocator-identifier> and '.' MAY be omitted.

3. If the Allocator Identifier is zero (0), and the Node Number is 2^32-1, i.e., the URI is a LocalNode ipn URI (Section 3.4.2), then the character '!' SHOULD be used instead of the digits 4294967295, although both forms are valid encodings.

Examples of the textual representation of ipn URIs can be found in Appendix A (Appendix A).

4.1. ipn URI Scheme Text Syntax

The text syntax of an ipn URI MUST comply with the following ABNF [RFC5234] syntax, and reiterates the core ABNF syntax rule for DIGIT defined by that specification:

```
ipn-uri = "ipn:" ipn-hier-part

ipn-hier-part = fqnn "." service-number

fqnn = "!" / allocator-part

allocator-part = [allocator-identifier "."] node-number

allocator-identifier = number

node-number = number

service-number = number

number = "0" / non-zero-number

non-zero-number = (%x31-39 *DIGIT)

DIGIT = %x30-39
```

5. Usage of ipn URIs with BPv7

From the earliest days of experimentation with the Bundle Protocol there has been a need to identify the source and destination of a bundle. The IRTF BPv6 experimental specification termed the logical source or destination of a bundle as an "Endpoint" identified by an "Endpoint Identifier" (EID). BPv6 EIDs are formatted as URIs. This definition and representation of EIDs was carried forward from the IRTF BPv6 specification to the IETF BPv7 specification. BPv7 additionally defined an IANA registry called the "Bundle Protocol URI Scheme Types" registry which identifies those URI schemes than might be used to represent EIDs. The ipn URI scheme is one such URI scheme.

This section identifies the behavior and interpretation of ipn scheme URIs that MUST be followed when using this URI scheme to represent EIDs in BPv7. An ipn URI used as a BPv7 or BPv6 EID is termed an "ipn EID".

5.1. Uniqueness Constraints

An ipn EID MUST identify a singleton endpoint. The bundle processing node that is the sole member of that endpoint MUST be the node identified by the Fully-qualified Node Number (Section 3.3.1) of the node.

A single bundle processing node MAY have multiple ipn EIDs associated with it. However, all ipn EIDs that share any single FQNN MUST refer to the same bundle processing node.

For example, ipn:977000.100.1, ipn:977000.100.2, and ipn:977000.100.3 MUST all refer to services registered on the bundle processing node identified with FQNN (977000,100). None of these EIDs could be registered on any other bundle processing node.

5.2. The Null Endpoint

Section 3.2 of [RFC9171] defines the concept of the 'null' endpoint, which is an endpoint that has no members and which is identified by a special 'null' EID.

Within the ipn URI scheme, the 'null' EID is represented by the Null ipn URI (Section 3.1). This means that the URIs dtn:none (Section 4.2.5.1.1 of [RFC9171]), ipn:0.0, and ipn:0.0.0 all refer to the BPv7 'null' endpoint.

5.3. BPv7 Node ID

Section 4.2.5.2 of [RFC9171] introduces the concept of a "Node ID" that has the same format as an EID and that uniquely identifies a bundle processing node.

Any ipn EID can serve as a "Node ID" for the bundle processing node identified by its Fully-qualified Node Number (Section 3.3.1). That is, any ipn EID of the form ipn:A.B.C may be used as the Source Node ID of any bundle created by the bundle processing node identified by the FQNN (A,B).

5.4. LocalNode ipn EIDs

When a LocalNode ipn URI (Section 3.4.2) is used as a BPv7 or BPv6 EID, it is termed a "LocalNode ipn EID".

Because a LocalNode ipn EID only has meaning on the local bundle node, any such EID MUST be considered 'non-routable'. This means that any bundle using a LocalNode ipn EID as a bundle source or bundle destination MUST NOT be allowed to leave the local node. Equally, all externally received bundles featuring LocalNode EIDs as a bundle source or bundle destination MUST be discarded as invalid.

LocalNode ipn EIDs MUST NOT be present in any other part of a bundle that is transmitted off of the local node. For example, a LocalNode ipn EID MUST NOT be used as a Bundle Protocol Security [RFC9172] security source for a bundle transmitted from the local bundle node, because such a source EID would have no meaning at a downstream bundle node.

LocalNode ipn EIDs MUST NOT be published in any node identification directory, such as a DNS registration, or presented as part of dynamic peer discovery, as the EID has no valid meaning for other nodes. For example, a LocalNode ipn EID MUST NOT be advertised as the peer Node ID during session negotiation in [RFC9174].

5.5. Private Use ipn EIDs

Bundles destined for EIDs that use an ipn URI with a Fully-qualified Node Number (Section 3.3.1) that is within the "Private Use" range of the Default Allocator (Section 3.2.2) are not universally unique, and therefore are only valid within the scope of the current administrative domain. This means that any bundle using a Private Use ipn EID as a bundle source or bundle destination MUST NOT be allowed to cross administrative domains. All implementations that could be deployed as a gateway between administrative domains MUST be sufficiently configurable to ensure that this is enforced, and operators MUST ensure correct configuration.

Private Use ipn EIDs MUST NOT be present in any other part of a bundle that is destined for another administrative domain when the lack of uniqueness prevents correct operation. For example, a Private Use ipn EID MUST NOT be used as a Bundle Protocol Security [RFC9172] security source for a bundle, when the bundle is destined for a different administrative domain.

5.6. Well-known Service Numbers

It is convenient for BPv7 services that have a public specification and wide adoption to be identified by a pre-agreed default Service Number, so that unless extra configuration is applied, such services can be sensibly assumed to be operating on the well-known Service Number on a particular node.

If a different service uses the number, or the service uses a different number, BPv7 will continue to operate, but some configuration may be required to make the individual service operational.

A new IANA "'ipn' Scheme URI Well-known Service Numbers for BPv7" registry is defined for the registration of well-known BPv7 Service Numbers, see 'ipn' Scheme URI Well-known Service Numbers for BPv7 registry (Section 9.3). This registry records the assignments of Service Numbers for well-known services, and also explicitly reserves ranges for both experimentation and private use.

5.7. Administrative Endpoints

The service identified by a Service Number of zero (0) MUST be interpreted as the Administrative Endpoint of the node, as defined in Section 3.2 of [RFC9171].

Non-zero Service Numbers MUST NOT be used to identify the Administrative Endpoint of a bundle node in an ipn EID.

6. CBOR representation of ipn URIs with BPv7

Section 4.2.5.1 of [RFC9171] requires that any URI scheme used to represent BPv7 EIDs MUST define how the scheme-specific part of the URI scheme is encoded with CBOR [RFC8949]. To meet this requirement, this section describes the CBOR encoding and decoding approach for ipn EIDs. The formal definition of the CBOR representation is specified, see ipn URI Scheme CBOR syntax (Section 6.3).

6.1. ipn EID CBOR Encoding

Generic URI approaches to encoding ipn EIDs are unlikely to be efficient because they do not consider the underlying structure of the ipn URI scheme. Since the creation of the ipn URI scheme was motivated by the need for concise identification and rapid processing, the encoding of ipn EIDs maintains these properties.

Fundamentally, [RFC9171] ipn EIDs are represented as a sequence of identifiers. In the text syntax, the numbers are separated with the '.' delimiter; in CBOR, this ordered series of numbers can be represented by an array. Therefore, when encoding ipn EIDs for use with BPv7, the scheme-specific part of an ipn URI MUST be represented as a CBOR array of either two (2) or three (3) elements. Each element of the array MUST be encoded as a single CBOR unsigned integer.

The structure and mechanisms of the two-element and three-element encodings are described below, and examples of the different encodings are provided in Appendix B (Appendix B).

6.1.1. Two-Element Scheme-Specific Encoding

In the two-element scheme-specific encoding of an ipn EID, the first element of the array is an encoding of the Fully-qualified Node Number (Section 3.3.1) and the second element of the array is the ipn EID Service Number.

The FQNN encoding MUST be a 64-bit unsigned integer constructed in the following way:

1. The least significant 32 bits MUST represent the Node Number associated with the ipn EID.

2. The most significant 32 bits MUST represent the Allocator Identifier associated with the ipn EID.

For example the ipn EID of ipn:977000.100.1 has an FQNN of (977000,100) which would be encoded as 0xEE868_00000064. The resulting two-element array [0xEE868_00000064, 0x01] would be encoded in CBOR as the following 11 octet sequence:

```
82                       # 2-Element Endpoint Encoding
   02                    # uri-code: 2 (IPN URI scheme)
   82                    # 2 Element ipn EID scheme-specific encoding
      1B 000EE86800000064 # Fully-qualified Node Number
      01                 # Service Number
```

The two-element scheme-specific encoding provides for backwards-compatibility with the encoding provided in Section 4.2.5.1.2 of [RFC9171]. When used in this way, the encoding of the FQNN replaces the use of the "Node Number" that was specified in RFC9171. When the Node Number is allocated by the Default Allocator (Section 3.2.2), the encoding of the FQNN and the RFC9171 encoding of the "Node Number" are identical.

6.1.2. Three-Element Scheme-Specific Encoding

In the three-element scheme-specific encoding of an ipn EID, the first element of the array is the Allocator Identifier, the second element of the array is the Node Number, and the third element of the array is the Service Number.

For example, the ipn EID of ipn:977000.100.1 would result in the three-element array of [977000,100,1] which would be encoded in CBOR as the following 9 octet sequence:

```
82                # 2-Element Endpoint Encoding
   02             # uri-code: 2 (IPN URI scheme)
   83             # 3 Element ipn EID scheme-specific encoding
      1A 000EE868 # Allocator Identifier
      64          # Node Number
      01          # Service Number
```

The three-element scheme-specific encoding allows for a more efficient representation of ipn EIDs using smaller Allocator Identifiers, and implementations are RECOMMENDED to use this encoding scheme, unless explicitly mitigating for interoperability issues, see Scheme Compatibility (Section 7.1).

When encoding an ipn EID using the Default Allocator (Section 3.2.2) with this encoding scheme, the first element of the array is the value zero (0). In this case using the equivalent Two-Element Scheme-Specific Encoding (Section 6.1.1) will result in a more concise CBOR representation, and therefore it is RECOMMENDED that implementations use that encoding instead.

6.2. ipn EID CBOR Decoding

The presence of different scheme-specific encodings does not introduce any decoding ambiguity.

An ipn EID CBOR decoder can reconstruct an ipn EID using the following logic. In this description, the term enc_eid refers to the CBOR encoded ipn EID, and the term ipn_eid refers to the decoded ipn EID.

```
if enc_eid.len() == 3
{
  ipn_eid.allocator_identifier := enc_eid[0];
  ipn_eid.node_number := enc_eid[1];
  ipn_eid.service_number := enc_eid[2];
}
else if enc_eid.len() == 2
{
  ipn_eid.allocator_identifier := enc_eid[0] >> 32;
  ipn_eid.node_number := enc_eid[0] & (2^32-1);
  ipn_eid.service_number := enc_eid[1];
}
```

6.3. ipn URI Scheme CBOR syntax

A BPv7 endpoint identified by an ipn URI, when encoded in Concise Binary Object Representation (CBOR) [RFC8949], MUST comply with the following Concise Data Definition Language (CDDL) [RFC8610] specification:

```
eid = $eid .within eid-structure

eid-structure = [
  uri-code: uint,
  SSP: any
]

; ... Syntax for other uri-code values defined in RFC9171 ...

$eid /= [
  uri-code: 2,
  SSP: ipn-ssp2 / ipn-ssp3
]
ipn-ssp2 = [
  fqnn: uint,  ; packed value
  service-number: uint
]
ipn-ssp3 = [
  allocator-identifier: uint .lt 4294967296,
  node-number: uint .lt 4294967296,
  service-number: uint
]
```

Note: The node-number component will be the numeric representation of the concatenation of the Allocator Identifier and Node Number when the 2-element encoding scheme has been used.

6.4. ipn EID Matching

Regardless of whether the two-element or three-element scheme-specific encoding is used, ipn EID matching MUST be performed on the decoded EID information itself. Different encodings of the same ipn EID MUST be treated as equivalent when performing EID-specific functions.

For example, the ipn EID of ipn:977000.100.1 can be represented as either the two-element encoding of 0x821B000EE8680000006401 or the three-element encoding of 0x831A000EE868186401. While message integrity and other syntax-based checks may treat these values differently, any EID-based comparisons MUST treat these values the same - as representing the ipn EID ipn:977000.100.1.

7. Special Considerations

The ipn URI scheme provides a compact and hierarchical mechanism for identifying services on network nodes. There is a significant amount of utility in the ipn URI scheme approach to identification. However, implementers should take into consideration the following observations on the use of the ipn URI scheme, particularly in regard to interoperability with implementations that pre-date this specification.

7.1. Scheme Compatibility

The ipn scheme update that has been presented in this document preserves backwards compatibility with any ipn URI scheme going back to the provisional definition of the ipn scheme in the experimental Compressed Bundle Header Encoding [RFC6260] specification in 2011. This means that any ipn URI that was valid prior to the publication of this update remains a valid ipn URI.

Similarly, the two-element scheme-specific encoding (Section 6.1.1) is also backwards-compatible with the encoding of ipn URIs provided in [RFC9171]. Any existing RFC9171-compliant implementation will produce an ipn URI encoding in compliance with this specification.

The introduction of optional non-default Allocator Identifiers and a three-element scheme-specific encoding does not make this ipn URI scheme update forwards-compatible. Existing implementations for which support of this update is desired MUST be updated to be able to process non-default Allocator Identifiers and three-element scheme-specific encodings. It is RECOMMENDED that BPv7 implementations upgrade to process these new features to benefit from the scalability provided by Allocator Identifiers and the encoding efficiencies provided by the three-element encoding.

7.2. CBOR Representation Interoperability

Care must be taken when deploying implementations that default to using the three-element encoding in networks that include implementations that only support the two-element [RFC9171] encoding. Because the existing implementations will reject bundles that use the three-element encoding as malformed, correct forwarding of semantically valid bundles will fail. The used mitigation for this issue depends on the nature of the interoperability required by the deployment. Techniques can include:

* A configuration option indicating when an implementation must use the two-element encoding for all ipn EIDs when processing bundles destined to a given endpoint: This would be suitable when adding a newer implementation to a network of existing implementations.

* Selective bundle encapsulation, whereby bundles that are known to originate from implementations that do not support the three-element encoding are tunnelled across regions of the network that require the three-element encoding: This would utilize specially configured 'gateway nodes' to perform the tunnel encapsulation and decapsulation, and would be suitable when joining an existing network to a larger network.

Techniques that do not mitigate the problem include:

* Heuristic determination of the correct encoding to use when responding to a bundle by examining the incoming bundle: It is not possible to determine whether the two-element encoding is required by the destination when composing a new bundle in response to the receipt of a bundle, such as a status report, because ipn EIDs assigned by the Default Allocator use the two-element encoding, whether the implementation supports the three-element encoding or not.

* Transcoding bundles at intermediate nodes: [RFC9171] requires the bundle primary block be immutable, and even if ipn EIDs in the primary block do not require rewriting, other blocks including the payload block may include ipn EIDs of which the transcoding node is unaware. Additionally, bundle blocks may be covered by [RFC9172] bundle security blocks or bundle integrity blocks, making them immutable.

7.3. Text Representation Compatibility

The textual representation of ipn URIs is not forwards-compatible with [RFC9171], therefore care must be taken when deploying implementations or tooling that use the textural representation of ipn URIs and support for non-default Allocator Identifiers is required. For example Section 4.6 of [RFC9174] specifies that the Session Initialization message "...SHALL contain the UTF-8 encoded node ID of the entity that sent the SESS_INIT message." In such cases the considerations that apply to the use of the 3-element CBOR encoding also apply to the text representation when a non-default Allocator Identifier is present.

7.4. Bundle Protocol Version 6 Compatibility

This document updates the use of ipn EIDs for BPv7, however the ipn URI scheme was originally defined for use with version 6 of the Bundle Protocol (BPv6). This document does not update any of the behaviors, wire-formats or mechanisms of BPv6. Therefore, ipn EIDs with non-default Allocator Identifiers MUST NOT be used with BPv6, and the Allocator Identifier prefix MUST be omitted from any textural representation. It should be noted that BPv6 has no concept of LocalNode EIDs, and will therefore treat such EIDs as routable.

7.5. Late Binding

[RFC9171] mandates the concept of "late binding" of an EID, whereby the address of the destination of a bundle is resolved from its identifier hop-by-hop as it transits a BPv7 network. This per-hop binding of identifiers to addresses underlines the fact that EIDs are purely names, and should not carry any implicit or explicit information concerning the current location or reachability of an identified node and service. This removes the need to rename a node as its location changes.

The concept of "late binding" is preserved in this ipn URI scheme. Elements of an ipn URI MUST NOT be regarded as carrying information relating to location, reachability, or other addressing/routing concern.

An example of incorrect behavior would be to assume that a given Allocator assigns Node Numbers derived from link-layer addresses and to interpret the Node Number component of an ipn URI directly as a link-layer address. No matter the mechanism an Allocator uses for the assignment of Node Numbers, they remain just numbers, without additional meaning.

8. Security Considerations

This section updates the security considerations from Section 4.2.5.1.2 of [RFC9171] to account for the inclusion of Allocator Identifiers in the ipn URI scheme when used with BPv7.

8.1. Reliability and consistency

None of the BPv7 endpoints identified by ipn EIDs are guaranteed to be reachable at any time, and the identity of the processing entities operating on those endpoints is never guaranteed by the Bundle Protocol itself. Verification of the signature provided by the Block Integrity Block targeting the bundle’s primary block, as defined by Bundle Protocol Security [RFC9172], is required for this purpose.

8.2. Malicious construction

Malicious construction of a conformant ipn URI is limited to the malicious selection of Allocator Identifiers, Node Numbers, and Service Numbers. That is, a maliciously constructed ipn EID could be used to direct a bundle to an endpoint that might be damaged by the arrival of that bundle or, alternatively, to declare a false source for a bundle and thereby cause incorrect processing at a node that receives the bundle. In both cases (and indeed in all bundle processing), the node that receives a bundle should verify its authenticity and validity before operating on it in any way, such as the use of BPSec [RFC9172], and TCPCLv4 with TLS [RFC9174].

8.3. Back-end transcoding

The limited expressiveness of URIs of the ipn scheme effectively eliminates the possibility of threat due to errors in back-end transcoding.

8.4. Local and Private Use ipn EIDs

Both LocalNode (Section 3.4.2) and Private Use (Section 3.4.3) ipn URIs present a risk to the stability of deployed BPv7 networks. If either type of ipn URI are allowed to propagate beyond the domain in which they are valid, then the required uniqueness of ipn URIs no longer holds, and this fact can be abused by a malicious node to prevent the correct functioning of the network as a whole.

See LocalNode ipn EIDs (Section 5.4) and Private Use ipn EIDs (Section 5.5) for required behaviors to mitigate against this form of abuse.

8.5. Sensitive information

Because ipn URIs are used only to represent the numeric identities of resources, the risk of disclosure of sensitive information due to interception of these URIs is minimal. Examination of ipn URIs could be used to support traffic analysis; where traffic analysis is a plausible danger, bundles should be conveyed by secure convergence-layer protocols that do not expose endpoint IDs, such as TCPCLv4 [RFC9174].

8.6. Semantic attacks

The simplicity of ipn URI scheme syntax minimizes the possibility of misinterpretation of a URI by a human user.

9. IANA Considerations

The following sections detail requests to IANA for the creation of two new registries, and the renaming of an existing registry.

IANA is requested to update the reference to the 'ipn' scheme in the "Uniform Resource Identifier (URI) Schemes" registry to this document.

IANA is requested to add the new registries, and relocate the existing registries under the "Uniform Resource Identifier (URI) Schemes" protocol registry.

9.1. 'ipn' Scheme URI Allocator Identifiers registry

IANA is requested to create a new registry entitled "'ipn' Scheme URI Allocator Identifiers". The registration policy for this registry, using terms defined in [RFC8126], is:

Range	Registration Policy
0..0xFFFF	Expert Review, Single Allocator Identifiers only
0x10000..0x3FFFFFFF	Expert Review
0x40000000..0x7FFFFFFF	Experimental Use
0x80000000..0xFFFFFFFF	Reserved, Future Expansion
>= 2^32	Reserved

Table 2: 'ipn' Scheme URI Numbering Allocator Identifiers registration policies

Each entry in this registry associates one or more Allocator Identifiers with a single organization. Within the registry, the organization is identified using the "Name" and "Point of Contact" fields. It is expected that each identified organization publishes some listing of allocated node numbers - the pointer to which is listed in the "Reference" field of the registry.

Note that the â\200\234Single Allocator Identifiers onlyâ\200\235 language in Registration Policy for this registry indicates that, within the indicated range, the allocation of a sequence of consecutive Allocator identifiers to a single organization is prohibited. IANA is requested to note this in the registration policy for this registry.

The initial values for the registry are:

Name	Range (dec)	Range (hex)	Range Length (Bits)	Reference	Point of Contact
Default Allocator (Section 3.2.2)	0	0x0	0	This document	IANA
Example Range	974848 .. 978943	0xEE000 .. 0xEEFFF	12 bits	This document	IANA

Table 3: 'ipn' Scheme URI Allocator Identifiers initial values

The "Example Range" is assigned for use in examples in documentation and sample code.

9.1.1. Guidance for Designated Experts

Due to the nature of the CBOR encoding of unsigned integers used for Allocator Identifiers with BPv7, Allocator Identifiers with a low value number are encoded more efficiently than larger numbers. This makes low value Allocator Identifiers more desirable than larger Allocator Identifiers, and therefore care must be taken when assigning Allocator Identifier ranges to ensure that a single applicant is not granted a large swathe of highly desirable numbers at the expense of other applicants. To this end, Designated Experts are strongly recommended to familiarize themselves with the CBOR encoding of unsigned integers in [RFC8949].

9.2. 'ipn' Scheme URI Default Allocator Node Numbers registry

IANA is requested to rename the "CBHE Node Numbers" registry defined in Section 3.2.1 of [RFC7116] to the "'ipn' Scheme URI Default Allocator Node Numbers" registry.

The registration policy for this registry, using terms defined in [RFC8126], is updated to be:

Range	Registration Policy
0	Reserved for the Null ipn URI (Section 3.1)
1..0x3FFF	Private Use
0x4000..0xFFFFFFFE	Expert Review
0xFFFFFFFF	Reserved for LocalNode ipn URIs (Section 3.4.2)
>= 2^32	Invalid

Table 4: 'ipn' Scheme URI Default Allocator Node Numbers registration policies

As IANA is requested to only rename the registry, all existing registrations will remain.

9.3. 'ipn' Scheme URI Well-known Service Numbers for BPv7 registry

IANA is requested to create a new registry entitled "'ipn' Scheme URI Well-known Service Numbers for BPv7" registry. The registration policy for this registry, using terms defined in [RFC8126], is:

Range	Registration Policy
0	Reserved for the Administrative Endpoint (Section 5.7)
1..127	Private Use
128..255	Standards Action
0x0100..0x7FFF	Private Use
0x8000..0xFFFF	Specification Required
0x10000..0xFFFFFFFF	Private Use
>= 2^32	Reserved for future expansion

Table 5: 'ipn' Scheme URI Well-known Service Numbers for BPv7 registration policies

The initial values for the registry are:

Value	Description	Reference
0	The Administrative Endpoint (Section 5.7)	[RFC9171], This document
0xEEE0 .. 0xEEEF	Example Range	This document

Table 6: 'ipn' Scheme URI Well-known Service Numbers for BPv7 initial values

The "Example Range" is assigned for use in examples in documentation and sample code.

9.3.1. Guidance for Designated Experts

This registry is intended to record the default Service Numbers for well-known, interoperable services available and of use to the entire BPv7 community, hence all ranges not marked for Private Use MUST have a corresponding publicly available specification describing how one interfaces with the service.

Services that are specific to a particular deployment or co-operation may require a registry to reduce administrative burden, but do not require an entry in this registry.

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/rfc/rfc2119>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/rfc/rfc5234>.

[RFC6260] Burleigh, S., "Compressed Bundle Header Encoding (CBHE)", RFC 6260, DOI 10.17487/RFC6260, May 2011, <https://www.rfc-editor.org/rfc/rfc6260>.

[RFC7116] Scott, K. and M. Blanchet, "Licklider Transmission Protocol (LTP), Compressed Bundle Header Encoding (CBHE), and Bundle Protocol IANA Registries", RFC 7116, DOI 10.17487/RFC7116, February 2014, <https://www.rfc-editor.org/rfc/rfc7116>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/rfc/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/rfc/rfc8610>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <https://www.rfc-editor.org/rfc/rfc8949>.

[RFC9171] Burleigh, S., Fall, K., and E. Birrane, III, "Bundle Protocol Version 7", RFC 9171, DOI 10.17487/RFC9171, January 2022, <https://www.rfc-editor.org/rfc/rfc9171>.

10.2. Informative References

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <https://www.rfc-editor.org/rfc/rfc1918>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/rfc/rfc3986>.

[RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", BCP 122, RFC 4632, DOI 10.17487/RFC4632, August 2006, <https://www.rfc-editor.org/rfc/rfc4632>.

[RFC5050] Scott, K. and S. Burleigh, "Bundle Protocol Specification", RFC 5050, DOI 10.17487/RFC5050, November 2007, <https://www.rfc-editor.org/rfc/rfc5050>.

[RFC9172] Birrane, III, E. and K. McKeever, "Bundle Protocol Security (BPSec)", RFC 9172, DOI 10.17487/RFC9172, January 2022, <https://www.rfc-editor.org/rfc/rfc9172>.

[RFC9174] Sipos, B., Demmer, M., Ott, J., and S. Perreault, "Delay-Tolerant Networking TCP Convergence-Layer Protocol Version 4", RFC 9174, DOI 10.17487/RFC9174, January 2022, <https://www.rfc-editor.org/rfc/rfc9174>.

Appendix A. ipn URI Scheme Text Representation Examples

This section provides some example ipn URIs in their textual representation.

A.1. Using the Default Allocator

Consider the ipn URI identifying Service Number 2 on Node Number 1 allocated by the Default Allocator (Section 3.2.2) (0).

The recommended seven character representation of this URI would be as follows:

ipn:1.2

The nine character representation of this URI, with explicit the Allocator Identifier, would be as follows:

ipn:0.1.2

A.2. Using a non-default Allocator

Consider the ipn URI identifying Service Number 3 on Node Number 1 allocated by Allocator 977000.

The 14 character representation of this URI would be as follows:

ipn:977000.1.3

A.3. The Null ipn URI

The Null ipn URI (Section 3.1) is represented as:

ipn:0.0

A.4. A LocalNode ipn URI

Consider the ipn URI identifying Service Number 7 on the local node.

The recommended seven character representation of this URI would be as follows:

ipn:!.7

The numeric 16 character representation of this URI would be as follows:

ipn:4294967295.7

Appendix B. ipn URI Scheme CBOR Encoding Examples

This section provides some example CBOR encodings of ipn EIDs.

B.1. Using the Default Allocator

Consider the ipn EID ipn:1.1. This textual representation of an ipn EID identifies Service Number 1 on Node Number 1 allocated by the Default Allocator (Section 3.2.2) (0).

The recommended five octet encoding of this EID using the two-element scheme-specific encoding would be as follows:

```
82       # 2-Element Endpoint Encoding
   02    # uri-code: 2 (IPN URI scheme)
   82    # 2 Element ipn EID scheme-specific encoding
      01 # Node Number
      01 # Service Number
```

The six octet encoding of this EID using the three-element scheme-specific encoding would be as follows:

```
82       # 2-Element Endpoint Encoding
   02    # uri-code: 2 (IPN URI scheme)
   83    # 3 Element ipn EID scheme-specific encoding
      00 # Default Allocator
      01 # Node Number
      01 # Service Number
```

B.2. Using a non-default Allocator

Consider the ipn EID ipn:977000.1.1. This textual representation of an ipn EID identifies Service Number 1 on Node Number 1 allocated by Allocator 977000.

The recommended 10 octet encoding of this EID using the three-element scheme-specific encoding would be as follows:

```
82               # 2-Element Endpoint Encoding
   02            # uri-code: 2 (IPN URI scheme)
   83            # 3 Element ipn EID scheme-specific encoding
      1A 000EE868 # Allocator Identifier
      01         # Node Number
      01         # Service Number
```

The 13 octet encoding of this EID using the two-element scheme-specific encoding would be as follows:

```
82                       # 2-Element Endpoint Encoding
   02                    # uri-code: 2 (IPN URI scheme)
   82                    # 2 Element ipn EID scheme-specific encoding
      1B 000EE86800000001 # Fully-qualified Node Number
      01                 # Service Number
```

B.3. The 'null' Endpoint

The 'null' EID of ipn:0.0 can be encoded in the following ways:

The recommended five octet encoding of the 'null' ipn EID using the two-element scheme-specific encoding would be as follows:

```
82       # 2-Element Endpoint Encoding
   02    # uri-code: 2 (IPN URI scheme)
   82    # 2 Element ipn EID scheme-specific encoding
      00 # Node Number
      00 # Service Number
```

The six octet encoding of the 'null' ipn EID using the three-element scheme-specific encoding would be as follows:

```
82       # 2-Element Endpoint Encoding
   02    # uri-code: 2 (IPN URI scheme)
   83    # 3 Element ipn EID scheme-specific encoding
      00 # Default Allocator
      00 # Node Number
      00 # Service Number
```

Acknowledgments

The following DTNWG participants contributed technical material, use cases, and critical technical review for this URI scheme update: Scott Burleigh of the IPNGROUP, Keith Scott, Brian Sipos of the Johns Hopkins University Applied Physics Laboratory, Jorge Amodio of LJCV Electronics, and Ran Atkinson.

Additionally, the authors wish to thank members of the CCSDS SIS-DTN working group at large who provided useful review and commentary on this document and its implications for the future of networked space exploration.

Authors' Addresses

Rick Taylor
Aalyria Technologies
Email: rtaylor@aalyria.com

Ed Birrane
JHU/APL
Email: Edward.Birrane@jhuapl.edu

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 29 May 2025

K. Smith
Vodafone

25 November 2024

api-catalog: a well-known URI and link relation to help discovery of APIs
draft-ietf-httpapi-api-catalog-06

Abstract

This document defines the "api-catalog" well-known URI and link relation. It is intended to facilitate automated discovery and usage of the APIs published by a given organisation or individual. A request to the api-catalog resource will return a document providing information about, and links to, the publisher's APIs.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://ietf-wg-httpapi.github.io/api-catalog/draft-ietf-httpapi-api-catalog.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-httpapi-api-catalog/.

Discussion of this document takes place on the Building Blocks for HTTP APIs Working Group mailing list (mailto:httpapi@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/httpapi/. Subscribe at https://www.ietf.org/mailman/listinfo/httpapi/.

Source for this draft and an issue tracker can be found at https://github.com/ietf-wg-httpapi/api-catalog.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

An organisation or individual may publish Application Programming Interfaces (APIs) to encourage requests for interaction from external parties. Such APIs must be discovered before they may be used - i.e., the external party needs to know what APIs a given publisher exposes, their purpose, any policies for usage, and the endpoint to interact with each API. To facilitate automated discovery of this information, and automated usage of the APIs, this document proposes:

* a well-known URI [WELL-KNOWN], 'api-catalog', encoded as a URI reference to an API catalog document describing a Publisher's API endpoints.

* a link relation [WEB-LINKING], 'api-catalog', of which the target resource is the Publisher's API Catalog document.

1.1. Goals and non-goals

The primary goal is to facilitate the automated discovery of a Publisher's public API endpoints, along with metadata that describes the purpose and usage of each API, by specifying a well-known URI that returns an API catalog document. The API catalog document is primarily machine-readable to enable automated discovery and usage of APIs, and it may also include links to human-readable documentation.

Non-goals: this document does not mandate paths for API endpoints. i.e., it does not mandate that my_example_api's endpoint should be https://www.example.com/.well-known/api-catalog/my_example_api , nor even to be hosted at www.example.com (although it is not forbidden to do so).

1.2. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

The term "content negotiation" and "status code" are from [HTTP]. The term "well-known URI" is from [WELL-KNOWN]. The term "link relation" is from [WEB-LINKING].

The term "Publisher" refers to an organisation, company or individual that publishes one or more APIs for usage by external third parties. A fictional Publisher named "example" is used throughout this document. The examples use the FQDNs "www.example.com", "developer.example.com", "apis.example.com", "apis.example.net", "gaming.example.com", "iot.example.net",where the use of the .com and .net TLDs and various subdomains are simply to illustrate that the "example" Publisher may have their API portfolio distributed across various domains for which they are the authority. For scenarios where the Publisher "example" is not the authority for a given _.example._ domain then that is made explicit in the text.

In this document, "API" means the specification resources required for an external party (or in the case of ’private’ APIs, an internal party) to implement software which uses the Publisher’s Application Programming Interface.

The specification recommends the use of TLS, hence "HTTPS" and "https://" are used throughout.

2. Using the ’api-catalog’ well-known URI

The api-catalog well-known URI is intended for HTTPS servers that publish APIs.

* The API Catalog MUST be named "api-catalog" in a well-known location as described by [WELL-KNOWN].

* The location of the API Catalog document is decided by the Publisher: the /.well-known/api-catalog URI provides a convenient reference to that location.

A Publisher supporting this URI:

* SHALL resolve an HTTPS GET request to /.well-known/api-catalog and return an API catalog document (as described in Section 4).

* SHOULD resolve an HTTPS HEAD request to /.well-known/api-catalog with a response including a Link header with the relation(s) defined in Section 3

3. The api-catalog link relation

This document introduces a new link relation [WEB-LINKING], "api-catalog". This identifies a target resource that represents a list of APIs available from the Publisher of the context resource. The target resource URI may be ./well-known/api-catalog , or any other URI chosen by the Publisher. For example, the Publisher 'example' could include the api-catalog link relation in the HTTP header and/or content payload when responding to a request to https://www.example.com :

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Location: /index.html
Link: </my_api_catalog.json>; rel=api-catalog
Content-Length: 356

<!DOCTYPE HTML>
  <html>
    <head>
      <title>Welcome to Example Publisher</title>
    </head>
    <body>
      <p>
       <a href="my_api_catalog.json" rel="api-catalog">
        Example Publisher's APIs
       </a>
      </p>
      <p>(remainder of content)</p>
    </body>
  </html>
```

3.1. Using additional link relations

* "item" [RFC6573]. When used in an API Catalog document, the "item" link relation identifies a target resource that represents an API that is a member of the API Catalog.

* Other link relations may be utilised in an API Catalog to convey metadata descriptions for API links.

4. The API Catalog document

The API Catalog is a document listing hyperlinks to a Publisher's APIs. The Publisher may host this API Catalog document at any URI(s) they choose. As illustration, the API Catalog document URI of https://www.example.com/my_api_catalog.json can be requested directly, or via a request to https://www.example.com/.well-known/api-catalog, which the Publisher will resolve to https://www.example.com/my_api_catalog.

The Publisher MUST publish the API Catalog document in the Linkset format application/linkset+json (section 4.2 of [RFC9264]). In addition, the Publisher MAY make additional formats available via content negotiation (section 5.3 of [HTTP]) to their /.well-known/api-catalog location. A non-exhaustive list of such formats that support the automated discovery, and machine (and human) usage of a Publisher's APIs, is listed below.

The API Catalog document MUST include hyperlinks to API endpoints, and is RECOMMENDED to include useful metadata, such as usage policies, API version information, links to the OpenAPI Specification [OAS] definitions for each API, etc.. If the Publisher does not include these metadata directly in the API Catalog document, they SHOULD make that metadata available at the API endpoint URIs they have listed (see Appendix A.2 for an example).

Some suitable API Catalog document formats include:

* A linkset in JSON Document format (section 4.2 of [RFC9264]) of API endpoints and information to facilitate API usage. The linkset SHOULD include a profile parameter (section 5 of [RFC9264]) with a Profile URI [RFC7284] value of 'THIS-RFC-URL' to indicate the linkset is representing an API Catalog document as defined above. Appendix A includes example API Catalog documents based on the linkset format.

* An APIs.json document [APIsjson].

* API bookmarks that represent an API entry-point URI, which may be followed to discover purpose and usage.

* A RESTDesc semantic description for hypermedia APIs [RESTdesc].

* A Hypertext Application Language document [HAL].

* An extension to the Schema.org WebAPI type [WebAPIext].

If a Publisher already lists their APIs in a format other than linkset but wish to utilise the /.well-known/api-catalog URI, then:

* They MUST also implement a linkset with, at minimum, hyperlinks to API endpoints - see the example of Appendix A.2 in Appendix A.

* They MAY support content negotiation at the /.well-known/api-catalog URI to allow their existing format to be returned.

4.1. Nesting API Catalog links

An API Catalog may itself contain links to other API Catalogs, by using the 'api-catalog' relation type for each link. An example of this is given in Appendix A.3.

5. Operational considerations

5.1. Accounting for APIs distributed across multiple domains

A Publisher ("example") may have their APIs hosted across multiple domains that they manage: e.g., at www.example.com, developer.example.com, apis.example.com, apis.example.net etc. They may also use a third-party API hosting provider which hosts APIs on a distinct domain.

To account for this scenario, it is RECOMMENDED that:

* The Publisher also publish the api-catalog well-known URI at each of their API domains e.g. https://apis.example.com/.well-known/api-catalog, https://developer.example.net/.well-known/api-catalog etc.

* An HTTPS GET request to any of these URIs returns the same result, namely, the API Catalog document.

* Since the physical location of the API Catalog document is decided by the Publisher, and may change, the Publisher choose one of their instances of /.well-known/api-catalog as a canonical reference to the location of the latest API Catalog. The Publisher's other instances of ./well-known/api-catalog SHOULD redirect to this canonical instance of /.well-known/api-catalog to ensure the latest API Catalog is returned.

For example, if the Publisher's primary API portal is https://apis.example.com, then https://apis.example.com/.well-known/api-catalog SHOULD resolve to the location of the Publisher's latest API Catalog document. If the Publisher is also the domain authority for www.example.net, which also hosts a selection of their APIs, then a request to https://www.example.net/.well-known/api-catalog SHOULD redirect to https://apis.example.com/.well-known/api-catalog .

If the Publisher is not the domain authority for www.example.net - or any third-party domain that hosts any of the Publisher's APIs - then the Publisher MAY include a link in its own API Catalog to that third-party domain's API Catalog. For example, the API Catalog available at https://apis.example.com/.well-known/api-catalog) may list APIs hosted at apis.example.com and also link to the API Catalog hosted at https://www.example.net/.well-known/api-catalog using the "api-catalog" link relation:

```
{
  "linkset": [
    {
      "anchor": "https://www.example.com/.well-known/api-catalog",
      "item": [
        {
          "href": "https://developer.example.com/apis/foo_api"
        },
        {
          "href": "https://developer.example.com/apis/bar_api"
        },
        {
          "href": "https://developer.example.com/apis/cantona_api"
        }
      ],
      "api-catalog": "https://www.example.net/./well-known/api-catalog"
    }
  ]
}
```

5.2. Internal use of api-catalog for private APIs

A Publisher may wish to use the api-catalog well-known URI on their internal network, to signpost authorised users (e.g. company employees) towards internal/private APIs not intended for third-party use. This scenario may incur additional security considerations, as noted in Section 8.

5.3. Scalability guidelines

In cases where a Publisher has a large number of APIs, potentially deployed across multiple domains, then two challenges may arise:

* Maintaining the catalog entries to ensure they are up to date and any errors corrected.

* Restricting the catalog size to help reduce network and client-processing overheads.

In both cases a Publisher may benefit from grouping their APIs, providing an API Catalog document for each group - and use the main API Catalog hosted at /.well-known/api-catalog to provide links to these. For example a Publisher may decide to group their APIs according to a business category (e.g. 'gaming APIs', 'anti-fraud APIs' etc.) or a technology category (e.g. ''IOT', 'networks', 'AI' etc.), or any other criterion. This grouping may already be implicit where the Publisher has already published their APIs across multiple domains, e.g. at gaming.example.com, iot.example.net, etc.

Section 4.1 below shows how the API Catalog at /.well-known/api-catalog can use the api-catalog link relation to point to other API Catalogs.

The Publisher SHOULD consider caching and compression techniques to reduce the network overhead of large API Catalogs.

5.4. Monitoring and maintenance

Publishers are RECOMMENDED to follow operational best practice when hosting API Catalog(s), including but not limited to:

* Health. The Publisher SHOULD monitor availability of the API Catalog, and consider alternate means to resolve requests to /.well-known/api-catalog during planned downtime of hosts.

* Performance. Although the performance of APIs listed in an API Catalog can demand high transactions per second and low-latency response, the retrieval of the API Catalog itself to discover those APIs is less likely to incur strict performance demands. That said, the Publisher SHOULD monitor the response time to fulfil a request for the API Catalog, and determine any necessary improvements (as with any other Web resource the Publisher serves). For large API Catalogs, the Publisher SHOULD consider the techniques described in Section 5.3.

* Usage. Since the goal of the api-catalog well-known URI is to facilitate discovery of APIs, the Publisher may wish to correlate requests to the /.well-known/api-catalog URI with subsequent requests to the API URIs listed in the catalog.

* Current data. The Publisher SHOULD include the removal of stale API entries from the API Catalog as part of their API release lifecycle. The Publisher MAY decide to include metadata regarding legacy API versions or deprecated APIs to help users of those APIs discover up-to-date alternatives.

* Correct metadata. The Publisher SHOULD include human and/or automated checks for syntax errors in the API Catalog. Automated checks include format validation (e.g. to ensure valid JSON syntax) and linting to enforce business rules - such as removing duplicate entries and ensuring descriptions are correctly named with valid values. A proofread of the API Catalog as part of the API release lifecycle is RECOMMENDED to detect any errors in business grammar (for example, an API entry that is described with valid syntax, but has been allocated an incorrect or outdated description.)

* Security best practice, as set out in Section 8

5.5. Integration with existing API management frameworks

A Publisher may already utilise an API management framework to produce their API portfolio. These frameworks typically include the publication of API endpoint URIs, deprecation and redirection of legacy API versions, API usage policies and documentation, etc. The api-catalog well-known URI and API Catalog document are intended to complement API management frameworks by facilitating the discovery of the framework's outputs - API endpoints, usage policies and documentation - and are not intended to replace any existing API discovery mechanisms the framework has implemented.

Providers of such frameworks may include the production of an API Catalog and the publication of the /.well-known/api-catalog URI as a final pre-release (or post-release) step in the release management workflow. The following steps are recommended:

If the ./well-known/api-catalog URI has not been published previously, the framework provider should:

* Collate and check the metadata for each API that will be included in the API Catalog. This metadata is likely to already exist in the framework.

* Determine which metadata to include in the API Catalog, following the requirements set out in Section 4 and the considerations set out in Section 5.

* Map the chosen metadata to the format(s) described in Section 4. Where only the hyperlinks to APIs are to be included in the API Catalog, then the structure suggested in Appendix A.2 may be followed. Where possible the API Catalog SHOULD include further metadata per the guidance in Section 4, in which case the structure suggested in Appendix A can be utilised and adapted (ensuring compliance to [RFC9264]) to reflect the nature of the chosen metadata.

* Publish the /.well-known/api-catalog URI following the guidance set out in Section 2.

If the ./well-known/api-catalog URI has previously been published, the framework provider should:

* Include a step in the release management lifecycle to refresh the API Catalog following any changes in API hyperlinks or published metadata. This could include placing triggers on certain metadata fields, so that as they are updated in pre-production on the API framework, the updates are pushed to a pre-production copy of the API Catalog to be pushed live when the release is published by the framework.

6. Conformance to RFC8615

The requirements in section 3 of [WELL-KNOWN] for defining Well-Known Uniform Resource Identifiers are met as described in the following sub-sections.

6.1. Path suffix

The api-catalog URI SHALL be appended to the /.well-known/ path-prefix for "well-known locations".

6.2. Formats and associated media types

A /.well-known/api-catalog location MUST support the Linkset [RFC9264] format of application/linkset+json, and MAY also support the other formats via content negotiation.

6.3. Registration of the api-catalog well-known URI

See Section 7 considerations below.

7. IANA Considerations

7.1. The api-catalog well-known URI

This specification registers the "api-catalog" well-known URI in the Well-Known URI Registry as defined by [WELL-KNOWN].

* URI suffix: api-catalog
* Change Controller: IETF
* Specification document(s): THIS-RFC
* Status: permanent

7.2. The api-catalog link relation

This specification registers the "api-catalog" link relation by following the procedures per section 2.1.1.1 of [WEB-LINKING]

* Relation Name: api-catalog
* Description: The link target identifies a catalog of the APIs published by the owner of the link target domain.
* Reference: THIS-RFC

7.3. The api-catalog Profile URI

This specification registers "THIS-RFC-URL" in the "Profile URIs" registry according to [RFC7284].

* Profile URI: THIS-RFC-URL
* Common Name: API Catalog
* Description: A profile URI to request or signal a linkset representing an API Catalog.
* Reference: THIS-RFC

RFC Editor's Note: IANA is kindly requested to replace all instances of THIS-RFC and THIS-RFC-URL with the actual RFC number/URL once assigned.

8. Security Considerations

For all scenarios:

* TLS SHOULD be used, i.e. make /.well-known/api-catalog available exclusively over HTTPS, to ensure no tampering of the API Catalog.

* The Publisher SHOULD take into account the Security Considerations from [WELL-KNOWN].

* The Publisher SHOULD perform a security and privacy review of the API Catalog prior to deployment, to ensure it does not leak personal, business or other sensitive metadata, nor expose any vulnerability related to the APIs listed.

* The Publisher SHOULD enforce read-only privileges for external requests to .well-known/api-catalog, and for internal systems and roles that monitor the .well-known/api-catalog URI. Write privileges SHOULD only be granted to roles that perform updates to the API Catalog and/or the forwarding rewrite rules for the .well-known/api-catalog URI.

* As with any Web offering, it is RECOMMENDED to apply rate-limiting measures to help mitigate abuse and prevent Denial-of-Service attacks on the API Catalog endpoint.

For the public-facing APIs scenario: security teams SHOULD additionally audit the API Catalog to ensure no APIs intended solely for internal use have been mistakenly included. For example, a catalog hosted on https://developer.example.com should not expose unnecessary metadata about any internal domains (e.g. https://internal.example.com).

For the internal/private APIs scenario: the Publisher SHOULD take steps to ensure that appropriate controls - such as CORS policies and access control lists - are in place to ensure only authorised roles and systems may access an internal api-catalog well-known URI.

9. References

9.1. Normative References

[HTTP] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[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/rfc/rfc2119>.

[RFC6573] Amundsen, M., "The Item and Collection Link Relations", RFC 6573, DOI 10.17487/RFC6573, April 2012, <https://www.rfc-editor.org/rfc/rfc6573>.

[RFC7284] Lanthaler, M., "The Profile URI Registry", RFC 7284, DOI 10.17487/RFC7284, June 2014, <https://www.rfc-editor.org/rfc/rfc7284>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC9264] Wilde, E. and H. Van de Sompel, "Linkset: Media Types and a Link Relation Type for Link Sets", RFC 9264, DOI 10.17487/RFC9264, July 2022, <https://www.rfc-editor.org/rfc/rfc9264>.

[WEB-LINKING] Nottingham, M., "Web Linking", RFC 8288, DOI 10.17487/RFC8288, October 2017, <https://www.rfc-editor.org/rfc/rfc8288>.

[WELL-KNOWN] Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", RFC 8615, DOI 10.17487/RFC8615, May 2019, <https://www.rfc-editor.org/rfc/rfc8615>.

9.2. Informative References

[APIsjson] Kin Lane and Steve Willmott, "APIs.json", 15 September 2020, <http://apisjson.org/format/apisjson_0.16.txt>.

[HAL] Mike Kelly, "JSON Hypertext Application Language", 15 September 2020, <https://datatracker.ietf.org/doc/html/draft-kelly-json-hal-11>.

[OAS] Darrel Miller, Jeremy Whitlock, Marsh Gardiner, Mike Ralphson, Ron Ratovsky, and Uri Sarid, "OpenAPI Specification 3.1.0", 15 February 2021, <https://spec.openapis.org/oas/latest>.

[RESTdesc] Ruben Verborgh, Erik Mannens, Rick Van de Walle, and Thomas Steiner, "RESTdesc", 15 September 2023, <http://apisjson.org/format/apisjson_0.16.txt>.

[RFC8631] Wilde, E., "Link Relation Types for Web Services", RFC 8631, DOI 10.17487/RFC8631, July 2019, <https://www.rfc-editor.org/rfc/rfc8631>.

[WebAPIext] Mike Ralphson and Nick Evans, "WebAPI type extension", 8 July 2020, <https://webapi-discovery.github.io/rfcs/rfc0001.html>.

Appendix A. Example API Catalog documents

This section is informative and provides and example of an API Catalog document using the RECOMMENDED linkset format.

A.1. Using Linkset with RFC8615 relations

This example uses the linkset format [RFC9264], and the following link relations defined in [RFC8631]:

* "service-desc", used to link to a description of the API that is primarily intended for machine consumption.

* "service-doc", used to link to API documentation that is primarily intended for human consumption.

* "service-meta", used to link to additional metadata about the API, and is primarily intended for machine consumption.

* "status", used to link to the API status (e.g. API "health" indication etc.) for machine and/or human consumption.

Client request:

```
GET .well-known/api-catalog HTTP/1.1
Host: example.com
Accept: application/linkset+json
```

Server response:

```
HTTP/1.1 200 OK
Date: Mon, 01 Jun 2023 00:00:01 GMT
Server: Apache-Coyote/1.1
Content-Type: application/linkset+json;
    profile="THIS-RFC-URL"
```

```
{
  "linkset": [
  {
    "anchor": "https://developer.example.com/apis/foo_api",
    "service-desc": [
      {
        "href": "https://developer.example.com/apis/foo_api/spec",
        "type": "application/yaml"
      }
    ],
    "status": [
      {
        "href": "https://developer.example.com/apis/foo_api/status",
        "type": "application/json"
      }
    ],
    "service-doc": [
      {
        "href": "https://developer.example.com/apis/foo_api/doc",
        "type": "text/html"
      }
    ],
    "service-meta": [
      {
        "href": "https://developer.example.com/apis/foo_api/policies",
        "type": "text/xml"
      }
    ]
  },
  {
    "anchor": "https://developer.example.com/apis/bar_api",
    "service-desc": [
      {
        "href": "https://developer.example.com/apis/bar_api/spec",
        "type": "application/yaml"
      }
    ],
    "status": [
      {
        "href": "https://developer.example.com/apis/bar_api/status",
       "type": "application/json"
      }
    ],
    "service-doc": [
      {
        "href": "https://developer.example.com/apis/bar_api/doc",
        "type": "text/plain"
      }
```

```
      ]
    },
    {
      "anchor": "https://apis.example.net/apis/cantona_api",
      "service-desc": [
        {
          "href": "https://apis.example.net/apis/cantona_api/spec",
          "type": "text/n3"
        }
      ],
      "service-doc": [
        {
          "href": "https://apis.example.net/apis/cantona_api/doc",
          "type": "text/html"
        }
      ]
    }
    ]
  }
```

A.2. Using Linkset with bookmarks

This example also uses the linkset format [RFC9264], listing the API endpoints in an array of bookmarks. Each link shares the same context anchor (the well-known URI of the API Catalog) and "item" [RFC9264] link relation (to indicate they are an item in the catalog). The intent is that by following a bookmark link, a machine-client can discover the purpose and usage policy for each API, hence the document targeted by the bookmark link should support this.

Client request:

```
GET .well-known/api-catalog HTTP/1.1
Host: example.com
Accept: application/linkset+json
```

Server response:

```
HTTP/1.1 200 OK
Date: Mon, 01 Jun 2023 00:00:01 GMT
Server: Apache-Coyote/1.1
Content-Type: application/linkset+json;
    profile="THIS-RFC-URL"
```

```
{ "linkset":
 [
   { "anchor": "https://www.example.com/.well-known/api-catalog",
     "item": [
       {"href": "https://developer.example.com/apis/foo_api"},
       {"href": "https://developer.example.com/apis/bar_api"},
       {"href": "https://developer.example.com/apis/cantona_api"}
     ]
   }
 ]
}
```

A.3. Nesting API Catalog links

In this example, a request to the /.well-known/api-catalog URI returns an array of links of relation type 'api-catalog'. This can be useful to Publishers with a large number of APIs, who wish to group them in smaller catalogs (as described in Section 5.3).

Client request:

```
GET .well-known/api-catalog HTTP/1.1
Host: example.com
Accept: application/linkset+json
```

Server response:

```
HTTP/1.1 200 OK
Date: Mon, 01 Jun 2023 00:00:01 GMT
Server: Apache-Coyote/1.1
Content-Type: application/linkset+json;
    profile="THIS-RFC-URL"
```

```
{
  "linkset": [
    {
      "anchor": "https://www.example.com/.well-known/api-catalog",
      "api-catalog": [
        {
          "href": "https://apis.example.com/iot/api-catalog"
        },
        {
          "href": "https://ecommerce.example.com/api-catalog"
        },
        {
          "href": "https://developer.example.com/gaming/api-catalog"
        }
      ]
    }
  ]
}
```

Appendix B. Acknowledgements

Thanks to Jan Algermissen, Phil Archer, Tim Bray, Ben Bucksch, Sanjay Dalal, David Dong, Mallory Knodel, Max Maton, Darrel Miller, Mark Nottingham, Roberto Polli, Joey Salazar, Rich Salz, Herbert Van De Sompel, Tina Tsou and Erik Wilde for their reviews, suggestions and support.

Author's Address

Kevin Smith
Vodafone
Email: kevin.smith@vodafone.com
URI: https://www.vodafone.com

LAMPS WG
Internet-Draft
Intended status: Standards Track
Expires: 8 May 2025

R. Mahy
Rohan Mahy Consulting Services

4 November 2024

X.509 Certificate Extended Key Usage (EKU) for Instant Messaging URIs
draft-ietf-lamps-im-keyusage-03

Abstract

RFC 5280 specifies several extended key purpose identifiers (KeyPurposeIds) for X.509 certificates. This document defines Instant Messaging (IM) identity KeyPurposeId for inclusion in the Extended Key Usage (EKU) extension of X.509 v3 public key certificates

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://rohanmahy.github.io/mahy-lamps-im-keyusage/draft-ietf-lamps-im-keyusage.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-lamps-im-keyusage/.

Discussion of this document takes place on the LAMPS WG Working Group mailing list (mailto:lamps@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/lamps/. Subscribe at https://www.ietf.org/mailman/listinfo/lamps/.

Source for this draft and an issue tracker can be found at https://github.com/rohanmahy/mahy-lamps-im-keyusage.

Status of This Memo

This Internet-Draft will expire on 8 May 2025.

Table of Contents

1. Introduction

Instant Messaging (IM) systems using the Messaging Layer Security (MLS) [RFC9420] protocol can incorporate per-client identity certificate credentials. The subjectAltName of these certificates can be an IM URI or XMPP URI, for example.

Organizations may be unwilling to issue certificates for Instant Message client using a general KeyPurposeId such as id-kp-serverAuth or id-kp-clientAuth, because of the risk that such certificates could be abused in a cross-protocol attack.

An explanation of MLS credentials as they apply to Instant Messaging is described in [I-D.barnes-mimi-identity-arch]. These credentials are expected to be heavily used in the More Instant Messaging Interoperability (MIMI) Working Group.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. The IM URI Extended Key Usage

This specification defines the KeyPurposeId id-kp-imUri, which MAY be included in certificates used to prove the identity of an Instant Messaging client. This Extended Key Usage is optionally critical.

```
id-kp  OBJECT IDENTIFIER  ::= {
  iso(1) identified-organization(3) dod(6) internet(1)
  security(5) mechanisms(5) pkix(7) kp(3) }

id-kp-imUri OBJECT IDENTIFIER ::= { id-kp TBD1 }
```

4. Security Considerations

The Security Considerations of [RFC5280] are applicable to this document. This extended key purpose does not introduce new security risks but instead reduces existing security risks by providing means to identify if the certificate is generated to sign IM identity credentials.

5. IANA Considerations

IANA is requested to register the following OIDs in the "SMI Security for PKIX Extended Key Purpose" registry (1.3.6.1.5.5.7.3). These OIDs are defined in Section 4.

Decimal	Description	References
TBD1	id-kp-imUri	This-RFC

Table 1

IANA is also requested to register the following ASN.1 [ITU.X690.2021] module OID in the "SMI Security for PKIX Module Identifier" registry (1.3.6.1.5.5.7.0). This OID is defined in Appendix A.

Decimal	Description	References
TBD2	id-mod-im-eku	This-RFC

Table 2

6. References

6.1. Normative References

[ITU.X680.2021]
International Telecommunications Union, "Information Technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, 2021.

[ITU.X690.2021]
International Telecommunications Union, "Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, 2021.

[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/rfc/rfc2119>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

6.2. Informative References

[I-D.barnes-mimi-identity-arch]
Barnes, R. and R. Mahy, "Identity for E2E-Secure Communications", Work in Progress, Internet-Draft, draft-barnes-mimi-identity-arch-01, 23 October 2023, <https://datatracker.ietf.org/doc/html/draft-barnes-mimi-identity-arch-01>.

[RFC9420] Barnes, R., Beurdouche, B., Robert, R., Millican, J., Omara, E., and K. Cohn-Gordon, "The Messaging Layer Security (MLS) Protocol", RFC 9420, DOI 10.17487/RFC9420, July 2023, <https://www.rfc-editor.org/rfc/rfc9420>.

Appendix A. ASN.1 Module

The following module adheres to ASN.1 specifications [ITU.X680.2021] and [ITU.X690.2021].

<CODE BEGINS>

```
IM-EKU
  { iso(1) identified-organization(3) dod(6) internet(1)
  security(5) mechanisms(5) pkix(7) id-mod(0)
  id-mod-im-eku (TBD2) }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- OID Arc

id-kp OBJECT IDENTIFIER ::=
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) kp(3) }

-- Extended Key Usage Values

id-kp-imUri OBJECT IDENTIFIER ::= { id-kp TBD1 }

END
```

<CODE ENDS>

Appendix B. Change log

RFC Editor, please remove this section on publication.

* made Proposed Standard
* added a MAY statement in Section 3
* corrected typo in registration of the ASN.1 module (Thanks Sean!)
* updated author affiliation
* added ASN.1 module

* specified that eku is optionally critical

Acknowledgments

Thanks to Sean Turner and Russ Housley for reviews, suggestions, corrections, and encouragement.

Author's Address

Rohan Mahy
Rohan Mahy Consulting Services
Email: rohan.ietf@gmail.com

LAMPS - Limited Additional Mechanisms for PKIX and SMIME
Internet-Draft
Intended status: Standards Track
Expires: 19 May 2025

D. Van Geest
CryptoNext Security
K. Bashiri
BSI
S. Fluhrer
Cisco Systems
S. Gazdag
genua GmbH
S. Kousidis
BSI

15 November 2024

Use of the HSS and XMSS Hash-Based Signature Algorithms in Internet X.509 Public Key Infrastructure
draft-ietf-lamps-x509-shbs-11

Abstract

This document specifies algorithm identifiers and ASN.1 encoding formats for the stateful hash-based signature (HBS) schemes Hierarchical Signature System (HSS), eXtended Merkle Signature Scheme (XMSS), and XMSS^MT, a multi-tree variant of XMSS. This specification applies to the Internet X.509 Public Key infrastructure (PKI) when those digital signatures are used in Internet X.509 certificates and certificate revocation lists.

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-lamps-x509-shbs/.

Discussion of this document takes place on the LAMPS Working Group mailing list (mailto:spasm@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/spasm/. Subscribe at https://www.ietf.org/mailman/listinfo/spasm/.

Source for this draft and an issue tracker can be found at https://github.com/x509-hbs/draft-x509-shbs.

Status of This Memo

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

This Internet-Draft will expire on 19 May 2025.

Table of Contents

1. Introduction

Stateful HBS schemes such as HSS, XMSS and XMSS^MT combine Merkle trees with One Time Signatures (OTS) in order to provide digital signature schemes that remain secure even when quantum computers become available. Their theoretic security is well understood and depends only on the security of the underlying hash function. As such they can serve as an important building block for quantum computer resistant information and communication technology.

A stateful HBS private key is a finite collection of OTS keys, hence only a limited number of messages can be signed and the private key's state must be updated and persisted after signing to prevent reuse of OTS keys. While the right selection of algorithm parameters would allow a private key to sign a virtually unbounded number of messages (e.g. 2^60), this is at the cost of a larger signature size and longer signing time. Due to the statefulness of the private key and the limited number of signatures that can be created, stateful HBS schemes might not be appropriate for use in interactive protocols. However, in some use cases the deployment of stateful HBS schemes may be appropriate. Such use cases are described and discussed in Section 3.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Use Cases of Stateful HBS Schemes in X.509

As described in the Security Considerations of Section 10, it is imperative that stateful HBS implementations do not reuse OTS signatures. This makes stateful HBS algorithms inappropriate for general use cases. The exact conditions under which stateful HBS certificates may be used is left to certificate policies [RFC3647]. However the intended use of stateful HBS schemes as described by [SP800208] can be used as a guideline:

> 1) it is necessary to implement a digital signature scheme in the near future;
> 2) the implementation will have a long lifetime; and
> 3) it would not be practical to transition to a different digital signature scheme once the implementation has been deployed.

In addition, since a stateful HBS private key can only generate a finite number of signatures, use cases for stateful HBS public keys in certificates should have a predictable range of the number of signatures that will be generated, falling safely below the maximum number of signatures that a private key can generate.

Use cases where stateful HBS public keys in certificates may be appropriate due to the relatively small number of signatures generated and the signer's ability to enforce security restrictions on the signing environment include:

* Firmware signing (Section 1.1 of [SP800208], Table IV of [CNSA2.0], Section 6.7 of [BSI])

* Software signing (Table IV of [CNSA2.0], [ANSSI])

* Certification Authority (CA) certificates.

In each of these cases, the operator is able to control their signing environment such that signatures are generated in hardware cryptographic modules and audited before the signature is published, in order to prevent OTS key reuse.

Generally speaking, stateful HBS public keys are not appropriate for use in end-entity certificates, however in the firmware and software signing cases signature generation will often be more tightly controlled. Some manufactures use common and well-established key formats like X.509 for their code signing and update mechanisms. Also there are multi-party IoT ecosystems where publicly trusted code signing certificates are useful.

In general, root CAs [RFC4949] generate signatures in a more secure environment and issue fewer certificates than subordinate CAs [RFC4949]. This makes the use of stateful HBS public keys more appropriate in root CA certificates than in subordinate CA certificates. However, if a subordinate CA can match the security and signature count restrictions of a root CA, for example if the subordinate CA only issues code-signing certificates, then using a stateful HBS public key in the subordinate CA certificate may be possible.

4. Algorithm Identifiers and Parameters

In this document, we define new OIDs for identifying the different stateful hash-based signature algorithms. An additional OID is defined in [I-D.ietf-lamps-rfc8708bis] and repeated here for convenience.

The AlgorithmIdentifier type is defined in [RFC5912] as follows:

```
AlgorithmIdentifier{ALGORITHM-TYPE, ALGORITHM-TYPE:AlgorithmSet} ::=
        SEQUENCE {
            algorithm   ALGORITHM-TYPE.&id({AlgorithmSet}),
            parameters  ALGORITHM-TYPE.
                    &Params({AlgorithmSet}{@algorithm}) OPTIONAL
        }
```

> NOTE: The above syntax is from [RFC5912] and is compatible with the 2021 ASN.1 syntax [X680]. See [RFC5280] for the 1988 ASN.1 syntax.

The fields in AlgorithmIdentifier have the following meanings:

* algorithm identifies the cryptographic algorithm with an object identifier.

* parameters, which are optional, are the associated parameters for the algorithm identifier in the algorithm field.

The parameters field of the AlgorithmIdentifier for HSS, XMSS, and XMSS^MT public keys MUST be absent.

4.1. HSS Algorithm Identifier

The object identifier and public key algorithm identifier for HSS is defined in [I-D.ietf-lamps-rfc8708bis]. The definitions are repeated here for reference.

The AlgorithmIdentifier for an HSS public key MUST use the id-alg-hss-lms-hashsig object identifier.

```
id-alg-hss-lms-hashsig  OBJECT IDENTIFIER ::= {
   iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
   smime(16) alg(3) 17 }
```

Note that the id-alg-hss-lms-hashsig algorithm identifier is also referred to as id-alg-mts-hashsig. This synonym is based on the terminology used in an early draft of the document that became [RFC8554].

The public key and signature values identify the hash function and the height used in the HSS/LMS tree. [RFC8554] and [SP800208] define these values, but an IANA registry [IANA-LMS] permits the registration of additional identifiers in the future.

4.2. XMSS Algorithm Identifier

The AlgorithmIdentifier for an XMSS public key MUST use the id-alg-xmss-hashsig object identifier.

```
id-alg-xmss-hashsig  OBJECT IDENTIFIER ::= {
   iso(1) identified-organization(3) dod(6) internet(1)
   security(5) mechanisms(5) pkix(7) algorithms(6) 34 }
```

The public key and signature values identify the hash function and the height used in the XMSS tree. [RFC8391] and [SP800208] define these values, but an IANA registry [IANA-XMSS] permits the registration of additional identifiers in the future.

4.3. XMSS^MT Algorithm Identifier

The AlgorithmIdentifier for an XMSS^MT public key MUST use the id-alg-xmssmt-hashsig object identifier.

```
id-alg-xmssmt-hashsig  OBJECT IDENTIFIER ::= {
   iso(1) identified-organization(3) dod(6) internet(1)
   security(5) mechanisms(5) pkix(7) algorithms(6) 35 }
```

The public key and signature values identify the hash function and the height used in the XMSS^MT tree. [RFC8391] and [SP800208] define these values, but an IANA registry [IANA-XMSS] permits the registration of additional identifiers in the future.

5. Public Key Identifiers

Certificates conforming to [RFC5280] can convey a public key for any public key algorithm. The certificate indicates the algorithm through an algorithm identifier. An algorithm identifier consists of an OID and optional parameters.

[RFC8554] defines the encoding of HSS public keys and [RFC8391] defines the encodings of XMSS and XMSS^MT public keys. When used in a SubjectPublicKeyInfo type, the subjectPublicKey BIT STRING contains these encodings of the public key.

This document defines ASN.1 [X680] OCTET STRING types for encoding the public keys when not used in a SubjectPublicKeyInfo. The OCTET STRING is mapped to a subjectPublicKey (a value of type BIT STRING) as follows: the most significant bit of the OCTET STRING value becomes the most significant bit of the BIT STRING value, and so on; the least significant bit of the OCTET STRING becomes the least significant bit of the BIT STRING.

5.1. HSS Public Keys

The HSS public key identifier is as follows:

```
pk-HSS-LMS-HashSig PUBLIC-KEY ::= {
   IDENTIFIER id-alg-hss-lms-hashsig
   -- KEY no ASN.1 wrapping --
   PARAMS ARE absent
   CERT-KEY-USAGE
      { digitalSignature, nonRepudiation, keyCertSign, cRLSign } }
```

The HSS public key is defined as follows:

```
HSS-LMS-HashSig-PublicKey ::= OCTET STRING
```

[RFC8554] defines the encoding of an HSS public key using the hss_public_key structure. See [SP800208] and [RFC8554] for more information on the contents and format of an HSS public key. Note that the single-tree signature scheme LMS is instantiated as HSS with number of levels being equal to 1.

5.2. XMSS Public Keys

The XMSS public key identifier is as follows:

```
pk-XMSS-HashSig PUBLIC-KEY ::= {
   IDENTIFIER id-alg-xmss-hashsig
   -- KEY no ASN.1 wrapping --
   PARAMS ARE absent
   CERT-KEY-USAGE
      { digitalSignature, nonRepudiation, keyCertSign, cRLSign } }
```

The XMSS public key is defined as follows:

```
XMSS-HashSig-PublicKey ::= OCTET STRING
```

[RFC8391] defines the encoding of an XMSS public key using the xmss_public_key structure. See [SP800208] and [RFC8391] for more information on the contents and format of an XMSS public key.

5.3. XMSS^MT Public Keys

The XMSS^MT public key identifier is as follows:

```
pk-XMSSMT-HashSig PUBLIC-KEY ::= {
   IDENTIFIER id-alg-xmssmt-hashsig
   -- KEY no ASN.1 wrapping --
   PARAMS ARE absent
   CERT-KEY-USAGE
      { digitalSignature, nonRepudiation, keyCertSign, cRLSign } }
```

The XMSS^MT public key is defined as follows:

```
XMSSMT-HashSig-PublicKey ::= OCTET STRING
```

[RFC8391] defines the encoding of an XMSS^MT public key using the xmssmt_public_key structure. See [SP800208] and [RFC8391] for more information on the contents and format of an XMSS^MT public key.

6. Key Usage Bits

The intended application for the key is indicated in the keyUsage certificate extension [RFC5280]. When id-alg-hss-lms-hashsig, id-alg-xmss-hashsig or id-alg-xmssmt-hashsig appears in the SubjectPublicKeyInfo field of a CA X.509 certificate [RFC5280], the certificate key usage extension MUST contain at least one of the following values: digitalSignature, nonRepudiation, keyCertSign, or cRLSign. However, it MUST NOT contain other values.

When id-alg-hss-lms-hashsig, id-alg-xmss-hashsig or id-alg-xmssmt-hashsig appears in the SubjectPublicKeyInfo field of an end entity X.509 certificate [RFC5280], the certificate key usage extension MUST contain at least one of the following values: digitalSignature, nonRepudiation or cRLSign. However, it MUST NOT contain other values.

7. Signature Algorithms

The same OIDs used to identify HSS, XMSS, and XMSS^MT public keys are also used to identify their respective signatures. When these algorithm identifiers appear in the algorithm field of an AlgorithmIdentifier, the encoding MUST omit the parameters field. That is, the AlgorithmIdentifier SHALL be a SEQUENCE of one component, one of the OIDs defined in the following subsections.

When the signature algorithm identifiers described in this document are used to create a signature on a message, no digest algorithm is applied to the message before signing. That is, the full data to be signed is signed rather than a digest of the data.

The format of an HSS signature is described in Section 6.2 of [RFC8554]. The format of an XMSS signature is described in Appendix B.2 of [RFC8391] and the format of an XMSS^MT signature is described in Appendix C.2 of [RFC8391]. The octet string representing the signature is encoded directly in a BIT STRING without adding any additional ASN.1 wrapping. For the Certificate and CertificateList structures, the octet string is encoded in the "signatureValue" BIT STRING field.

7.1. HSS Signature Algorithm

The id-alg-hss-lms-hashsig OID is used to specify that an HSS signature was generated on the full message, i.e. the message was not hashed before being processed by the HSS signature algorithm.

See [SP800208] and [RFC8554] for more information on the contents and format of an HSS signature.

7.2. XMSS Signature Algorithm

The id-alg-xmss-hashsig OID is used to specify that an XMSS signature was generated on the full message, i.e. the message was not hashed before being processed by the XMSS signature algorithm.

See [SP800208] and [RFC8391] for more information on the contents and format of an XMSS signature.

The signature generation MUST be performed according to 7.2 of [SP800208].

7.3. XMSS^MT Signature Algorithm

The id-alg-xmssmt-hashsig OID is used to specify that an XMSS^MT signature was generated on the full message, i.e. the message was not hashed before being processed by the XMSS^MT signature algorithm.

See [SP800208] and [RFC8391] for more information on the contents and format of an XMSS^MT signature.

The signature generation MUST be performed according to 7.2 of [SP800208].

8. Key Generation

The key generation for XMSS and XMSS^MT MUST be performed according to 7.2 of [SP800208]

9. ASN.1 Module

For reference purposes, the ASN.1 syntax is presented as an ASN.1 module here [X680]. Note that as per [RFC5280], certificates use the Distinguished Encoding Rules; see [X690]. This ASN.1 Module builds upon the conventions established in [RFC5911]. This module imports objects from [RFC5911] and [I-D.ietf-lamps-rfc8708bis].

RFC EDITOR: Please replace [I-D.ietf-lamps-rfc8708bis] in the module with a reference to the published RFC.

```
X509-SHBS-2024
  { iso(1) identified-organization(3) dod(6) internet(1) security(5)
    mechanisms(5) pkix(7) id-mod(0) id-mod-pkix1-shbs-2024(TBD) }

DEFINITIONS IMPLICIT TAGS ::= BEGIN

EXPORTS ALL;

IMPORTS
  PUBLIC-KEY, SIGNATURE-ALGORITHM
    FROM AlgorithmInformation-2009  -- [RFC5911]
      { iso(1) identified-organization(3) dod(6) internet(1)
        security(5) mechanisms(5) pkix(7) id-mod(0)
        id-mod-algorithmInformation-02(58) }

  sa-HSS-LMS-HashSig, pk-HSS-LMS-HashSig
    FROM MTS-HashSig-2013 -- [I-D.ietf-lamps-rfc8708bis]
```

```
        { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
          id-smime(16) id-mod(0) id-mod-mts-hashsig-2013(64) };

   --
   -- Object Identifiers
   --

   -- id-alg-hss-lms-hashsig is defined in [I-D.ietf-lamps-rfc8708bis]

   id-alg-xmss-hashsig  OBJECT IDENTIFIER ::= {
      iso(1) identified-organization(3) dod(6) internet(1) security(5)
      mechanisms(5) pkix(7) algorithms(6) 34 }

   id-alg-xmssmt-hashsig  OBJECT IDENTIFIER ::= {
      iso(1) identified-organization(3) dod(6) internet(1) security(5)
      mechanisms(5) pkix(7) algorithms(6) 35 }

   --
   -- Signature Algorithms and Public Keys
   --

   -- sa-HSS-LMS-HashSig is defined in [I-D.ietf-lamps-rfc8708bis]

   sa-XMSS-HashSig SIGNATURE-ALGORITHM ::= {
      IDENTIFIER id-alg-xmss-hashsig
      PARAMS ARE absent
      PUBLIC-KEYS { pk-XMSS-HashSig }
      SMIME-CAPS { IDENTIFIED BY id-alg-xmss-hashsig } }

   sa-XMSSMT-HashSig SIGNATURE-ALGORITHM ::= {
      IDENTIFIER id-alg-xmssmt-hashsig
      PARAMS ARE absent
      PUBLIC-KEYS { pk-XMSSMT-HashSig }
      SMIME-CAPS { IDENTIFIED BY id-alg-xmssmt-hashsig } }

   -- pk-HSS-LMS-HashSig is defined in [I-D.ietf-lamps-rfc8708bis]

   pk-XMSS-HashSig PUBLIC-KEY ::= {
      IDENTIFIER id-alg-xmss-hashsig
      -- KEY no ASN.1 wrapping --
      PARAMS ARE absent
      CERT-KEY-USAGE
         { digitalSignature, nonRepudiation, keyCertSign, cRLSign } }

   XMSS-HashSig-PublicKey ::= OCTET STRING

   pk-XMSSMT-HashSig PUBLIC-KEY ::= {
      IDENTIFIER id-alg-xmssmt-hashsig
```

```
      -- KEY no ASN.1 wrapping --
      PARAMS ARE absent
      CERT-KEY-USAGE
         { digitalSignature, nonRepudiation, keyCertSign, cRLSign } }

   XMSSMT-HashSig-PublicKey ::= OCTET STRING

   --
   -- Public Key (pk-) Algorithms
   --
   PublicKeys PUBLIC-KEY ::= {
      -- This expands PublicKeys from RFC 5912
      pk-HSS-LMS-HashSig |
      pk-XMSS-HashSig |
      pk-XMSSMT-HashSig,
      ...
   }

   --
   -- Signature Algorithms (sa-)
   --
   SignatureAlgs SIGNATURE-ALGORITHM ::= {
      -- This expands SignatureAlgorithms from RFC 5912
      sa-HSS-LMS-HashSig |
      sa-XMSS-HashSig |
      sa-XMSSMT-HashSig,
      ...
   }

   END
```

10. Security Considerations

The security requirements of [SP800208] MUST be taken into account.

As stateful HBS private keys can only generate a limited number of signatures, a user needs to be aware of the total number of signatures they intend to generate in their use case, otherwise they risk exhausting the number of OTS keys in their private key.

For stateful HBS schemes, it is crucial to stress the importance of correct state management. If an attacker were able to obtain signatures for two different messages created using the same OTS key, then it would become computationally feasible for that attacker to create forgeries [BH16]. As noted in [MCGREW] and [ETSI-TR-103-692], extreme care needs to be taken in order to avoid the risk that an OTS key will be reused accidentally. This is a new requirement that most developers will not be familiar with and requires careful handling.

Various strategies for a correct state management can be applied:

* Implement a record of all signatures generated by a key pair associated with a stateful HBS instance, for example by logging the OTS key indexes as signatures are generated. This record may be stored outside the device which is used to generate the signature. Check the record to prevent OTS key reuse before a new signature is released. If OTS key reuse is detected, freeze all new signature generation by the private key, re-audit previously released signatures (possibly revoking the private key if previously released signatures showed OTS key reuse), and perform a post-failure audit.

* Use a stateful HBS instance only for a moderate number of signatures such that it is always practical to keep a consistent record and be able to unambiguously trace back all generated signatures.

* Apply the state reservation strategy described in Section 5 of [MCGREW], where upcoming states are reserved in advance by the signer. In this way the number of state synchronisations between nonvolatile and volatile memory is reduced.

11. Backup and Restore Management

Certificate Authorities have high demands in order to ensure the availability of signature generation throughout the validity period of signing key pairs.

Usual backup and restore strategies when using a stateless signature scheme (e.g. SLH-DSA) are to duplicate private keying material and to operate redundant signing devices or to store and safeguard a copy of the private keying material such that it can be used to set up a new signing device in case of technical difficulties.

For stateful HBS schemes, such straightforward backup and restore strategies will lead to OTS reuse with high probability as a correct state management is not guaranteed. Strategies for maintaining availability and keeping a correct state are described in Section 7 of [SP800208].

12. IANA Considerations

One object identifier for the ASN.1 module in Section 9 is requested for the SMI Security for PKIX Module Identifiers (1.3.6.1.5.5.7.0) registry:

Decimal	Description	References
TBD	id-mod-pkix1-shbs-2024	[EDNOTE: THIS RFC]

Table 1

IANA has updated the "SMI Security for PKIX Algorithms" (1.3.6.1.5.5.7.6) registry [SMI-PKIX] with two additional entries:

Decimal	Description	References
34	id-alg-xmss-hashsig	[EDNOTE: THIS RFC]
35	id-alg-xmssmt-hashsig	[EDNOTE: THIS RFC]

Table 2

13. References

13.1. Normative References

[I-D.ietf-lamps-rfc8708bis]
Housley, R., "Use of the HSS/LMS Hash-Based Signature Algorithm in the Cryptographic Message Syntax (CMS)", Work in Progress, Internet-Draft, draft-ietf-lamps-rfc8708bis-03, 19 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-rfc8708bis-03>.

[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/rfc/rfc2119>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC5911] Hoffman, P. and J. Schaad, "New ASN.1 Modules for Cryptographic Message Syntax (CMS) and S/MIME", RFC 5911, DOI 10.17487/RFC5911, June 2010, <https://www.rfc-editor.org/rfc/rfc5911>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8391] Huelsing, A., Butin, D., Gazdag, S., Rijneveld, J., and A. Mohaisen, "XMSS: eXtended Merkle Signature Scheme", RFC 8391, DOI 10.17487/RFC8391, May 2018, <https://www.rfc-editor.org/rfc/rfc8391>.

[RFC8554] McGrew, D., Curcio, M., and S. Fluhrer, "Leighton-Micali Hash-Based Signatures", RFC 8554, DOI 10.17487/RFC8554, April 2019, <https://www.rfc-editor.org/rfc/rfc8554>.

[SP800208] National Institute of Standards and Technology (NIST), "Recommendation for Stateful Hash-Based Signature Schemes", 29 October 2020, <https://doi.org/10.6028/NIST.SP.800-208>.

[X680] ITU-T, "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, ISO/IEC 8824-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.680>.

[X690] ITU-T, "Information technology - Abstract Syntax Notation One (ASN.1): ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, ISO/IEC 8825-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.690>.

13.2. Informative References

[ANSSI] Agence nationale de la sÃ©curitÃ© des systÃ¨mes d'information (ANSSI), "ANSSI views on the Post-Quantum Cryptography transition (2023 follow up)", 21 December 2023, <https://cyber.gouv.fr/sites/default/files/document/follow_up_position_paper_on_post_quantum_cryptography.pdf>.

[BH16] Bruinderink, L. and S. HÃ¼lsing, "Oops, I did it again â\200\223 Security of One-Time Signatures under Two-Message Attacks.", 2016, <https://eprint.iacr.org/2016/1042.pdf>.

[BSI] Bundesamt fÃ¼r Sicherheit in der Informationstechnik (BSI), "Quantum-safe cryptography â\200\223 fundamentals, current developments and recommendations", 18 May 2022, <https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Brochure/quantum-safe-cryptography.pdf>.

[CNSA2.0] National Security Agency (NSA), "Commercial National Security Algorithm Suite 2.0 (CNSA 2.0) Cybersecurity Advisory (CSA)", 7 September 2022, <https://media.defense.gov/2022/Sep/07/2003071834/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS_.PDF>.

[ETSI-TR-103-692] European Telecommunications Standards Institute (ETSI), "State management for stateful authentication mechanisms", November 2021, <https://www.etsi.org/deliver/etsi_tr/103600_103699/103692/01.01.01_60/tr_103692v010101p.pdf>.

[IANA-LMS] IANA, "Leighton-Micali Signatures (LMS)", n.d., <https://www.iana.org/assignments/leighton-micali-signatures/>.

[IANA-XMSS] IANA, "XMSS: Extended Hash-Based Signatures", n.d., <https://iana.org/assignments/xmss-extended-hash-based-signatures/>.

[MCGREW] McGrew, D., Kampanakis, P., Fluhrer, S., Gazdag, S., Butin, D., and J. Buchmann, "State Management for Hash-Based Signatures", 2 November 2016, <https://eprint.iacr.org/2016/357>.

[RFC3279] Bassham, L., Polk, W., and R. Housley, "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3279, DOI 10.17487/RFC3279, April 2002, <https://www.rfc-editor.org/rfc/rfc3279>.

[RFC3647] Chokhani, S., Ford, W., Sabett, R., Merrill, C., and S. Wu, "Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework", RFC 3647, DOI 10.17487/RFC3647, November 2003, <https://www.rfc-editor.org/rfc/rfc3647>.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007, <https://www.rfc-editor.org/rfc/rfc4949>.

[RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", RFC 5912, DOI 10.17487/RFC5912, June 2010, <https://www.rfc-editor.org/rfc/rfc5912>.

[RFC8410] Josefsson, S. and J. Schaad, "Algorithm Identifiers for Ed25519, Ed448, X25519, and X448 for Use in the Internet X.509 Public Key Infrastructure", RFC 8410, DOI 10.17487/RFC8410, August 2018, <https://www.rfc-editor.org/rfc/rfc8410>.

[RFC8411] Schaad, J. and R. Andrews, "IANA Registration for the Cryptographic Algorithm Object Identifier Range", RFC 8411, DOI 10.17487/RFC8411, August 2018, <https://www.rfc-editor.org/rfc/rfc8411>.

[SMI-PKIX] IANA, "SMI Security for PKIX Algorithms", n.d., <https://www.iana.org/assignments/smi-numbers/smi-numbers.xhtml#smi-numbers-1.3.6.1.5.5.7.6>.

Appendix A. HSS X.509 v3 Certificate Example

This section shows a self-signed X.509 v3 certificate using HSS.

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            e8:91:d6:06:91:4f:ce:f3
        Signature Algorithm: hss
        Issuer: C = US, ST = VA, L = Herndon, O = Bogus CA
        Validity
            Not Before: May 14 08:58:11 2024 GMT
            Not After : May 14 08:58:11 2034 GMT
        Subject: C = US, ST = VA, L = Herndon, O = Bogus CA
        Subject Public Key Info:
            Public Key Algorithm: hss
                hss public key:
                PQ key material:
                    00:00:00:01:00:00:00:05:00:00:00:04:c0:96:12:
                    8b:ea:38:30:78:eb:f6:fb:43:d7:7f:9f:9e:81:39:
                    e2:7c:b9:34:4e:6e:53:19:f0:ee:68:75:85:83:d3:
                    2b:e9:7b:14:46:9e:4e:c5:e3:5a:18:0b:30:e5:13
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                58:15:AB:F4:CF:03:69:02:60:7A:57:4D:C5:D5:B3:72:
                8A:19:21:68
            X509v3 Authority Key Identifier:
                58:15:AB:F4:CF:03:69:02:60:7A:57:4D:C5:D5:B3:72:
                8A:19:21:68
            X509v3 Basic Constraints: critical
                CA:TRUE
            X509v3 Key Usage: critical
```

```
                Certificate Sign, CRL Sign
        Signature Algorithm: hss
        Signature Value:
            00:00:00:00:00:00:00:00:00:00:04:9c:37:52:ff:b9:d7:
            df:f5:5b:01:ba:50:c2:50:cc:6f:f3:b1:73:df:0c:2a:ea:b3:
            ed:96:1e:ce:e7:58:05:da:8d:a7:77:21:42:32:d9:f9:4a:4d:
            f7:2b:18:2a:1c:5c:69:03:f3:1c:9c:95:6d:31:9a:c9:ca:84:
            4d:ae:b3:8b:c3:71:ac:3f:87:51:be:38:b4:bf:d9:dc:90:1f:
            1e:54:bd:f9:1a:65:70:d4:46:b6:ad:4d:6d:16:b9:fb:29:f4:
            e3:86:42:4a:3f:a4:8f:01:84:9b:44:0b:23:22:9c:97:6d:d5:
            b9:26:39:11:ab:46:82:bd:10:6c:b4:7a:64:ed:c7:40:b0:33:
            f0:b5:81:1c:b4:41:54:9c:30:d9:d2:93:ba:48:8c:4f:d0:25:
            41:60:7b:90:5e:12:20:b7:30:16:16:1e:b7:ee:d8:4b:ee:ed:
            3c:70:fc:ff:36:18:aa:24:23:87:91:65:a8:95:2d:b6:1c:d1:
            02:7b:70:81:8a:18:17:c0:45:62:fe:47:a1:3e:69:54:31:67:
            58:9a:e1:e3:c9:8d:ee:1e:2a:d1:46:75:e9:e4:90:67:01:57:
            92:54:db:b4:ea:de:8b:e7:eb:fc:27:80:9b:d5:da:e0:8e:b0:
            b3:08:ca:6f:a1:1c:f4:40:65:b0:f6:f8:c9:a7:97:04:c8:7c:
            9e:56:ec:2f:4b:cd:45:8b:d7:e6:a7:50:c7:e6:21:2c:17:31:
            23:11:7a:ae:9a:b5:84:5f:e6:5c:82:99:a8:3a:a9:91:87:9a:
            24:5c:83:01:91:7c:fc:cd:be:2e:92:50:fb:12:11:96:08:0d:
            c9:24:0d:bb:6f:fb:59:05:af:7f:96:bc:a3:f4:58:e2:fa:0a:
            4a:f2:4c:f7:b3:1b:81:dd:4a:41:a0:b1:dd:52:4c:bb:6d:c0:
            a8:d9:bb:29:c8:fc:e3:7e:f8:6a:e5:5e:c4:e4:e8:7c:0b:00:
            87:15:75:a2:06:50:97:c6:1f:14:52:79:04:a8:9c:ec:b1:c7:
            6a:46:33:98:b8:63:f7:a7:2c:d4:62:78:94:1c:5d:9d:4f:a6:
            0a:ae:39:50:85:b2:09:8d:62:c9:4c:11:9f:0c:91:a5:ac:2d:
            11:bd:71:b6:0c:ea:34:98:53:fc:2e:cc:7b:a4:9c:2e:7a:a4:
            8d:e2:e8:8c:01:a9:9c:3e:b5:34:77:33:82:01:d4:ef:72:04:
            d6:5b:e5:f6:2c:1b:ae:86:c4:73:02:44:85:d6:f7:ac:a3:e8:
            f6:a9:b5:5c:6d:46:88:da:55:b8:2b:7a:4c:0c:9a:e7:cd:5d:
            62:8a:ca:c8:96:ce:8d:71:7b:d2:c1:0d:9a:35:55:2b:84:3e:
            0e:a5:fa:d6:a0:76:8e:23:b3:df:c9:3b:4f:68:56:1e:e9:3c:
            79:5b:d3:25:54:11:ad:a6:ac:58:11:49:8f:4d:c4:c1:39:99:
            76:3a:a6:d1:2f:57:ad:bf:7c:9d:57:cc:37:0d:29:84:29:7b:
            cb:46:85:c3:81:c5:33:9a:65:c3:2f:01:48:ca:44:6c:f1:84:
            3d:d0:49:c2:c1:05:db:77:4c:b9:72:3d:6f:ce:69:f2:91:c6:
            15:25:8f:da:38:7e:ef:5b:3e:5f:35:ab:a6:78:16:28:42:c1:
            2c:2f:9e:11:53:2c:bd:c4:24:7b:e9:c4:ce:3d:d6:41:c7:5d:
            92:91:c3:37:cb:72:44:d7:0d:70:85:13:0b:ac:b3:0f:b0:e5:
            e3:2e:48:b9:9c:b8:d7:3e:7c:50:69:03:7a:5f:ae:f8:6c:09:
            61:97:6b:ce:cd:e5:f0:55:fe:05:f8:97:1d:9e:81:65:f5:ff:
            9a:7a:8c:96:d8:f8:cf:d8:dc:55:ce:67:7a:00:6b:fd:bb:3f:
            1b:3d:65:94:c1:5a:b6:a0:8e:be:a4:be:26:90:5f:1f:06:d4:
            ea:3f:a6:97:40:8e:bf:18:5c:92:0f:15:e3:05:4a:14:51:1e:
            23:81:ef:cf:f7:a8:88:75:f8:2d:28:37:26:87:27:63:5c:01:
            53:0e:5e:53:d2:a7:18:eb:2f:c0:82:49:05:b0:4d:33:6f:94:
            10:91:77:f8:90:9e:ca:fe:bb:3d:c4:42:d6:89:84:98:42:f4:
```

```
24:b3:b4:db:5e:2b:66:a9:ff:6c:18:d4:79:f8:72:73:53:9b:
02:ed:04:73:77:a4:68:cf:4b:be:4b:16:50:62:87:f9:49:99:
e3:a1:0c:42:92:bc:a9:e3:2d:22:82:35:7f:71:15:88:70:6a:
01:ab:44:64:ad:e5:52:d4:97:ee:bb:44:7b:6e:08:7f:dd:94:
fd:c9:1c:6b:59:d1:92:51:29:03:ce:ec:bf:41:a5:14:69:54:
3a:b4:39:d9:44:5d:f1:b2:f4:5c:6b:9f:c9:5f:bb:fc:c8:c7:
a3:8b:e1:ec:e2:d0:69:5a:40:1c:9c:9d:8a:3d:77:3b:c1:5d:
c0:72:61:4b:37:c5:96:8c:6d:8b:f8:56:da:ac:3e:3c:72:09:
ce:f6:c3:fe:5d:cf:37:d9:68:cd:a7:dd:f7:96:63:da:8c:1d:
df:b8:32:cf:eb:97:11:83:fe:6b:aa:b9:e2:4b:b2:ea:62:73:
c3:1c:e9:40:90:56:4f:12:c3:ba:f4:2b:d9:1c:50:cc:e0:51:
d8:eb:bf:67:28:0c:2d:13:8d:b3:6f:13:6a:1d:a7:54:20:ba:
82:5b:b8:e5:1f:89:f1:67:26:c1:dc:1b:60:57:ed:a6:2c:f2:
17:01:7f:a5:e7:5c:64:c9:3c:08:f2:cf:48:ec:88:84:ef:03:
c2:f5:eb:05:31:7d:fe:7f:3c:71:41:28:17:64:5f:b9:ec:54:
79:d0:b3:98:fb:84:9c:36:8b:43:0b:d4:c9:ec:09:4a:70:13:
62:f2:36:c8:b4:75:cc:2a:77:08:a0:9d:ef:19:d6:88:dc:e2:
b2:4e:40:61:71:cb:c7:c3:de:16:6f:49:7f:5e:d5:17:00:00:
00:05:79:47:12:9f:ce:eb:1d:a8:fd:0d:b0:18:44:6a:ef:54:
28:46:e4:19:f6:2d:3e:74:bb:9d:36:0a:ae:67:4a:28:7a:1b:
80:39:a0:08:2a:28:a0:ec:55:ee:55:aa:a1:cc:94:d4:36:1a:
b3:57:25:30:ad:2c:5e:63:ba:22:fc:aa:7a:59:64:f6:d8:03:
20:28:71:f9:dc:09:fa:4c:81:b9:64:1b:ad:ea:cb:db:18:17:
5d:d8:98:bd:d2:8d:c5:04:7c:5b:92:9a:89:f6:bc:d6:55:c7:
08:5d:3c:58:8e:18:ac:6f:88:a8:d7:9e:d4:ee:5d:f5:21:4e:
a5:8b:19:5f:e3:f4:66:f9:25:4d:f9:c6:60:62:31:72:5c:34:
34:67:1a:a7:6a:7d:54:a3:d8:9b:1f:5b:f8:08:41:79:5b:43
```

```
-----BEGIN CERTIFICATE-----
MIIGnjCCAXagAwIBAgIJAOiR1gaRT87zMA0GCyqGSIb3DQEJEAMRMD8xCzAJBgNV
BAYTAlVTMQswCQYDVQQIDAJWQTEQMA4GA1UEBwwHSGVybmRvbjERMA8GA1UECgwI
Qm9ndXMgQ0EwHhcNMjQwNTE0MDg1ODExWhcNMzQwNTE0MDg1ODExWjA/MQswCQYD
VQQGEwJVUzELMAkGA1UECAwCVkExEDAOBgNVBAcMB0hlcm5kb24xETAPBgNVBAoM
CEJvZ3VzIENBME4wDQYLKoZIhvcNAQkQAxEDPQAAAAABAAAABQAAAATAlhKL6jgw
eOv2+0PXf5+egTnifLk0Tm5TGfDuaHWFg9Mr6XsURp5OxeNaGAsw5ROjYzBhMB0G
A1UdDgQWBBRYFav0zwNpAmB6V03F1bNyihkhaDAfBgNVHSMEGDAWgBRYFav0zwNp
AmB6V03F1bNyihkhaDAPBgNVHRMBAf8EBTADAQH/MA4GA1UdDwEB/wQEAwIBBjAN
BgsqhkiG9w0BCRADEQOCBREAAAAAAAAAAAAAAAAAEnDdS/7nX3/VbAbpQwlDMb/Ox
c98MKuqz7ZYezudYBdqNp3chQjLZ+UpN9ysYKhxcaQPzHJyVbTGaycqETa6zi8Nx
rD+HUb44tL/Z3JAfHlS9+RplcNRGtq1NbRa5+yn044ZCSj+kjwGEm0QLIyKcl23V
uSY5EatGgr0QbLR6ZO3HQLAz8LWBHLRBVJww2dKTukiMT9AlQWB7kF4SILcwFhYe
t+7YS+7tPHD8/zYYqiQjh5FlqJUtthzRAntwgYoYF8BFYv5HoT5pVDFnWJrh48mN
7h4q0UZ16eSQZwFXklTbtOrei+fr/CeAm9Xa4I6wswjKb6Ec9EBlsPb4yaeXBMh8
nlbsL0vNRYvX5qdQx+YhLBcxIxF6rpq1hF/mXIKZqDqpkYeaJFyDAZF8/M2+LpJQ
+xIRlggNySQNu2/7WQWvf5a8o/RY4voKSvJM97Mbgd1KQaCx3VJMu23AqNm7Kcj8
4374auVexOTofAsAhxV1ogZQl8YfFFJ5BKic7LHHakYzmLhj96cs1GJ4lBxdnU+m
Cq45UIWyCY1iyUwRnwyRpawtEb1xtgzqNJhT/C7Me6ScLnqkjeLojAGpnD61NHcz
ggHU73IE1lvl9iwbrobEcwJEhdb3rKPo9qm1XG1GiNpVuCt6TAya581dYorKyJbO
jXF70sENmjVVK4Q+DqX61qB2jiOz38k7T2hWHuk8eVvTJVQRraasWBFJj03EwTmZ
djqm0S9Xrb98nVfMNw0phCl7y0aFw4HFM5plwy8BSMpEbPGEPdBJwsEF23dMuXI9
b85p8pHGFSWP2jh+71s+XzWrpngWKELBLC+eEVMsvcQke+nEzj3WQcddkpHDN8ty
RNcNcIUTC6yzD7Dl4y5IuZy41z58UGkDel+u+GwJYZdrzs3l8FX+BfiXHZ6BZfX/
mnqMltj4z9jcVc5negBr/bs/Gz1llMFatqCOvqS+JpBfHwbU6j+ml0COvxhckg8V
4wVKFFEeI4Hvz/eoiHX4LSg3JocnY1wBUw5eU9KnGOsvwIJJBbBNM2+UEJF3+JCe
yv67PcRC1omEmEL0JLO0214rZqn/bBjUefhyc1ObAu0Ec3ekaM9LvksWUGKH+UmZ
46EMQpK8qeMtIoI1f3EViHBqAatEZK3lUtSX7rtEe24If92U/ckca1nRklEpA87s
v0GlFGlUOrQ52URd8bL0XGufyV+7/MjHo4vh7OLQaVpAHJydij13O8FdwHJhSzfF
loxti/hW2qw+PHIJzvbD/l3PN9lozafd95Zj2owd37gyz+uXEYP+a6q54kuy6mJz
wxzpQJBWTxLDuvQr2RxQzOBR2Ou/ZygMLRONs28Tah2nVCC6glu45R+J8Wcmwdwb
YFftpizyFwF/pedcZMk8CPLPSOyIhO8DwvXrBTF9/n88cUEoF2RfuexUedCzmPuE
nDaLQwvUyewJSnATYvI2yLR1zCp3CKCd7xnWiNzisk5AYXHLx8PeFm9Jf17VFwAA
AAV5RxKfzusdqP0NsBhEau9UKEbkGfYtPnS7nTYKrmdKKHobgDmgCCoooOxV7lWq
ocyU1DYas1clMK0sXmO6Ivyqellk9tgDIChx+dwJ+kyBuWQbrerL2xgXXdiYvdKN
xQR8W5Kaifa81lXHCF08WI4YrG+IqNee1O5d9SFOpYsZX+P0ZvklTfnGYGIxclw0
NGcap2p9VKPYmx9b+AhBeVtD
-----END CERTIFICATE-----
```

Appendix B. XMSS X.509 v3 Certificate Example

This section shows a self-signed X.509 v3 certificate using XMSS.

```
   Certificate:
       Data:
           Version: 3 (0x2)
           Serial Number:
               54:7e:64:70:29:9e:03:c5:7a:a5:5c:78:d1:27:87:8c:
               54:35:17:5d
           Signature Algorithm: xmss
           Issuer: C = FR, L = Paris, O = Bogus XMSS CA
           Validity
               Not Before: Jul 10 08:27:24 2024 GMT
               Not After : Jul  8 08:27:24 2034 GMT
           Subject: C = FR, L = Paris, O = Bogus XMSS CA
           Subject Public Key Info:
               Public Key Algorithm: xmss
                   xmss public key:
                   PQ key material:
                       00:00:00:01:2b:eb:bf:66:14:de:6f:96:5b:4d:2a:
                       50:00:7b:ad:5c:22:b0:13:79:72:02:14:a9:5f:fc:
                       96:e0:9b:78:8e:d6:be:8c:1c:70:3c:d8:dd:78:b2:
                       1a:14:47:be:1f:0d:74:72:3f:36:76:c2:cb:19:ad:
                       29:90:0b:82:de:9b:7f:df
           X509v3 extensions:
               X509v3 Subject Key Identifier:
                   62:CE:35:A5:47:77:FF:21:87:2E:BC:2D:27:E7:8E:F4:
                   35:6B:CF:D8
               X509v3 Authority Key Identifier:
                   62:CE:35:A5:47:77:FF:21:87:2E:BC:2D:27:E7:8E:F4:
                   35:6B:CF:D8
               X509v3 Basic Constraints: critical
                   CA:TRUE
               X509v3 Key Usage: critical
                   Certificate Sign, CRL Sign
       Signature Algorithm: xmss
       Signature Value:
           00:00:00:00:e5:88:a8:b8:73:ad:4d:92:f8:5c:81:c5:8a:63:
           57:6a:a7:3b:54:aa:b6:06:8a:d9:f1:c2:0b:c8:27:1e:4b:a2:
           cf:e2:da:44:ea:e8:f2:40:a8:b9:54:9c:49:36:12:24:df:74:
           ad:e5:29:ef:4f:da:88:0d:21:5d:3b:64:63:27:d0:84:b5:95:
           7a:30:18:37:cd:34:17:dd:ac:9d:9e:48:db:74:07:79:84:21:
           5a:f0:26:cd:21:64:7b:77:33:48:58:67:9b:2c:b2:85:6d:cc:
           ec:31:4b:2f:51:55:3a:85:e1:ca:04:15:ce:6e:47:39:f5:e9:
           31:45:41:ed:71:c6:4f:96:f5:ae:64:6a:bd:72:d0:8c:17:02:
           99:10:1d:14:34:ca:e5:47:e3:f7:66:96:96:11:d5:97:76:76:
           83:f1:84:a5:b6:00:5e:3e:67:97:7a:32:dc:c8:eb:4c:29:46:
           77:99:d6:da:45:e6:7b:8c:45:6d:b5:29:6b:fd:98:a2:89:8d:
           0c:30:42:f5:0b:7c:97:c5:b1:1d:e2:da:67:a9:48:a4:9e:29:
           f4:60:3f:4d:1d:48:83:82:38:ef:fa:cb:1d:86:11:a1:15:94:
           fb:d5:ee:68:f9:44:b9:3d:54:70:f3:be:17:8d:d7:2e:85:2d:
```

```
5c:d0:a0:c5:99:52:cc:79:e7:1c:18:d9:6e:3d:0f:6c:05:51:
33:28:35:e2:02:59:5f:1f:ed:78:0a:c6:62:f0:7d:fe:73:96:
03:4c:b4:42:e3:00:c2:d7:cb:eb:51:10:c4:0c:64:b8:37:fe:
85:d0:8e:11:6d:a6:16:77:b1:1e:01:d9:1e:f3:10:9c:dd:01:
bc:38:75:5e:8f:58:9e:5b:6c:7b:0a:41:08:59:35:a9:3a:83:
19:e0:7d:a1:f5:cf:a3:1c:4e:07:e1:ad:03:95:f2:d3:8b:79:
33:f8:52:22:53:1b:1e:32:9a:61:3f:c4:7c:9a:e8:d5:b5:28:
f1:84:65:d5:c1:fc:4d:16:93:88:93:69:ca:fa:94:a0:95:4e:
23:ae:1e:60:e0:e8:b4:bf:ff:16:95:71:0f:31:74:bb:be:b8:
5a:eb:24:95:8b:95:28:13:cd:e3:a9:65:f7:f5:6e:9b:a9:a9:
7a:05:ce:ab:f0:54:62:d9:12:f8:a1:1a:68:df:af:15:8f:8a:
df:67:27:c9:ed:bd:e1:81:a6:8d:9a:84:f3:91:36:d9:89:74:
8e:ef:84:dc:5c:03:1a:08:e4:d7:f0:72:fc:6d:8a:01:34:94:
e5:ff:08:51:1b:80:5f:e7:07:d8:9f:25:e4:1d:c3:f8:e5:d0:
9c:50:cf:66:71:f9:cc:f7:c0:a7:d0:66:01:b7:17:a0:5f:66:
97:a4:ff:62:ac:1c:a0:63:0d:30:28:e9:90:d5:59:a4:48:d8:
07:87:02:4b:3f:68:23:a5:04:dc:b3:d7:45:f6:dc:b0:ec:c6:
90:a6:1c:a1:f8:7e:84:ba:63:7e:5a:64:14:78:58:f5:75:c0:
f5:e1:1d:bd:49:57:c0:40:08:07:99:7f:43:2e:e2:25:d8:ed:
a3:1a:e3:78:f1:78:af:02:49:54:36:59:8e:d3:72:a5:0b:52:
32:bd:17:a2:cf:e1:47:21:28:3d:ba:b6:24:d9:18:f9:44:73:
35:ed:29:a4:18:bc:ed:68:cd:4a:9a:34:cb:1a:2f:b3:5f:ba:
73:9b:18:ee:7a:a8:92:25:65:25:81:04:63:1c:22:2b:b8:ba:
81:21:bc:f9:9d:a8:78:98:75:bc:ed:4a:c6:b7:6f:c0:91:24:
eb:1d:f9:5d:e0:e3:78:4e:05:f6:34:0f:7b:41:54:49:20:a2:
30:66:94:f1:da:c1:6c:3f:5e:10:92:92:a3:0c:7e:e8:8b:26:
11:1c:d7:68:c9:31:79:b3:a4:d5:63:00:68:c3:e3:86:2d:09:
92:4b:2d:63:7d:b8:03:a4:4c:60:b4:2c:12:d5:0b:9f:16:28:
ea:88:2f:bb:1c:19:0b:0f:40:3d:67:e8:0b:fa:c6:e3:39:44:
b2:bd:8a:3f:21:dd:aa:ec:a3:8c:48:dd:4c:99:43:86:d7:48:
81:6b:e5:b9:bb:59:9f:1c:0f:3f:11:f7:7c:4b:67:a8:95:c2:
7c:cb:3b:66:b0:79:a6:55:6f:6d:b0:29:8a:5e:7b:ee:30:68:
f3:dd:41:29:91:f6:79:71:ae:8d:21:70:78:1d:5d:d2:f7:cf:
e7:42:38:d1:8c:52:a6:a6:f6:b1:38:b1:2b:23:81:e1:1f:21:
6d:99:3f:10:eb:b1:a9:73:b8:3e:31:99:cc:dd:2b:df:58:27:
db:0b:5a:29:99:8f:b1:9f:e9:31:42:d0:26:db:53:b7:7e:30:
41:95:c3:f0:07:83:bb:b0:63:b5:16:48:f2:a6:60:2f:32:5d:
22:a1:da:76:4e:37:26:53:0d:95:7b:2d:b9:05:2f:93:2b:d4:
df:c1:02:5b:f7:a5:a2:4f:11:5c:80:f4:f0:bd:c7:ea:3c:db:
6f:e2:eb:6c:7f:c3:58:d9:31:77:4b:4d:f7:ce:bb:d6:c8:64:
a3:01:d5:f9:a4:8d:e8:f0:ee:09:06:2c:0b:3c:ac:0a:57:d8:
e4:81:79:ea:4a:bd:51:03:88:4c:d0:4c:0b:c4:0c:7e:2d:e7:
df:1b:67:62:c0:d1:9c:ad:bb:d3:f0:75:dd:83:aa:70:99:2c:
19:78:3d:26:2b:47:6f:24:c1:60:02:1e:4b:75:04:91:1f:08:
1c:b3:79:a0:9b:db:fb:5d:3f:c7:e3:09:1f:41:3e:64:bb:ad:
19:3d:35:e1:a6:f4:69:0b:a2:04:37:42:95:c6:c7:e5:f4:56:
0e:67:5b:78:34:bb:07:f1:8f:e7:73:5b:87:d7:df:c9:2d:8d:
8c:42:76:87:15:85:4b:23:03:20:34:e1:1b:f6:0c:1e:84:53:
```

```
d9:1b:4e:d9:31:43:38:3b:88:12:84:d8:2a:38:b1:ce:0f:c7:
07:d4:63:2d:97:89:1c:b3:44:99:eb:d4:df:32:74:be:0d:63:
11:22:fd:fa:8e:e2:0b:56:12:56:0c:46:16:ad:44:10:26:98:
dc:cf:c9:95:67:3e:11:c1:76:fa:b8:12:ea:96:f6:d9:91:ac:
bf:49:b9:1c:8e:15:05:53:ac:9e:04:d2:5b:b8:87:bf:81:50:
f7:02:a4:c0:9c:18:0f:45:ac:7a:82:cf:46:15:42:40:09:32:
89:a5:ea:90:a5:99:68:f9:93:0c:7b:d6:7a:a8:e9:51:e2:90:
9e:b9:ed:21:db:d9:7e:de:dc:62:6b:44:6b:9f:81:c5:77:39:
8e:1d:78:30:de:dc:53:80:e0:c3:fa:fa:94:68:28:91:98:86:
ff:86:04:a9:bd:58:7c:31:37:1f:db:9a:29:f3:c1:48:10:20:
71:5f:fc:35:13:eb:7b:12:e2:7d:1c:cc:97:fe:8f:5c:a2:dd:
f6:d2:a3:b2:ea:51:b3:ef:b1:1e:79:0b:00:53:f4:f2:52:75:
5a:d7:17:c5:31:a0:54:4e:2b:28:2c:4f:6b:7a:27:3a:2c:04:
da:b3:1d:04:4e:a4:4e:94:5c:a8:91:70:ab:c0:4b:75:9f:b3:
6a:a9:4e:8a:22:e9:7f:fd:ec:53:e7:6a:6d:32:0b:8b:ab:4c:
e7:7d:72:ec:04:62:1c:1a:45:1e:33:8e:37:ae:6a:2f:c8:fb:
f3:69:ed:11:01:f3:f4:57:e9:29:d5:3b:0c:9c:0c:c4:cb:c3:
38:5c:01:e7:d6:31:c3:d8:ce:24:d7:be:71:9b:c8:96:13:ca:
5c:5d:e4:92:40:af:86:a0:4b:ff:a7:55:39:70:fd:ac:0a:e1:
87:c7:01:4b:c3:41:36:c6:c6:33:8f:4f:25:4a:8d:70:92:ac:
7c:95:cc:49:a9:dc:d6:6a:67:52:a5:5b:7f:2f:bb:91:e3:be:
d6:28:fc:22:d0:72:66:e8:09:73:a7:23:c6:a6:89:38:0b:e5:
d0:b3:f1:40:38:9c:4d:17:96:11:17:44:ef:e3:94:51:91:4c:
5d:fe:d9:ed:c3:76:a0:2d:3b:dc:8d:b9:31:15:f6:75:58:74:
2f:57:b4:29:21:29:6d:5f:eb:06:71:0a:f4:db:ff:c6:2f:16:
73:a7:76:6b:d0:5b:a7:21:5c:fd:f0:11:e8:6f:9b:d0:c9:c9:
fe:35:76:4a:4a:63:9b:ba:48:ac:af:4f:91:67:9c:5c:47:d8:
e3:2d:03:12:5e:f1:cb:56:34:75:69:95:ad:68:96:6c:e7:4a:
91:72:fb:9b:ba:e8:92:56:fb:9a:5b:5d:3b:9d:d3:c5:c4:52:
42:1b:f9:4a:47:42:dd:77:49:da:2b:bd:d7:94:5f:7b:b8:64:
b9:06:32:7c:ea:d1:36:f6:95:b8:57:41:1b:6e:66:31:2c:ee:
87:7a:5c:19:2f:d8:95:4a:16:93:48:f3:97:25:3d:24:61:1e:
d0:63:37:ee:3a:c9:a3:46:c5:94:a0:7e:24:cc:7f:72:8d:14:
9e:3c:33:ec:cd:9a:dd:b5:08:90:98:19:95:85:38:ff:ff:d2:
1e:bf:a6:c4:97:13:2b:3d:47:e9:57:59:d3:7d:99:01:6e:53:
4d:c0:82:97:fb:89:d6:7c:b7:23:0e:7d:6e:23:88:53:06:8f:
16:ff:40:0a:1b:cd:d5:1e:91:01:3e:77:3a:5f:c1:57:3a:7b:
c6:d5:51:d7:e2:ec:89:12:6b:9d:03:e4:9d:bb:7d:4e:02:bf:
67:8d:03:ca:90:56:f0:9a:97:4b:02:2d:4c:31:89:82:76:97:
fe:2f:d5:0a:3d:ea:0d:38:6c:30:75:5f:ae:91:53:d7:45:64:
df:ba:0b:22:80:44:85:6d:0e:5c:29:7f:82:9e:54:a3:7a:95:
be:96:79:66:9d:5b:a2:d6:2e:47:c6:99:7d:2b:32:dc:f2:b6:
02:91:6d:63:d4:93:45:60:c4:42:71:10:9e:fb:90:2f:e6:75:
71:ce:78:70:c1:da:ff:e1:47:fe:79:2b:8e:9a:81:bf:dd:02:
e3:78:39:71:17:b3:23:14:11:9d:29:8e:21:a1:98:b0:ac:03:
5a:6c:9e:62:64:ef:4f:03:ca:37:a6:ed:e4:78:d5:0d:99:29:
f5:5c:61:e6:48:cb:97:0e:5e:f9:2c:f6:b6:c7:7c:0c:a4:f7:
1a:f7:67:b5:5c:03:bf:bf:7a:e2:4d:a2:9b:5d:5d:5f:51:d0:
```

```
            d6:52:8f:2a:20:68:08:bb:f0:9c:05:0e:ef:b3:49:0c:2a:1d:
            8f:f9:03:b7:61:09:71:88:7d:e2:8c:e4:b8:ac:98:1b:c3:80:
            55:a1:6b:dd:13:a2:29:4f:93:93:d3:d5:01:31:3f:7b:39:0e:
            3a:57:6c:eb:5c:6a:5f:1b:ad:97:bd:97:23:18:91:05:0e:2b:
            b4:b1:11:ee:f8:58:c7:08:d0:de:a2:3e:ba:54:8d:3d:63:da:
            91:50:3a:24:8d:19:18:23:2e:cf:30:8d:5d:e3:e7:02:93:fa:
            c8:f8:ea:05:e6:eb:06:80:90:4d:15:58:3d:26:98:13:4b:b0:
            ac:dd:90:2e:d0:e1:eb:71:32:83:5d:2a:a9:b9:b5:24:fc:e9:
            ec:18:ca:c9:a1:05:59:3e:fa:af:ed:4e:86:b1:fe:40:47:9b:
            42:77:af:9c:2b:a0:e2:3e:fd:51:ab:02:77:e8:f1:39:45:aa:
            54:b6:14:d4:14:20:fc:36:81:e6:04:98:8a:a0:c0:8a:cf:ae:
            f6:b5:dc:b7:eb:26:86:d3:cf:1c:38:65:54:04:b1:b5:09:48:
            f5:2d:07:ba:f8:eb:49:bd:d9:b1:54:ea:ac:c2:0d:20:10:79:
            c1:cb:e9:dc:2d:ff:55:50:4f:f6:05:02:78:31:33:6f:15:7e:
            24:5a:66:23:70:b3:b2:0c:17:39:ce:15:38:c5:ff:60:16:38:
            60:74:72:c9:70:d8:59:b7:80:7f:da:f6:67:3f:d0:ba:be:1b:
            a1:87:da:92:2d:a3:6c:99:29:57:aa:cb:d1:8d:66:f1:2d:c9:
            56:60:24:56:4b:19:9f:f5:65:84:89:86:7d:4d:8b:f8:5b:60:
            dd:af:2d:66:76:6c:66:d9:c6:f5:39:25:6c:e5:7b:43:97:64:
            5c:c5:20:1e:3d:b5:dc:92:b2:9c:d8:1b:1b:e0:bc:44:7b:9c:
            95:c5:53:48:91:b2:a5:46:16:bf:50:af:a5:44:cc:54:78:3f:
            ed:20:d8:2e:0b:41:3d:f1:04:9d:df:3c:4a:d7:81:04:ff:8c:
            b7:79:f8:51:8d:b7:2e:ac:2c:54:e6:fc:43:76:8e:f9:be:8c:
            b8:5c:ad:c4:13:af:b0:6e:3b:d1:82:57:1e:f5:52:84:ca:cc:
            d2:68:f3:2d:04:ff:27:0a:e6:a2:fa:c0:a9:97:d6:64:45:18:
            5c:6f:9e:c1:64:22:66:db:56:02:c3:a8:57:fc:87:1b:5c:43:
            15:8e:58:fc:f2:00:0b:4f:6a:4b:a0:5c:da:f2:e5:1b:82:4a:
            6b:ef:db:63:d7:7d:93:1d:2f:20:78:37:17:22:82:cd:6b:c1:
            83:61:05:81:99:0c:25:29:d6:5f:22:bc:06:67:7d:67
```

```
-----BEGIN CERTIFICATE-----
MIILSDCCAW+gAwIBAgIUVH5kcCmeA8V6pVx40SeHjFQ1F10wCgYIKwYBBQUHBiIw
NTELMAkGA1UEBhMCRlIxDjAMBgNVBAcMBVBhcmlzMRYwFAYDVQQKDA1Cb2d1cyBY
TVNTIENBMB4XDTI0MDcxMDA4MjcyNFoXDTM0MDcwODA4MjcyNFowNTELMAkGA1UE
BhMCRlIxDjAMBgNVBAcMBVBhcmlzMRYwFAYDVQQKDA1Cb2d1cyBYTVNTIENBMFMw
CgYIKwYBBQUHBiIDRQAAAAABK+u/ZhTeb5ZbTSpQAHutXCKwE3lyAhSpX/yW4Jt4
jta+jBxwPNjdeLIaFEe+Hw10cj82dsLLGa0pkAuC3pt/36NjMGEwHQYDVR0OBBYE
FGLONaVHd/8hhy68LSfnjvQ1a8/YMB8GA1UdIwQYMBaAFGLONaVHd/8hhy68LSfn
jvQ1a8/YMA8GA1UdEwEB/wQFMAMBAf8wDgYDVR0PAQH/BAQDAgEGMAoGCCsGAQUF
BwYiA4IJxQAAAAAA5YiouHOtTZL4XIHFimNXaqc7VKq2BorZ8cILyCceS6LP4tpE
6ujyQKi5VJxJNhIk33St5SnvT9qIDSFdO2RjJ9CEtZV6MBg3zTQX3aydnkjbdAd5
hCFa8CbNIWR7dzNIWGebLLKFbczsMUsvUVU6heHKBBXObkc59ekxRUHtccZPlvWu
ZGq9ctCMFwKZEB0UNMrlR+P3ZpaWEdWXdnaD8YSltgBePmeXejLcyOtMKUZ3mdba
ReZ7jEVttSlr/ZiiiY0MMEL1C3yXxbEd4tpnqUiknin0YD9NHUiDgjjv+ssdhhGh
FZT71e5o+US5PVRw874XjdcuhS1c0KDFmVLMeeccGNluPQ9sBVEzKDXiAllfH+14
CsZi8H3+c5YDTLRC4wDC18vrURDEDGS4N/6F0I4RbaYWd7EeAdke8xCc3QG8OHVe
j1ieW2x7CkEIWTWpOoMZ4H2h9c+jHE4H4a0DlfLTi3kz+FIiUxseMpphP8R8mujV
tSjxhGXVwfxNFpOIk2nK+pSglU4jrh5g4Oi0v/8WlXEPMXS7vrha6ySVi5UoE83j
```

```
qWX39W6bqal6Bc6r8FRi2RL4oRpo368Vj4rfZyfJ7b3hgaaNmoTzkTbZiXSO74Tc
XAMaCOTX8HL8bYoBNJTl/whRG4Bf5wfYnyXkHcP45dCcUM9mcfnM98Cn0GYBtxeg
X2aXpP9irBygYw0wKOmQ1VmkSNgHhwJLP2gjpQTcs9dF9tyw7MaQphyh+H6EumN+
WmQUeFj1dcD14R29SVfAQAgHmX9DLuIl2O2jGuN48XivAklUNlmO03KlC1IyvRei
z+FHISg9urYk2Rj5RHM17SmkGLztaM1KmjTLGi+zX7pzmxjueqiSJWUlgQRjHCIr
uLqBIbz5nah4mHW87UrGt2/AkSTrHfld4ON4TgX2NA97QVRJIKIwZpTx2sFsP14Q
kpKjDH7oiyYRHNdoyTF5s6TVYwBow+OGLQmSSy1jfbgDpExgtCwS1QufFijqiC+7
HBkLD0A9Z+gL+sbjOUSyvYo/Id2q7KOMSN1MmUOG10iBa+W5u1mfHA8/Efd8S2eo
lcJ8yztmsHmmVW9tsCmKXnvuMGjz3UEpkfZ5ca6NIXB4HV3S98/nQjjRjFKmpvax
OLErI4HhHyFtmT8Q67Gpc7g+MZnM3SvfWCfbC1opmY+xn+kxQtAm21O3fjBBlcPw
B4O7sGO1FkjypmAvMl0iodp2TjcmUw2Vey25BS+TK9TfwQJb96WiTxFcgPTwvcfq
PNtv4utsf8NY2TF3S033zrvWyGSjAdX5pI3o8O4JBiwLPKwKV9jkgXnqSr1RA4hM
0EwLxAx+LeffG2diwNGcrbvT8HXdg6pwmSwZeD0mK0dvJMFgAh5LdQSRHwgcs3mg
m9v7XT/H4wkfQT5ku60ZPTXhpvRpC6IEN0KVxsfl9FYOZ1t4NLsH8Y/nc1uH19/J
LY2MQnaHFYVLIwMgNOEb9gwehFPZG07ZMUM4O4gShNgqOLHOD8cH1GMtl4kcs0SZ
69TfMnS+DWMRIv36juILVhJWDEYWrUQQJpjcz8mVZz4RwXb6uBLqlvbZkay/Sbkc
jhUFU6yeBNJbuIe/gVD3AqTAnBgPRax6gs9GFUJACTKJpeqQpZlo+ZMMe9Z6qOlR
4pCeue0h29l+3txia0Rrn4HFdzmOHXgw3txTgODD+vqUaCiRmIb/hgSpvVh8MTcf
25op88FIECBxX/w1E+t7EuJ9HMyX/o9cot320qOy6lGz77EeeQsAU/TyUnVa1xfF
MaBUTisoLE9reic6LATasx0ETqROlFyokXCrwEt1n7NqqU6KIul//exT52ptMguL
q0znfXLsBGIcGkUeM443rmovyPvzae0RAfP0V+kp1TsMnAzEy8M4XAHn1jHD2M4k
175xm8iWE8pcXeSSQK+GoEv/p1U5cP2sCuGHxwFLw0E2xsYzj08lSo1wkqx8lcxJ
qdzWamdSpVt/L7uR477WKPwi0HJm6AlzpyPGpok4C+XQs/FAOJxNF5YRF0Tv45RR
kUxd/tntw3agLTvcjbkxFfZ1WHQvV7QpISltX+sGcQr02//GLxZzp3Zr0FunIVz9
8BHob5vQycn+NXZKSmObukisr0+RZ5xcR9jjLQMSXvHLVjR1aZWtaJZs50qRcvub
uuiSVvuaW107ndPFxFJCG/lKR0Ldd0naK73XlF97uGS5BjJ86tE29pW4V0EbbmYx
LO6HelwZL9iVShaTSPOXJT0kYR7QYzfuOsmjRsWUoH4kzH9yjRSePDPszZrdtQiQ
mBmVhTj//9Iev6bElxMrPUfpV1nTfZkBblNNwIKX+4nWfLcjDn1uI4hTBo8W/0AK
G83VHpEBPnc6X8FXOnvG1VHX4uyJEmudA+Sdu31OAr9njQPKkFbwmpdLAi1MMYmC
dpf+L9UKPeoNOGwwdV+ukVPXRWTfugsigESFbQ5cKX+CnlSjepW+lnlmnVui1i5H
xpl9KzLc8rYCkW1j1JNFYMRCcRCe+5Av5nVxznhwwdr/4Uf+eSuOmoG/3QLjeDlx
F7MjFBGdKY4hoZiwrANabJ5iZO9PA8o3pu3keNUNmSn1XGHmSMuXDl75LPa2x3wM
pPca92e1XAO/v3riTaKbXV1fUdDWUo8qIGgIu/CcBQ7vs0kMKh2P+QO3YQlxiH3i
jOS4rJgbw4BVoWvdE6IpT5OT09UBMT97OQ46V2zrXGpfG62XvZcjGJEFDiu0sRHu
+FjHCNDeoj66VI09Y9qRUDokjRkYIy7PMI1d4+cCk/rI+OoF5usGgJBNFVg9JpgT
S7Cs3ZAu0OHrcTKDXSqpubUk/OnsGMrJoQVZPvqv7U6Gsf5AR5tCd6+cK6DiPv1R
qwJ36PE5RapUthTUFCD8NoHmBJiKoMCKz672tdy36yaG088cOGVUBLG1CUj1LQe6
+OtJvdmxVOqswg0gEHnBy+ncLf9VUE/2BQJ4MTNvFX4kWmYjcLOyDBc5zhU4xf9g
FjhgdHLJcNhZt4B/2vZnP9C6vhuhh9qSLaNsmSlXqsvRjWbxLclWYCRWSxmf9WWE
iYZ9TYv4W2Ddry1mdmxm2cb1OSVs5XtDl2RcxSAePbXckrKc2Bsb4LxEe5yVxVNI
kbKlRha/UK+lRMxUeD/tINguC0E98QSd3zxK14EE/4y3efhRjbcurCxU5vxDdo75
voy4XK3EE6+wbjvRglce9VKEyszSaPMtBP8nCuai+sCpl9ZkRRhcb57BZCJm21YC
w6hX/IcbXEMVjlj88gALT2pLoFza8uUbgkpr79tj132THS8geDcXIoLNa8GDYQWB
mQwlKdZfIrwGZ31n
-----END CERTIFICATE-----
```

Appendix C. XMSS^MT X.509 v3 Certificate Example

This section shows a self-signed X.509 v3 certificate using XMSS^MT.

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            5c:22:ad:8a:06:51:9e:67:02:6a:2d:43:3e:8b:c7:23:
            43:77:80:c8
        Signature Algorithm: xmssmt
        Issuer: C = FR, L = Paris, O = Bogus XMSSMT CA
        Validity
            Not Before: Jul 10 08:28:04 2024 GMT
            Not After : Jul  8 08:28:04 2034 GMT
        Subject: C = FR, L = Paris, O = Bogus XMSSMT CA
        Subject Public Key Info:
            Public Key Algorithm: xmssmt
                xmssmt public key:
                PQ key material:
                    00:00:00:01:4b:a7:89:11:6f:fc:1d:fb:d3:e7:71:
                    73:b8:a2:48:ef:53:b9:9d:1f:c6:8a:7c:be:4f:8a:
                    29:fa:41:fd:bd:da:20:7f:f6:3b:b0:c5:b8:a7:c2:
                    f2:5a:f2:26:14:eb:36:f0:26:2f:87:74:fb:0e:d5:
                    7e:17:a0:d1:4d:b6:cf:51
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                7C:7D:59:B8:95:61:D5:03:6A:1E:3D:F1:24:AB:1D:ED:
                04:CD:DB:5F
            X509v3 Authority Key Identifier:
                7C:7D:59:B8:95:61:D5:03:6A:1E:3D:F1:24:AB:1D:ED:
                04:CD:DB:5F
            X509v3 Basic Constraints: critical
                CA:TRUE
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
    Signature Algorithm: xmssmt
    Signature Value:
        00:00:00:57:c4:98:89:ff:d9:0a:8e:6e:6f:16:95:8c:ec:35:
        42:21:c2:ca:56:ed:f8:81:f1:b2:4f:2b:6d:73:f4:37:55:fc:
        f4:4e:15:eb:6b:90:de:34:fe:d6:96:70:94:8d:c1:e7:4a:32:
        49:30:3a:40:a4:67:d2:fb:da:f8:d8:a1:7a:48:22:1c:e3:98:
        bc:d0:68:85:29:c9:e5:f7:5c:56:d8:9c:80:be:68:ed:11:eb:
        39:0f:ef:cb:09:b2:28:30:a6:2b:05:bc:de:11:22:be:c4:dc:
        08:9a:3d:b4:49:37:1f:54:5e:5f:2d:93:62:b0:95:c5:5d:23:
        92:f3:55:40:78:19:00:56:9e:a2:f1:0e:4b:ae:75:d6:92:09:
        b1:79:ec:c9:18:67:19:09:86:83:74:5d:0a:06:ab:da:f0:af:
        02:97:4d:d7:73:06:8b:a2:84:c7:09:af:dd:8b:15:39:e4:30:
```

```
9f:c9:00:25:a8:33:4d:de:e8:25:b6:35:0b:51:bf:7a:34:a7:
e8:84:e8:fa:39:5b:aa:37:6e:95:89:ac:26:4a:4e:ca:be:29:
08:4b:3c:28:a7:85:6a:ad:5a:d2:93:eb:12:e1:9a:87:1c:40:
3b:cf:15:6c:43:4e:88:21:54:52:7e:0d:6d:17:29:8d:15:6f:
ef:42:5a:a9:25:d0:97:80:61:31:22:a4:9f:25:17:51:ad:0b:
a1:cb:93:b4:f5:a6:b0:22:1b:6d:50:64:2a:48:bd:05:16:88:
00:e3:7b:56:d0:03:b3:7a:2d:6a:0b:f3:de:a2:8c:6e:81:80:
2c:8f:e9:d8:78:ed:5b:99:c9:13:d1:b6:eb:78:c3:40:2b:a1:
7a:84:0a:ba:12:87:5e:1d:38:24:22:8f:c0:a3:65:1c:1c:ce:
2d:8e:e5:2f:1f:be:93:5c:fe:1c:cd:a8:9d:7e:7e:cf:18:e2:
9c:c5:54:dc:62:61:74:23:55:64:66:21:96:4c:a7:2e:8a:94:
a6:35:10:a5:e8:5e:6e:91:ac:a8:cb:ed:51:2b:66:45:03:f5:
87:ed:4d:8c:4e:6d:54:80:a1:33:8a:84:9d:23:31:90:c6:05:
11:a7:9d:bd:51:0a:73:47:bc:08:49:11:b3:98:ff:01:14:69:
d7:c0:a0:0c:55:e4:5e:e2:fa:84:ac:27:b3:85:2c:99:71:52:
9c:33:f8:9d:8c:d2:13:bc:6e:18:79:15:a7:02:ee:15:eb:27:
d8:af:24:38:02:9c:ca:30:f3:e2:30:41:2f:62:a2:2c:a5:81:
1b:71:6d:b1:94:bd:c6:3d:9e:5e:51:45:de:5b:f4:d7:e6:35:
e7:d8:7c:d5:98:ec:7e:0e:f8:9d:c1:a7:7b:b3:65:b1:a1:4b:
2d:ec:d9:12:45:6b:1f:0b:1c:6b:3b:0a:66:76:39:f4:cc:9b:
e1:b7:17:f7:53:fc:c3:a6:18:f7:2e:45:52:b1:18:99:75:d1:
69:bb:77:c8:1a:84:5f:06:b5:8b:cb:02:b0:b2:0f:bf:17:18:
65:3d:a7:72:5b:71:9f:92:7e:3a:df:84:cc:65:5c:c4:5b:70:
fd:cc:38:9e:12:6e:f9:ff:1f:02:fc:ca:f5:68:86:fc:ca:71:
f1:3d:7b:32:b4:d4:c3:a2:20:16:3f:12:07:71:95:3b:d4:b1:
1e:fc:8c:1f:34:8c:c8:ab:8c:bb:75:93:c1:1a:d2:85:3e:9a:
e6:04:86:88:de:27:46:ca:f3:f7:f3:8e:54:18:ea:aa:ae:14:
02:b1:4a:6a:e0:24:77:40:28:8d:37:27:9c:87:6a:81:09:d2:
01:4d:20:7f:de:84:a8:80:8c:8e:63:82:be:66:df:87:30:5c:
b8:71:0a:e9:91:68:71:6e:97:97:f0:27:4e:fa:ae:6a:85:ac:
80:cd:38:48:49:c1:2b:9d:db:54:c5:f0:bf:fa:06:e8:96:3a:
c0:95:f0:88:bd:8e:80:78:3d:dc:ad:5d:0a:56:dd:c7:80:9f:
fc:64:58:4d:6d:27:f6:d7:1a:8c:b2:1c:09:ea:7d:4f:74:99:
0d:4a:0c:b8:b0:ef:74:dd:6f:6f:dc:e5:83:e1:e3:c2:e8:58:
17:b8:44:8a:2d:ec:df:54:f6:1f:67:a2:b3:c5:19:fb:b9:c7:
1b:3c:ea:bd:2c:e1:43:65:d1:5a:17:dc:93:9d:c5:85:0c:55:
34:13:49:15:92:e2:52:14:d1:81:aa:62:02:1a:ba:c9:b0:53:
85:8e:7b:d1:4e:34:76:ac:79:d7:b3:48:92:bf:55:7e:2d:5c:
cd:32:9b:c1:41:a7:a3:cd:b7:94:5c:96:1e:3e:27:4d:eb:f0:
61:4b:a4:e3:3c:bb:69:85:37:e9:9c:98:f4:68:7a:61:77:8c:
bd:b9:30:d6:f1:fd:69:78:3f:96:99:7b:69:39:90:b3:7c:b6:
88:ed:cd:19:da:42:64:e5:32:4c:a2:30:f7:c4:e8:27:93:70:
ed:fa:5e:ca:8e:7a:d1:13:af:15:b1:59:c9:9b:91:61:0b:06:
d5:cc:2e:80:bb:49:93:dd:be:53:88:be:af:80:64:7c:5e:be:
7b:8b:e7:5f:39:af:ab:67:42:6b:06:aa:ef:d6:69:af:a9:00:
1f:a0:15:10:04:3e:db:93:b2:37:db:eb:85:59:43:a2:8d:8f:
06:8c:cb:a2:1d:a8:3c:9f:f4:a4:7c:c8:cd:ff:f0:a8:79:0f:
e7:d8:94:67:ec:17:3f:fa:6e:04:07:4f:bf:86:04:6c:fc:46:
```

```
87:b5:10:85:a4:07:e8:af:a9:ec:5d:28:5c:80:8c:31:cc:c7:
b3:81:17:0b:4b:7d:1c:9e:74:02:1e:ef:de:0d:1b:c1:c0:04:
4d:46:fd:dc:0b:a4:c6:33:e6:85:0a:60:39:4d:0b:f9:49:44:
33:e0:15:99:19:bf:c7:8a:c6:96:04:93:37:6b:5d:e8:be:73:
d4:80:b8:81:0f:9a:91:44:cf:72:02:d3:c9:f8:e0:7d:d2:9b:
2b:ff:eb:42:6e:38:7e:dc:cd:a7:90:c5:2c:2b:a0:23:37:b9:
64:10:a6:27:68:47:c5:f1:e8:8d:41:c1:49:e8:35:48:ce:c8:
08:4c:ad:f2:ad:5d:e9:62:eb:c9:3c:61:85:18:c6:34:73:fd:
26:a4:f0:50:83:9b:64:54:aa:55:6c:d8:a2:21:81:ff:9c:27:
39:1f:c3:a2:0e:e5:53:b1:d7:fa:1f:ef:29:8b:c2:90:98:ea:
2e:dd:45:bf:c3:6c:a3:93:47:99:03:18:25:e8:a5:ee:2e:77:
eb:7f:f4:49:49:59:98:c1:fc:ab:1e:ad:20:bd:f8:24:fd:21:
1b:da:5a:07:55:c8:50:05:31:50:93:b2:f8:6e:db:73:4d:5f:
34:aa:f3:34:83:90:f0:41:6d:c8:43:56:d1:75:07:f5:16:20:
b3:99:b2:c7:34:25:c4:0e:74:5a:51:0f:7b:3b:7f:6a:a9:41:
17:b5:47:62:2d:4f:b9:61:97:60:e9:ae:ca:ad:31:6e:4b:0a:
47:9c:53:66:a3:4e:c3:96:7c:01:a0:8e:ae:83:45:42:e6:92:
12:8e:97:6f:e8:a0:b7:7d:a6:74:24:aa:20:b0:fa:9e:98:e8:
7c:b4:da:30:e9:94:08:96:b7:b9:53:4f:75:5f:0c:4d:82:e3:
cf:6e:bc:fa:23:4f:fa:33:17:7c:98:b6:1e:47:89:3e:d9:a1:
aa:42:19:25:ae:9e:3f:53:44:ac:91:96:d8:55:c3:40:1d:fa:
ad:86:38:62:bd:27:2f:26:34:be:ad:9a:01:44:42:c8:54:a5:
3a:e9:0a:ff:f8:41:6d:38:1e:e2:3d:08:3a:94:4f:1e:60:d0:
b1:c2:8e:94:34:f0:30:3e:f0:91:25:ee:98:34:b4:8d:95:4e:
cf:ed:1d:61:89:c9:59:10:68:f2:bc:2e:5c:bd:c0:0f:1d:9c:
2f:7c:c0:27:25:14:9b:de:a3:74:64:28:14:2c:a2:b2:90:3a:
a4:6a:50:e9:8e:ca:78:e5:b6:74:56:e0:92:69:7d:b4:2e:e0:
e7:66:92:16:92:a0:c3:db:4f:d3:d0:57:4d:4a:28:ee:b7:cc:
04:ef:17:d9:fc:01:bb:1e:b2:5b:02:3d:1f:5a:85:73:a1:81:
96:b7:33:5d:79:e5:6b:c9:29:73:34:01:69:ea:57:f0:01:be:
4e:f3:5c:f3:0a:a7:37:08:ad:18:9c:c7:4c:59:d0:5d:bb:01:
f1:53:76:cb:cd:d9:84:5e:bc:22:11:76:01:d9:e3:af:17:03:
01:ef:38:4c:ad:c1:7d:a9:c6:61:2b:ba:9c:81:95:86:af:bb:
73:90:dc:d9:2f:d1:3f:95:6a:b9:46:0f:fb:84:64:7c:7d:86:
65:aa:10:71:56:19:5f:60:52:7f:19:fa:d5:5a:e0:90:e4:b9:
62:55:71:2a:61:f9:37:2f:5e:07:71:43:cf:06:ca:6a:d5:52:
c8:33:e1:ad:b2:3e:a4:61:01:00:bc:55:5d:0a:f3:e6:4f:35:
06:c4:a8:3f:4c:8b:9b:c9:41:4b:f4:c1:57:ee:3c:c0:44:68:
52:5a:2d:b9:a7:f2:41:da:c4:8d:7d:db:40:b6:fc:47:63:5a:
69:a1:c7:8c:cc:3f:af:51:94:37:95:58:82:79:d2:16:4a:bf:
12:0b:59:a5:a5:11:71:e6:1c:63:3b:ea:f0:2f:10:e0:97:9a:
a1:04:53:d0:72:f4:3c:77:3b:78:ee:b5:aa:6b:f5:bb:5c:e9:
35:4f:69:65:87:29:24:ec:47:7b:78:5a:a7:c1:e5:f1:73:7d:
4d:79:ef:ef:4e:75:87:db:8f:36:fd:50:3e:74:dc:17:d4:c3:
3f:4f:82:24:51:1b:12:16:26:61:db:93:15:19:39:55:f5:05:
2c:6e:85:dd:b2:cc:4f:c0:09:0a:76:46:d8:e4:f2:11:92:a1:
e0:36:a8:25:c7:45:19:6c:98:eb:9a:fa:c1:ec:80:18:ce:d1:
f8:c4:23:9a:f9:b8:1f:05:67:8e:45:cb:e6:ee:0b:fa:db:67:
```

```
1f:62:2c:49:78:bb:55:98:1e:33:42:63:f2:db:ee:73:f7:60:
80:6d:5f:9a:e8:8c:89:39:5b:b2:84:e2:c3:99:77:f3:5f:19:
ec:b8:2b:ce:60:59:2c:66:06:f9:c1:43:b9:fd:94:35:9e:28:
9d:a0:8e:fd:0d:c6:1a:bb:20:93:b0:63:6a:83:2f:0a:db:c2:
b3:8e:b1:dd:f5:ab:19:09:53:7a:db:72:3f:1e:25:07:eb:1a:
7d:21:da:88:22:e6:f0:ba:b3:15:6f:95:f3:72:d2:cb:6d:48:
b8:ba:7b:aa:40:7f:81:fe:ba:15:c2:77:9d:86:58:bc:7d:89:
2e:7b:3a:96:04:9f:f1:3a:50:48:5a:25:4d:91:b6:ed:de:f6:
2e:4d:e5:77:11:6d:76:f4:23:5f:91:f0:0f:79:59:7a:f3:32:
24:11:c4:88:30:21:26:3b:f1:79:0f:04:06:ad:82:6d:ea:58:
4e:aa:4e:0a:7f:7b:5c:a5:ab:de:76:a9:a9:c7:d9:e3:eb:d6:
84:80:02:ab:da:4c:5b:49:90:29:c5:cb:5b:1c:06:61:e8:9a:
cf:a4:ea:9d:31:16:6a:21:3a:d9:22:25:b8:39:9d:4c:e3:86:
76:a8:dd:d8:b4:db:88:f9:5e:61:c3:1d:87:df:a9:31:33:7a:
b3:50:3e:f2:cd:ad:a0:9d:98:5f:6c:e2:f0:d8:27:b9:c2:37:
7f:8d:b4:f8:84:13:5f:22:6d:9b:81:bd:1c:e5:75:ae:b5:95:
d1:cb:d0:c6:e3:78:ec:8c:71:6d:8c:5d:40:79:7d:58:3d:5c:
63:77:cc:2e:a2:63:a9:71:30:2f:59:2a:ec:82:b1:e5:b9:d6:
bf:fb:21:e6:97:fc:70:45:9a:c7:e8:d2:81:73:b1:f5:bc:76:
ca:b4:be:9f:39:b5:2d:f2:3e:c5:32:e3:ae:3c:fd:74:a1:36:
5a:5c:4d:f6:de:d2:d5:66:61:74:88:2e:4b:69:7c:29:2f:e0:
2a:d6:d8:93:99:41:bc:7b:7f:fc:c3:1c:84:ed:16:c0:08:78:
fb:57:61:9e:83:7a:d1:e9:b7:ad:9a:85:1c:c3:ba:a3:e4:18:
b6:00:f6:35:27:e2:27:1d:10:dc:44:1d:11:05:a2:db:df:0a:
59:98:9c:f3:ca:3a:b3:26:2d:d1:c4:3c:fc:21:f3:3c:39:62:
7f:f4:bd:91:74:ef:02:83:da:4a:22:40:60:9f:6a:9f:8b:8f:
f1:e4:1e:99:d5:17:55:62:1c:60:01:7d:c7:41:db:19:9e:29:
01:ba:a0:5f:41:f3:61:ed:9d:0c:9c:ef:32:8b:b0:8a:89:b1:
e4:06:c9:2f:4d:42:2a:01:84:29:ac:f1:41:a0:a1:c9:b4:83:
d9:87:1a:53:1f:7f:d4:85:12:2e:79:f3:2c:88:06:73:62:ee:
16:bc:c7:8b:e7:09:96:ba:02:b5:56:ab:6f:c0:cf:76:64:62:
0e:1e:b5:e4:69:42:4d:ed:56:96:d9:1d:8d:07:40:7a:c5:bd:
d3:9f:43:07:e4:9d:b6:26:2b:33:6a:79:d9:8a:ec:ee:51:73:
f1:91:b0:e8:90:42:db:11:55:57:1b:01:10:fc:11:ff:77:b4:
09:01:6d:f8:8c:cf:72:16:df:09:12:09:bd:49:ef:33:b9:c5:
8d:35:60:77:80:8f:ee:98:18:be:bb:3a:61:e9:5b:6a:09:b0:
0a:1e:38:80:e9:71:46:77:a1:19:7a:c3:04:57:a5:77:e6:5a:
01:77:d2:92:90:f6:99:50:87:3f:30:8a:37:3d:37:1e:6b:1d:
a4:71:3c:6b:15:07:01:f6:3d:43:96:a3:f7:30:cf:08:2c:32:
a3:ca:67:6e:59:da:51:2e:96:bc:97:41:4b:7c:5f:97:a3:cf:
46:20:9e:64:96:08:f7:0c:03:4b:b4:83:09:db:6c:bb:94:23:
4e:ff:7b:fb:2f:84:66:0a:96:f9:e1:58:ff:0d:3c:84:62:9c:
6b:60:9f:7e:39:cf:33:f3:03:2f:c7:d0:8b:6f:f3:9a:62:cc:
33:c4:bd:b4:fc:b8:80:9d:fe:9e:c2:f0:d0:9e:07:71:a8:f9:
1f:a7:64:4d:63:f9:6b:ce:3e:44:0a:3f:05:58:90:0d:0c:20:
7d:4e:c7:52:d0:e5:b7:61:d3:6a:52:08:37:91:15:3c:cf:41:
ec:ef:88:56:dc:14:2a:12:55:cb:05:01:23:89:c0:fe:ca:de:
40:d2:d0:96:a3:1f:07:4a:58:96:fa:b2:ef:78:96:f0:73:25:
```

```
c8:2e:20:3b:d8:02:cf:e7:ca:b0:29:1a:25:7f:15:96:2d:fd:
52:bb:29:c3:fc:bf:b1:7c:d8:0f:76:21:05:28:2e:89:d9:82:
0e:cb:cd:03:1f:c3:71:b4:0f:75:52:e5:b4:93:8c:ac:ed:d5:
30:5a:b9:33:84:fd:3c:da:dc:e6:84:6d:c2:66:be:93:ad:67:
7f:db:d0:08:95:64:5a:2c:13:7f:e2:05:b5:dc:d0:bf:4d:6e:
93:c2:3b:8c:3b:b1:5c:3a:28:e8:c3:96:ed:59:e2:62:52:8e:
95:8d:b5:e1:c1:f2:34:5b:bf:5a:cc:f1:ee:ec:3d:6c:61:99:
f2:c8:e4:05:5f:ea:d5:74:3c:ff:df:1b:20:bd:35:30:c0:27:
f8:a4:6e:73:45:81:e2:b9:15:52:c7:a0:e7:c8:fd:7b:8e:f7:
d2:0c:c4:e9:22:69:4e:70:62:c7:8a:a2:a6:61:7c:0b:5a:74:
8d:0f:c0:e5:66:dc:18:7b:74:3b:72:ab:1a:53:b3:49:ef:50:
aa:76:80:e7:11:53:90:ab:24:d1:2e:fc:66:41:cf:b3:cc:ae:
ac:f9:eb:1e:19:f7:bc:54:00:16:da:b0:d4:2b:74:c7:35:fb:
08:ff:67:14:83:5a:eb:6b:b7:b4:63:28:e2:b6:b8:d4:0c:13:
6a:8c:bb:30:c1:fb:6c:42:df:23:c4:f0:be:25:df:2b:39:11:
bb:82:c3:e7:f9:04:48:77:cf:d0:5e:3d:6e:19:7f:b3:c4:2f:
c4:ec:51:5f:9d:c7:8f:88:9f:21:79:8d:a0:17:3e:17:73:b4:
f5:a2:71:70:e6:99:c4:fd:4c:f2:63:64:23:22:c3:72:71:52:
43:42:a5:90:e3:59:77:50:ff:a1:09:2e:c7:f6:7e:17:f2:a2:
d6:7e:2c:75:f2:ab:9e:36:78:ab:57:be:c5:91:71:70:2c:ba:
03:91:80:97:f4:9e:16:bc:fa:80:f4:22:2a:b5:75:15:57:d9:
b0:92:9e:b1:35:db:26:96:77:28:9c:89:99:db:9b:55:d4:29:
15:5f:54:8a:0d:58:a8:95:13:95:17:6c:6b:b0:2a:a3:fa:1a:
ec:2e:b4:0e:08:ea:8f:e1:8c:59:cf:7d:60:00:f3:bf:b7:e4:
5f:08:a6:02:ef:ce:d7:9c:8d:6f:56:d7:c9:35:e9:e5:cf:d2:
f5:28:ca:e6:36:ef:c4:26:52:d5:4d:04:ec:50:73:87:dc:70:
1f:1a:db:07:bf:4c:e9:ec:57:98:7f:bc:c8:31:9e:7e:e6:3a:
b4:c4:77:93:39:56:57:67:05:84:8d:03:02:d9:bf:04:6b:fe:
71:8a:be:b6:8a:ae:44:b0:dd:db:1f:6a:26:e5:50:d5:ff:03:
81:d8:1b:9f:3f:a6:bc:1b:52:b5:49:93:b0:27:fd:59:d4:7d:
69:e9:63:35:0b:9b:de:a1:d4:70:0c:08:41:4b:76:d6:cd:c8:
65:8c:bb:9a:6e:e4:f1:e2:30:13:9d:a3:c7:67:16:0f:7d:bd:
ac:dc:aa:9c:17:01:a6:27:14:fa:4a:c1:27:3f:07:7b:9f:2f:
47:56:cc:f0:96:38:e9:58:7c:1f:6c:73:10:3c:11:68:2a:3c:
5f:74:fe:37:ae:8b:e9:eb:c6:06:30:6f:62:3c:5c:6c:2d:c7:
5b:24:6d:cc:75:3f:d7:d4:e6:72:64:8a:ad:03:67:ad:cd:cb:
2d:7c:82:49:a9:ef:e8:b9:be:f2:6c:98:42:4e:26:46:04:58:
a5:2b:c9:88:9b:a4:91:7f:22:09:12:52:2a:d1:4e:36:22:d8:
53:bc:38:93:ad:11:19:c5:e7:c9:83:00:b4:b6:b0:ac:96:32:
ca:d0:08:69:e4:d2:29:86:74:74:49:be:4a:b2:bf:f2:2f:c2:
52:fd:15:3c:8d:07:12:3a:98:c7:49:67:81:1d:b1:5d:e8:f4:
42:79:a0:f7:44:b8:95:9f:e1:37:41:5b:c9:b1:89:90:7b:66:
96:eb:8e:dc:1b:d7:73:b2:eb:c1:42:41:e8:2d:28:ba:74:ea:
7c:77:87:76:5b:36:10:3d:87:08:52:94:e6:60:95:c1:1b:c9:
27:c1:42:aa:32:62:ed:ca:6f:04:4e:11:3a:3d:3d:e0:d8:3a:
c0:ff:b9:9a:94:b1:79:f3:01:14:3a:99:34:59:8e:d9:ac:f1:
a9:77:b5:2d:59:e1:29:96:1b:13:80:8b:10:94:3e:c2:51:db:
c1:24:06:02:47:96:9b:ae:5d:25:34:af:4b:65:f3:8a:eb:65:
```

```
7c:a5:5e:7c:a2:d6:1d:41:20:13:0b:5e:ea:67:b2:eb:bf:6c:
44:fb:76:31:58:5e:d2:33:6d:6f:9c:3a:41:70:34:11:6f:99:
8c:42:9d:d6:2b:14:79:b0:ac:d4:de:3a:b0:d8:d2:97:88:9a:
17:68:3e:79:a8:b0:4a:d7:a7:3c:63:c5:29:c1:65:76:74:7e:
c2:de:b8:49:ce:26:5f:d2:62:2d:0f:5c:cc:6c:53:c0:a4:75:
05:52:d1:52:38:ae:72:17:7c:02:67:6b:76:38:e7:72:aa:38:
70:5e:af:a2:98:c0:c1:7a:a0:6d:ec:90:51:8d:d5:99:8b:39:
05:6a:eb:0c:87:37:5b:4b:00:91:2c:7d:8a:6d:c1:23:10:44:
26:5a:47:f7:7f:8f:86:1c:c2:a7:9f:9e:48:f6:42:cd:d1:3c:
d9:e8:95:de:00:3c:ec:db:a1:a3:c0:7f:f7:17:3b:4a:dc:d2:
f5:d4:9b:12:19:0f:6d:13:38:72:06:21:eb:94:88:87:8f:a1:
de:f6:d7:a0:88:aa:e3:47:bb:69:e8:30:59:82:d2:3a:6d:c7:
26:95:92:a4:58:07:eb:db:a5:d1:bb:51:00:28:ef:6f:c8:ce:
9c:0f:d9:8d:e0:b3:14:db:90:dd:f9:26:af:b0:88:48:ae:22:
71:26:af:d5:e0:4d:5c:41:e6:0b:f2:5c:9b:bb:69:82:09:5a:
58:63:b9:0c:8a:22:37:aa:a2:71:2a:a5:d9:a7:7b:9f:d5:f4:
17:8d:bd:4e:de:08:6a:a4:20:ce:a6:85:c7:fa:05:c7:d8:03:
77:0c:dd:40:32:11:43:2a:8c:50:22:4b:fa:a1:d1:f1:94:42:
3f:d5:b8:a0:dd:01:71:6e:30:34:ff:a6:76:80:e6:c1:04:8b:
f0:c3:38:14:98:ae:eb:fd:05:98:d1:96:7e:b4:bf:51:ce:aa:
b4:66:71:30:9f:7a:45:b6:ed:d1:6e:8f:b0:6c:a5:f5:4f:ee:
bc:ea:65:5e:24:43:73:4b:50:8e:c8:68:0f:23:48:ed:dd:ff:
84:97:9b:31:0d:bb:2c:db:69:6b:0c:34:73:3e:ae:69:d2:f5:
be:a8:99:be:7b:40:82:f4:fe:35:f5:3d:a3:b1:b4:e2:6c:79:
b7:0b:29:ad:30:3d:56:9d:bc:24:e9:e6:a5:6d:cc:83:18:7b:
d5:98:a3:5f:dd:71:72:29:71:45:8f:41:52:ce:86:99:5c:f1:
40:0c:1e:b1:97:da:3a:14:4a:a7:02:48:d8:4e:63:12:99:da:
28:e9:de:0d:17:90:3a:f5:da:9a:01:7c:15:12:bf:00:48:7d:
63:8c:89:0b:b9:77:95:01:27:b2:33:73:4b:ab:a8:f3:24:ee:
c1:d3:0c:a3:9e:26:fe:24:23:3b:82:b4:1a:5e:72:dc:9e:91:
3a:7b:85:64:0d:30:2e:6b:55:53:7e:a2:4f:b7:10:e4:77:a1:
01:4a:b2:d7:7f:1c:94:a6:a7:e5:66:e2:c7:e5:37:6d:89:2c:
72:b1:53:cf:d6:67:0f:77:f8:bf:07:20:98:99:60:ef:2e:72:
c0:72:9e:79:2a:ca:a2:f7:bc:82:db:53:f7:68:e3:ed:4f:38:
64:83:1b:dd:a5:78:dc:db:08:a9:34:35:f6:f1:9c:76:85:5e:
cd:59:a3:c8:89:50:5b:bd:a0:64:06:b4:d7:db:7a:e1:75:57:
13:90:ce:05:4b:a0:f6:22:70:0b:78:a0:84:46:87:b4:a7:0d:
88:c6:41:c5:93:cb:77:37:d1:af:37:48:b9:47:db:99:7a:98:
36:82:cb:27:6a:9a:de:80:24:3a:29:eb:ab:bd:b0:40:0d:a6:
50:e5:a4:72:a3:19:cb:f3:52:8e:2f:1d:10:ef:7d:0a:15:6c:
49:08:53:55:84:85:5c:73:53:ce:3e:18:e5:04:92:a6:99:db:
4d:7b:c7:a9:99:ce:aa:90:48:73:7a:61:f5:92:73:da:b4:26:
74:a1:39:74:e3:82:f9:32:e0:08:ef:bc:2f:9f:6d:e1:da:3d:
f0:a5:46:b6:17:95:b8:6b:13:7d:f3:a1:31:8d:b7:47:a0:45:
aa:20:53:d6:f0:3c:eb:a2:e7:7a:26:8c:c6:c7:cb:0f:21:5a:
df:46:06:c5:b2:2d:a5:3b:b7:01:fd:0f:55:1b:5e:58:00:70:
94:a3:7f:48:8e:4a:67:a4:14:5d:e0:ba:b6:f9:9b:e7:de:61:
d8:67:83:ac:b7:01:eb:62:c5:22:b8:48:3a:96:55:fb:1a:4a:
```

```
          c4:63:30:f3:78:05:a6:ab:0c:e7:33:a0:88:f7:e2:e3:4a:1b:
          fd:66:3c:14:be:ee:20:d1:32:95:db:97:ff:d9:c2:bc:7a:c8:
          e4:ba:24:c5:b2:2e:16:f8:53:af:b4:57:56:25:26:f5:36:48:
          eb:0c:20:f9:3b:73:ff:dd:bd:20:81:0c:f5:55:89:7d:46:1b:
          05:b6:25:df:96:99:ea:09:79:60:72:d8:37:92:a8:f1:75:a3:
          5c:6d:54:b7:f3:32:17:35:1a:2d:96:e5:5e:fc:cd:54:30:49:
          af:6f:1a:42:d9:98:52:72:73:74:72:b7:72:95:80:1d:31:5a:
          e4:83:b7:b6:d4:14:00:0b:59:ce:7c:bc:1d:72:24:ab:74:d6:
          2c:9c:20:b1:0a:78:6f:a9:76:8d:6c:37:02:35:bd:6f:99:ee:
          d1:45:36:f1:34:60:7a:12:57:27:68:05:26:14:75:3c:9f:0d:
          3e:b7:5d:b8:2a:6c:1d:a7:b0:41:c4:f4:3d:ae:8e:51:54:37:
          65:ad:0a:c9:28:a0:3f:04:ed:54:59:c4:9f:1d:3d:70:97:5f:
          f9:44:53:ff:15:9f:03:13:7b:41:6b:c0:f7:8f:a3:27:2b:03:
          39:37:8f:bd:91:65:4d:74:a9:9f:45:6a:a4:25:dc:4c:f9:7e:
          59:fc:4e:93:7c:89:8f:71:8e:a6:99:66:5e:6a:25:a4:c0:a6:
          fa:25:f7:68:5c:8a:02:f5:7b:49:cd:89:e1:77:78:95:1b:a9:
          21:78:6e:f4:7a:e2:04:e5:0e:21:52:bf:04:cd:0c:69:5d:d7:
          f2:57:71:9f:d8:01:e0:f3:10:cc:15:2d:fd:99:78:ff:dc:1f:
          8f:a9:31:0d:0f:9f:f4:2c:a1:3d:4f:b2:51:92:68:f0:ec:d8:
          5f:c4:55:a1:4c:c8:12:e9:05:7e:05:93:5f:f9:76:99:85:18:
          29:24:60:14:5d:b3:79:f9:4b:7c:e4:22:71:8a:c2:66:45:d2:
          41:14:5d:59:4c:0a:b5:2b:ab:bd:c6:50:f8:87:37:42:e6:d4:
          96:72:cf:45:f0:d4:bf:0d:c5:17:9f:f1:b9:12:5c:a8:74:89:
          9e:56:07:cf:8f:98:9a:da:d7:db:7f:c7:d0:3a:0a:14:cd:5a:
          66:0c:eb:02:76:a0:d4:56:e6:e8:be:a1:f0:c7:23:b3:4f:86:
          90:1a:5a:16:8e:07:0d:24:d1:ee:03:98:9f
```

```
-----BEGIN CERTIFICATE-----
MIIU6zCCAXOgAwIBAgIUXCKtigZRnmcCai1DPovHI0N3gMgwCgYIKwYBBQUHBiMw
NzELMAkGA1UEBhMCRlIxDjAMBgNVBAcMBVBhcmlzMRgwFgYDVQQKDA9Cb2d1cyBY
TVNTTVQgQ0EwHhcNMjQwNzEwMDgyODA0WhcNMzQwNzA4MDgyODA0WjA3MQswCQYD
VQQGEwJGUjEOMAwGA1UEBwwFUGFyaXMxGDAWBgNVBAoMD0JvZ3VzIFhNU1NNVCBD
QTBTMAoGCCsGAQUFBwYjA0UAAAAAAUuniRFv/B370+dxc7iiSO9TuZ0fxop8vk+K
KfpB/b3aIH/2O7DFuKfC8lryJhTrNvAmL4d0+w7Vfheg0U22z1GjYzBhMB0GA1Ud
DgQWBBR8fVm4lWHVA2oePfEkqx3tBM3bXzAfBgNVHSMEGDAWgBR8fVm4lWHVA2oe
PfEkqx3tBM3bXzAPBgNVHRMBAf8EBTADAQH/MA4GA1UdDwEB/wQEAwIBBjAKBggr
BgEFBQcGIwOCE2QAAAAAV8SYif/ZCo5ubxaVjOw1QiHCylbt+IHxsk8rbXP0N1X8
9E4V62uQ3jT+1pZwlI3B50oySTA6QKRn0vva+NihekgiHOOYvNBohSnJ5fdcVtic
gL5o7RHrOQ/vywmyKDCmKwW83hEivsTcCJo9tEk3H1ReXy2TYrCVxV0jkvNVQHgZ
AFaeovEOS6511pIJsXnsyRhnGQmGg3RdCgar2vCvApdN13MGi6KExwmv3YsVOeQw
n8kAJagzTd7oJbY1C1G/ejSn6ITo+jlbqjdulYmsJkpOyr4pCEs8KKeFaq1a0pPr
EuGahxxAO88VbENOiCFUUn4NbRcpjRVv70JaqSXQl4BhMSKknyUXUa0LocuTtPWm
sCIbbVBkKki9BRaIAON7VtADs3otagvz3qKMboGALI/p2HjtW5nJE9G263jDQCuh
eoQKuhKHXh04JCKPwKNlHBzOLY7lLx++k1z+HM2onX5+zxjinMVU3GJhdCNVZGYh
lkynLoqUpjUQpehebpGsqMvtUStmRQP1h+1NjE5tVIChM4qEnSMxkMYFEaedvVEK
c0e8CEkRs5j/ARRp18CgDFXkXuL6hKwns4UsmXFSnDP4nYzSE7xuGHkVpwLuFesn
2K8kOAKcyjDz4jBBL2KiLKWBG3FtsZS9xj2eXlFF3lv01+Y159h81Zjsfg74ncGn
e7NlsaFLLezZEkVrHwscazsKZnY59Myb4bcX91P8w6YY9y5FUrEYmXXRabt3yBqE
```

Xwa1i8sCsLIPvxcYZT2ncltxn5J+Ot+EzGVcxFtw/cw4nhJu+f8fAvzK9WiG/Mpx
8T17MrTUw6IgFj8SB3GVO9SxHvyMHzSMyKuMu3WTwRrShT6a5gSGiN4nRsrz9/OO
VBjqqq4UArFKauAkd0AojTcnnIdqgQnSAU0gf96EqICMjmOCvmbfhzBcuHEK6ZFo
cW6Xl/AnTvquaoWsgM04SEnBK53bVMXwv/oG6JY6wJXwiL2OgHg93K1dClbdx4Cf
/GRYTW0n9tcajLIcCep9T3SZDUoMuLDvdN1vb9zlg+HjwuhYF7hEii3s31T2H2ei
s8UZ+7nHGzzqvSzhQ2XRWhfck53FhQxVNBNJFZLiUhTRgapiAhq6ybBThY570U40
dqx517NIkr9Vfi1czTKbwUGno823lFyWHj4nTevwYUuk4zy7aYU36ZyY9Gh6YXeM
vbkw1vH9aXg/lpl7aTmQs3y2iO3NGdpCZOUyTKIw98ToJ5Nw7fpeyo560ROvFbFZ
yZuRYQsG1cwugLtJk92+U4i+r4BkfF6+e4vnXzmvq2dCawaq79Zpr6kAH6AVEAQ+
25OyN9vrhVlDoo2PBozLoh2oPJ/0pHzIzf/wqHkP59iUZ+wXP/puBAdPv4YEbPxG
h7UQhaQH6K+p7F0oXICMMczHs4EXC0t9HJ50Ah7v3g0bwcAETUb93AukxjPmhQpg
OU0L+UlEM+AVmRm/x4rGlgSTN2td6L5z1IC4gQ+akUTPcgLTyfjgfdKbK//rQm44
ftzNp5DFLCugIze5ZBCmJ2hHxfHojUHBSeg1SM7ICEyt8q1d6WLryTxhhRjGNHP9
JqTwUIObZFSqVWzYoiGB/5wnOR/Dog7lU7HX+h/vKYvCkJjqLt1Fv8Nso5NHmQMY
Jeil7i5363/0SUlZmMH8qx6tIL34JP0hG9paB1XIUAUxUJOy+G7bc01fNKrzNIOQ
8EFtyENW0XUH9RYgs5myxzQlxA50WlEPezt/aqlBF7VHYi1PuWGXYOmuyq0xbksK
R5xTZqNOw5Z8AaCOroNFQuaSEo6Xb+igt32mdCSqILD6npjofLTaMOmUCJa3uVNP
dV8MTYLjz268+iNP+jMXfJi2HkeJPtmhqkIZJa6eP1NErJGW2FXDQB36rYY4Yr0n
LyY0vq2aAURCyFSlOukK//hBbTge4j0IOpRPHmDQscKOlDTwMD7wkSXumDS0jZVO
z+0dYYnJWRBo8rwuXL3ADx2cL3zAJyUUm96jdGQoFCyispA6pGpQ6Y7KeOW2dFbg
kml9tC7g52aSFpKgw9tP09BXTUoo7rfMBO8X2fwBux6yWwI9H1qFc6GBlrczXXnl
a8kpczQBaepX8AG+TvNc8wqnNwitGJzHTFnQXbsB8VN2y83ZhF68IhF2AdnjrxcD
Ae84TK3BfanGYSu6nIGVhq+7c5Dc2S/RP5VquUYP+4RkfH2GZaoQcVYZX2BSfxn6
1VrgkOS5YlVxKmH5Ny9eB3FDzwbKatVSyDPhrbI+pGEBALxVXQrz5k81BsSoP0yL
m8lBS/TBV+48wERoUlotuafyQdrEjX3bQLb8R2NaaaHHjMw/r1GUN5VYgnnSFkq/
EgtZpaURceYcYzvq8C8Q4JeaoQRT0HL0PHc7eO61qmv1u1zpNU9pZYcpJOxHe3ha
p8Hl8XN9TXnv7051h9uPNv1QPnTcF9TDP0+CJFEbEhYmYduTFRk5VfUFLG6F3bLM
T8AJCnZG2OTyEZKh4DaoJcdFGWyY65r6weyAGM7R+MQjmvm4HwVnjkXL5u4L+ttn
H2IsSXi7VZgeM0Jj8tvuc/dggG1fmuiMiTlbsoTiw5l3818Z7LgrzmBZLGYG+cFD
uf2UNZ4onaCO/Q3GGrsgk7BjaoMvCtvCs46x3fWrGQlTettyPx4lB+safSHaiCLm
8LqzFW+V83LSy21IuLp7qkB/gf66FcJ3nYZYvH2JLns6lgSf8TpQSFolTZG27d72
Lk3ldxFtdvQjX5HwD3lZevMyJBHEiDAhJjvxeQ8EBq2CbepYTqpOCn97XKWr3nap
qcfZ4+vWhIACq9pMW0mQKcXLWxwGYeiaz6TqnTEWaiE62SIluDmdTOOGdqjd2LTb
iPleYcMdh9+pMTN6s1A+8s2toJ2YX2zi8NgnucI3f420+IQTXyJtm4G9HOV1rrWV
0cvQxuN47IxxbYxdQHl9WD1cY3fMLqJjqXEwL1kq7IKx5bnWv/sh5pf8cEWax+jS
gXOx9bx2yrS+nzm1LfI+xTLjrjz9dKE2WlxN9t7S1WZhdIguS2l8KS/gKtbYk5lB
vHt//MMchO0WwAh4+1dhnoN60em3rZqFHMO6o+QYtgD2NSfiJx0Q3EQdEQWi298K
WZic88o6syYt0cQ8/CHzPDlif/S9kXTvAoPaSiJAYJ9qn4uP8eQemdUXVWIcYAF9
x0HbGZ4pAbqgX0HzYe2dDJzvMouwiomx5AbJL01CKgGEKazxQaChybSD2YcaUx9/
1IUSLnnzLIgGc2LuFrzHi+cJlroCtVarb8DPdmRiDh615GlCTe1WltkdjQdAesW9
059DB+SdtiYrM2p52Yrs7lFz8ZGw6JBC2xFVVxsBEPwR/3e0CQFt+IzPchbfCRIJ
vUnvM7nFjTVgd4CP7pgYvrs6YelbagmwCh44gOlxRnehGXrDBFeld+ZaAXfSkpD2
mVCHPzCKNz03HmsdpHE8axUHAfY9Q5aj9zDPCCwyo8pnblnaUS6WvJdBS3xfl6PP
RiCeZJYI9wwDS7SDCdtsu5QjTv97+y+EZgqW+eFY/w08hGKca2CffjnPM/MDL8fQ
i2/zmmLMM8S9tPy4gJ3+nsLw0J4Hcaj5H6dkTWP5a84+RAo/BViQDQwgfU7HUtDl
t2HTalIIN5EVPM9B7O+IVtwUKhJVywUBI4nA/sreQNLQlqMfB0pYlvqy73iW8HMl
yC4gO9gCz+fKsCkaJX8Vli39Urspw/y/sXzYD3YhBSguidmCDsvNAx/DcbQPdVLl
tJOMrO3VMFq5M4T9PNrc5oRtwma+k61nf9vQCJVkWiwTf+IFtdzQv01uk8I7jDux

XDoo6MOW7VniYlKOlY214cHyNFu/Wszx7uw9bGGZ8sjkBV/q1XQ8/98bIL01MMAn
+KRuc0WB4rkVUseg58j9e4730gzE6SJpTnBix4qipmF8C1p0jQ/A5WbcGHt0O3Kr
GlOzSe9QqnaA5xFTkKsk0S78ZkHPs8yurPnrHhn3vFQAFtqw1Ct0xzX7CP9nFINa
62u3tGMo4ra41AwTaoy7MMH7bELfI8TwviXfKzkRu4LD5/kESHfP0F49bhl/s8Qv
xOxRX53Hj4ifIXmNoBc+F3O09aJxcOaZxP1M8mNkIyLDcnFSQ0KlkONZd1D/oQku
x/Z+F/Ki1n4sdfKrnjZ4q1e+xZFxcCy6A5GAl/SeFrz6gPQiKrV1FVfZsJKesTXb
JpZ3KJyJmdubVdQpFV9Uig1YqJUTlRdsa7Aqo/oa7C60Dgjqj+GMWc99YADzv7fk
XwimAu/O15yNb1bXyTXp5c/S9SjK5jbvxCZS1U0E7FBzh9xwHxrbB79M6exXmH+8
yDGefuY6tMR3kzlWV2cFhI0DAtm/BGv+cYq+toquRLDd2x9qJuVQ1f8Dgdgbnz+m
vBtStUmTsCf9WdR9aeljNQub3qHUcAwIQUt21s3IZYy7mm7k8eIwE52jx2cWD329
rNyqnBcBpicU+krBJz8He58vR1bM8JY46Vh8H2xzEDwRaCo8X3T+N66L6evGBjBv
YjxcbC3HWyRtzHU/19TmcmSKrQNnrc3LLXyCSanv6Lm+8myYQk4mRgRYpSvJiJuk
kX8iCRJSKtFONiLYU7w4k60RGcXnyYMAtLawrJYyytAIaeTSKYZ0dEm+SrK/8i/C
Uv0VPI0HEjqYx0lngR2xXej0Qnmg90S41Z/hN0FbybGJkHtmluuO3BvXc7LrwUJB
6C0ounTqfHeHdls2ED2HCFKU5mCVwRvJJ8FCqjJi7cpvBE4ROj094Ng6wP+5mpSx
efMBFDqZNFmO2azxqXe1LVnhKZYbE4CLEJQ+wlHbwSQGAkeWm65dJTSvS2Xziutl
fKVefKLWHUEgEwte6mey679sRPt2MVhe0jNtb5w6QXA0EW+ZjEKd1isUebCs1N46
sNjSl4iaF2g+eaiwStenPGPFKcFldnR+wt64Sc4mX9JiLQ9czGxTwKR1BVLRUjiu
chd8Amdrdjjncqo4cF6vopjAwXqgbeyQUY3VmYs5BWrrDIc3W0sAkSx9im3BIxBE
JlpH93+PhhzCp5+eSPZCzdE82eiV3gA87Nuho8B/9xc7StzS9dSbEhkPbRM4cgYh
65SIh4+h3vbXoIiq40e7aegwWYLSOm3HJpWSpFgH69ul0btRACjvb8jOnA/ZjeCz
FNuQ3fkmr7CISK4icSav1eBNXEHmC/Jcm7tpgglaWGO5DIoiN6qicSql2ad7n9X0
F429Tt4IaqQgzqaFx/oFx9gDdwzdQDIRQyqMUCJL+qHR8ZRCP9W4oN0BcW4wNP+m
doDmwQSL8MM4FJiu6/0FmNGWfrS/Uc6qtGZxMJ96Rbbt0W6PsGyl9U/uvOplXiRD
c0tQjshoDyNI7d3/hJebMQ27LNtpaww0cz6uadL1vqiZvntAgvT+NfU9o7G04mx5
twsprTA9Vp28JOnmpW3Mgxh71ZijX91xcilxRY9BUs6GmVzxQAwesZfaOhRKpwJI
2E5jEpnaKOneDReQOvXamgF8FRK/AEh9Y4yJC7l3lQEnsjNzS6uo8yTuwdMMo54m
/iQjO4K0Gl5y3J6ROnuFZA0wLmtVU36iT7cQ5HehAUqy138clKan5Wbix+U3bYks
crFTz9ZnD3f4vwcgmJlg7y5ywHKeeSrKove8gttT92jj7U84ZIMb3aV43NsIqTQ1
9vGcdoVezVmjyIlQW72gZAa019t64XVXE5DOBUug9iJwC3ighEaHtKcNiMZBxZPL
dzfRrzdIuUfbmXqYNoLLJ2qa3oAkOinrq72wQA2mUOWkcqMZy/NSji8dEO99ChVs
SQhTVYSFXHNTzj4Y5QSSppnbTXvHqZnOqpBIc3ph9ZJz2rQmdKE5dOOC+TLgCO+8
L59t4do98KVGtheVuGsTffOhMY23R6BFqiBT1vA866LneiaMxsfLDyFa30YGxbIt
pTu3Af0PVRteWABwlKN/SI5KZ6QUXeC6tvmb595h2GeDrLcB62LFIrhIOpZV+xpK
xGMw83gFpqsM5zOgiPfi40ob/WY8FL7uINEylduX/9nCvHrI5LokxbIuFvhTr7RX
ViUm9TZI6wwg+Ttz/929IIEM9VWJfUYbBbYl35aZ6gl5YHLYN5Ko8XWjXG1Ut/My
FzUaLZblXvzNVDBJr28aQtmYUnJzdHK3cpWAHTFa5IO3ttQUAAtZzny8HXIkq3TW
LJwgsQp4b6l2jWw3AjW9b5nu0UU28TRgehJXJ2gFJhR1PJ8NPrdduCpsHaewQcT0
Pa6OUVQ3Za0KySigPwTtVFnEnx09cJdf+URT/xWfAxN7QWvA94+jJysDOTePvZFl
TXSpn0VqpCXcTPl+WfxOk3yJj3GOpplmXmolpMCm+iX3aFyKAvV7Sc2J4Xd4lRup
IXhu9HriBOUOIVK/BM0MaV3X8ldxn9gB4PMQzBUt/Zl4/9wfj6kxDQ+f9CyhPU+y
UZJo8OzYX8RVoUzIEukFfgWTX/l2mYUYKSRgFF2zeflLfOQicYrCZkXSQRRdWUwK
tSurvcZQ+Ic3QubUlnLPRfDUvw3FF5/xuRJcqHSJnlYHz4+YmtrX23/H0DoKFM1a
ZgzrAnag1Fbm6L6h8Mcjs0+GkBpaFo4HDSTR7gOYnw==
-----END CERTIFICATE-----

Acknowledgments

Thanks for Russ Housley, Panos Kampanakis, Michael StJohns and Corey Bonnell for helpful suggestions and reviews.

This document uses a lot of text from similar documents [SP800208], ([RFC3279] and [RFC8410]) as well as [I-D.ietf-lamps-rfc8708bis]. Thanks go to the authors of those documents. "Copying always makes things easier and less error prone" - [RFC8411].

Authors' Addresses

Daniel Van Geest
CryptoNext Security
Email: daniel.vangeest@cryptonext-security.com

Kaveh Bashiri
BSI
Email: kaveh.bashiri.ietf@gmail.com

Scott Fluhrer
Cisco Systems
Email: sfluhrer@cisco.com

Stefan-Lukas Gazdag
genua GmbH
Email: ietf@gazdag.de

Stavros Kousidis
BSI
Email: kousidis.ietf@gmail.com

MPLS Working Group
Internet-Draft
Updates: 4928 (if approved)
Intended status: Standards Track
Expires: 16 May 2025

K. Kompella
Juniper Networks
S. Bryant
University of Surrey 5GIC
M. Bocci
Nokia
G. Mirsky, Ed.
Ericsson
L. Andersson
J. Dong
Huawei Technologies

12 November 2024

IANA Registry for the First Nibble Following a Label Stack
draft-ietf-mpls-1stnibble-11

Abstract

This memo creates a new IANA registry (called the Post-stack First Nibble registry) for the first nibble (4-bit field) immediately following an MPLS label stack. The memo offers a rationale for such a registry, describes how the registry should be managed, and provides some initial entries. Furthermore, this memo sets out some documentation requirements for registering new values. Finally, it provides some recommendations that make processing MPLS packets easier and more robust.

The relationship between the IANA IP Version Numbers (RFC 2780) and the Post-stack First Nibble registry is described in this document.

This document updates RFC 4928 by deprecating the heuristic method for identifying the type of packet encapsulated in MPLS.

This Internet-Draft will expire on 16 May 2025.

Table of Contents

1. Introduction

An MPLS packet consists of a label stack, an optional "post-stack header" (PSH) and an optional embedded packet (in that order). By PSH, we mean existing artifacts such as Control Words [RFC4385], BIER (Bit-Indexed Explicit Replication) headers [RFC8296] and the like, as well as new types of PSH being discussed by the MPLS Working Group. However, in the data plane, there are scant clues regarding the PSH,

and no clue as to the type of embedded packet; this information is communicated via other means, such as the routing protocols that signal the labels in the stack. Nonetheless, in order to better handle an MPLS packet in the data plane, it is common practice for network equipment to "guess" the type of embedded packet. Such equipment may also need to process the PSH. Both of these require parsing the data after the label stack. To do this, the "first nibble" (the top four bits of the first octet following the label stack) is often used. Although some existing network devices may use such a method, it needs to be stressed that the correct interpretation of the Post-stack First Nibble (PFN) in a PSH can be made only in the context associated using the control or management plane with the Label Stack Element (LSE) or group of LSEs in the preceding label stack that characterize the type of the PSH, and that any attempt to rely on the value in any other context is unreliable.

The semantics and usage of the first nibble are not well documented, nor are the assignments of values. This memo serves four purposes:

* To document the values already in use.

* To provide a mechanism to document future assignments through the creation of a new IANA "Post-stack First Nibble registry", and document the relationship between it and the IANA IP Version Numbers [RFC2780].

* Provide a method for tracking usage by requiring more detailed documentation.

* To stress the importance that any MPLS packet not carrying plain IPv4 or IPv6 packets contains a PSH, including any new version of IP (Section 2.4).

Based on the analysis of load-balancing techniques in Section 2.1.1, this memo, in Section 2.1.1.1, introduces a requirement that deprecates the use of the heuristic and recommends using a dedicated label value for load balancing. The intent of both is for legacy routers to continue operating as they have, with no new problems introduced as a result of this memo. However, new implementations that follow this memo enable a more robust network operation.

Furthermore, this document updates [RFC4928] by deprecating the heuristic method for identifying the type of packet encapsulated in MPLS. This document clearly states that the type of encapsulated packet cannot be determined based on the PFN alone.

1.1. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

MPLS packet: one whose Layer 2 header declares the type to be MPLS. For Ethernet, that means the Ethertype is 0x8847 or 0x8848.

Label Stack: (of an MPLS packet) all labels (four-octet fields) after the Layer 2 header, up to and including the label with the Bottom of Stack bit set ([RFC3032]).

Post-stack First Nibble (PFN): the most significant four bits of the first octet following the label stack.

MPLS Payload: all data after the label stack, including the PFN, an optional post-stack header, and the embedded packet.

Post-stack Header (PSH): optional field of interest to the egress Label Switching Router (LSR) (and possibly to transit LSRs). Examples include a control word [RFC4385], [RFC8964] or an associated channel [RFC4385], [RFC5586], [RFC9546]. The PSH MUST indicate its length, so that a parser knows where the embedded packet starts.

Embedded Packet: an embedded packet follows immediately after the MPLS Label Stack and an optional PSH. That could be an IPv4 or IPv6 packet, an Ethernet packet (for VPLS ([RFC4761], [RFC4762]) or EVPN [RFC7432]), or some other type of Layer 2 frame [RFC4446].

Deprecation: regardless of how the deprecation is understood in other IETF documents, the interpretation in this document is that if a practice has been deprecated, that practice should not be included in new implementations or deployed in new deployments.

1.2. Reference Figures

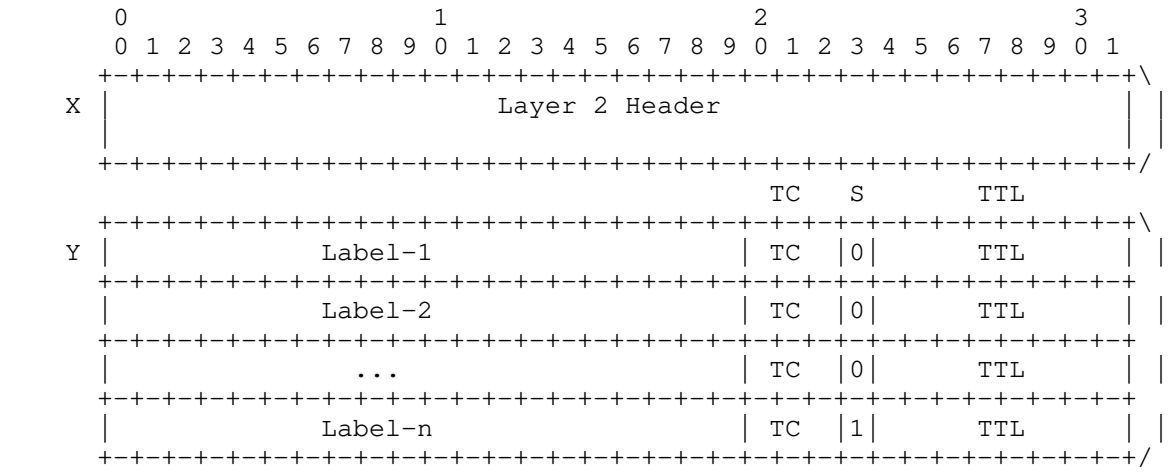


Figure 1: Example of an MPLS Packet With Label Stack

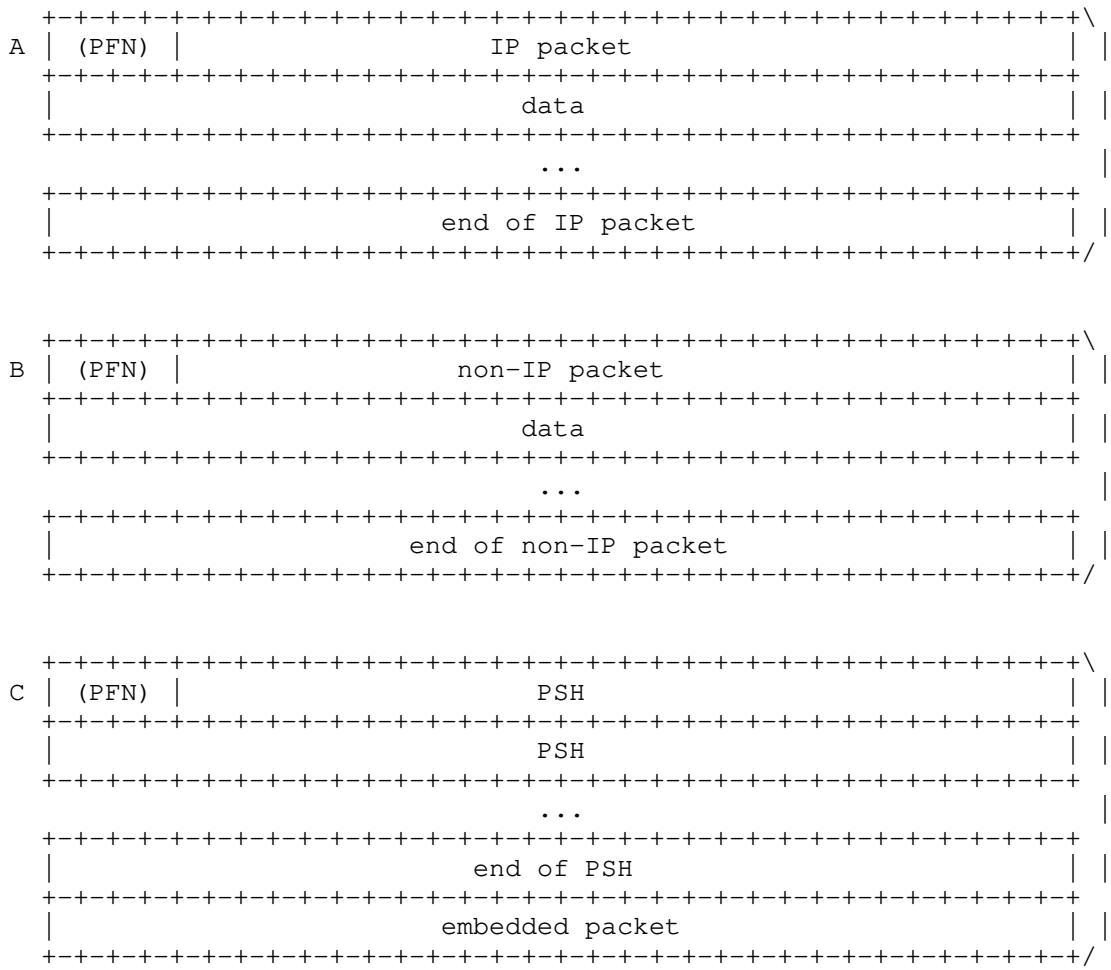


Figure 2: Examples of an MPLS Packet Payload With and Without Post-Stack Header

Figure 1 shows an MPLS packet with Layer 2 header X and a label stack Y ending with Label-n. Then, there are three examples of an MPLS payload displayed in Figure 2. The complete MPLS packet thus would consist of [X Y A], or [X Y B], or [X Y C].

A. The first payload is a bare IP packet, i.e., no PSH. The PFN in this case overlaps with the IP version number.

B. The next payload is a bare non-IP packet; again, no PSH. The PFN here is the first nibble of the payload, whatever it happens to be.

C. The last example is an MPLS Payload that starts with a PSH followed by the embedded packet. Here, the embedded packet could be IP or non-IP.

1.3. Abbreviations

LSR: Label Switching Router

LSE: Label Stack Element

PSH: Post-Stack Header

PFN: Post-stack First Nibble

FAT: Flow-Aware Transport

SPL: Special Purpose Label

PW: Pseudowire

MNA: MPLS Network Action

BIER: Bit-Indexed Explicit Replication

2. Rationale

2.1. Why Look at the First Nibble

An MPLS packet can contain one of many types of embedded packets. Three common types are:

1. An IPv4 packet (whose IP header has version number 4).

2. An IPv6 packet (whose IP header has version number 6).

3. A Layer 2 Ethernet frame (i.e., not including the Preamble or the Start frame delimiter), starting with the destination MAC address.

Many other packet types are possible; in principle, any Layer 2 embedded packet is permissible. Indeed, in the past, packets of Point-to-Point Protocol, Frame Relay, and Asynchronous Transfer Mode protocols were reasonably common.

In addition, there may be a PSH ahead of the embedded packet. The value of PFN is considered to ensure that the PSH can be correctly parsed. If no PSH follows the label stack, then the value of PFN indicates the version number of the IP packet header.

2.1.1. Load Balancing

There are four common ways to load balance an MPLS packet:

1. One can use the top label alone.

2. One can do better by using all of the non-SPLs (Special Purpose Labels) [RFC7274] in the stack.

3. One can do even better by "divining" the type of embedded packet, and using fields from the guessed header. The ramifications of using this load-balancing technique are discussed in detail in Section 2.1.1.1.

4. One can do best by using either an Entropy Label [RFC6790] or a Flow-Aware Transport (FAT) Pseudowire Label [RFC6391] (see Section 2.1.1.1).

Load balancing based on just the top label means that all packets with that top label will go the same way -- this is far from ideal. Load balancing based on the entire label stack (not including SPLs) is better, but it may still be uneven. If, however, the embedded packet is an IP packet, then the combination of (<source IP address>, <dest IP address>, <transport protocol>, <source port>, and <dest port>) from the IP header of the embedded packet forms an excellent basis for load-balancing. This is what is typically used for load balancing IP packets.

An MPLS packet doesn't, however, carry a payload type identifier. There is a simple (but risky) heuristic that is commonly used to guess the type of the embedded packet. The first nibble, i.e., the four most significant bits of the first octet, of an IP header contains the IP version number. That, in turn, indicates where to find the relevant fields for load-balancing. The heuristic goes roughly as described in Section 2.1.1.1.

2.1.1.1. Heuristic for Load Balancing

1. If the PFN is 0x4 (0100b), treat the payload as an IPv4 packet, and find the relevant fields for load-balancing on that basis.

2. If the PFN is 0x6 (0110b), treat the payload as an IPv6 packet, and find the relevant fields for load-balancing on that basis.

3. If the PFN is anything else, the MPLS payload is not an IP packet; fall back to load-balancing using the label stack.

This heuristic has been implemented in many (legacy) routers, and performs well in the case of Figure 2, A. However, this heuristic can work very badly for Figure 2, B. For example, if payload B is an Ethernet frame, then the PFN is the first nibble of the Organizationally Unique Identifier of the destination MAC address, which can be 0x4 or 0x6, and if so would lead to the packet being treated as an IPv4 or IPv6 packet such that data at the offsets of specific relevant fields would be used as input to the load-balancing heuristic resulting in unpredictable load balancing. This behavior can happen to other types of non-IP payloads as well.

That, in turn, led to the idea of inserting a PSH (e.g., a pseudowire control word [RFC4385], a DetNet control word [RFC8964], a Network Service Header [RFC8300], or a BIER header [RFC8296]) where the PFN is not 0x4 or 0x6, to explicitly prevent forwarding engines from confusing the MPLS payload with an IP packet. [RFC8469] recommends the use of a control word when the embedded packet is an Ethernet frame. RFC 8469 was published at the request of the operator community and the IEEE Registration Authority Committee as a result of operational difficulties with pseudowires that did not contain the control word.

It is RECOMMENDED that where load-balancing of MPLS packets is desired, the load-balancing mechanism uses the value of a dedicated label, for example, either an Entropy Label [RFC6790] or a FAT Pseudowire Label [RFC6391]. Furthermore, the heuristic of guessing the type of the embedded packet, as discussed above, SHOULD NOT be used.

A consequence of the heuristic approach is that while legacy routers may look for a PFN of 0x4 [RFC0791] or 0x6 [RFC8200], no legacy router will look for any other PFN, regardless of what future IP version numbers will be, for load-balancing purposes. This means that the values 0x4 and 0x6 are used to (sometimes incorrectly) identify IPv4 and IPv6 packets, but no other of PFN values will be used to identify IP packets.

This document creates a new PFN Registry for all 16 possible values.

2.2. Updates of RFC 4928

Paragraph 3 in Section 3 of RFC 4928 [RFC4928] states that:

OLD TEXT

> It is REQUIRED, however, that applications dependent upon in-order packet delivery restrict the first nibble values to 0x0 and 0x1. This will ensure that their traffic flows will not be affected if some future routing equipment does similar snooping on some future version(s) of IP.

END

The text in RFC 4928 [RFC4928] concerning the first nibble after the MPLS Label Stack has been updated by this document and the heuristic for snooping this nibble has been deprecated. RFC 4928 is now updated as follows:

NEW TEXT:

> Network equipment MUST use a PSH (Post-Stack Header) with a PFN (Post-stack First Nibble) value that is neither 0x4 nor 0x6 in all cases when the MPLS payload is not an IP packet.

END

The recommendation (see Section 2.1.1.1) replaces the paragraph 4 in Section 3 of RFC 4928 [RFC4928] as follows:

OLD TEXT:

> This behavior implies that if in the future an IP version is defined with a version number of 0x0 or 0x1, then equipment complying with this BCP would be unable to look past one or more MPLS headers, and load-split traffic from a single LSP across multiple paths based on a hash of specific fields in the IPv0 or IPv1 headers. That is, IP traffic employing these version numbers would be safe from disturbances caused by inappropriate load-splitting, but would also not be able to get the performance benefits.

NEW TEXT:

> The practice of deducing the payload type based on the PFN value is deprecated to avoid inaccurate load balancing. This means that older implementations and deployments can continue to use that heuristic, while it must not be part of new implementations or deployments. It also means that concerns about load balancing for future IP versions with a version number of 0x0 or 0x1 are no longer relevant.

END

Furthermore, the following text is appended to Section 1.1 of RFC 4928 [RFC4928]:

NEW TEXT:

> PSH: Post-Stack Header
>
> PFN: Post-stack First Nibble

END

2.3. Why Create a Registry

Support for MPLS Network Actions (MNAs) is described in [I-D.ietf-mpls-mna-fwk] and is an enhancement to the MPLS architecture. The use of post-stack data (PSD) to encode the MNA indicators and ancillary data is described in section 3.6 might place data in the PFN that could conflict with other uses of that nibble. This issue is described in section 3.6.1 of [I-D.ietf-mpls-mna-fwk] and is further illustrated by the PFN value of 0x0 which has two different formats depending on whether the PSH is a pseudowire control word or a DetNet control word: disambiguation requires the context of the service label.

With a registry, PSHs become easier to parse; not needing means outside the data plane to interpret them correctly; and their semantics and usage are documented.

2.4. IP Version Numbers versus Post-stack First Nibble Values

The use of the PFN stemmed from the desire to heuristically identify IP packets for load-balancing purposes. It was then discovered that non-IP packets, misidentified as IP when the heuristic failed, were being badly load balanced, leading to [RFC4928]. This situation may confuse some as to the relationship between the Post-stack First Nibble Registry and the IP Version Numbers registry. These registries are quite different:

1. The IP Version Numbers registry's explicit purpose is to track IP version numbers in an IP header.

2. The Post-stack First Nibble registry's purpose is to track PSH types.

The only intersection points between the two registries is for values 0x4 and 0x6 (for backward compatibility). There is no need to track future IP version number allocations in the Post-stack First Nibble registry.

2.5. Next Step to More Deterministic Load -balancing in an MPLS Network

Network evolution is impossible to control, but it develops over a period of time determined by various factors. This document prevents further proliferation of the implementations that could lead to undesired effects affecting data flow. At some time in the future, it was planned to obsolete MPLS encapsulations without PSH of non-IP payload. Before that it is paramount to collect sufficient evidence that there are no marketed or deployed implementations using the heuristic practice to load-balancing MPLS data flows.

3. IANA Considerations

3.1. The Post-stack First Nibble Registry

This memo requests IANA to create a registry group called "Post-Stack First Nibble Registry" that consists of a single registry called "Post-Stack First Nibble Registry". The registry should be created as shown in Table 1. The assignment policy for the registry is Standards Action [RFC8126]. It is important to note, that the same PFN value can be used in more than one protocol. The correct interpretation of the PFN in a PSH can be made only in the context of the LSE or a group of LSEs in the preceding label stack that characterize the type of the PSH and, consequently, PFN.

Protocol	Value	Description	Reference
DetNet	0x0	DetNet Control Word	RFC 8964
NSH	0x0	NSH (Network Service Header) Base Header, payload	RFC 8300
PW	0x0	PW Control Word	RFC 4385
DetNet	0x1	DetNet Associated Channel	RFC 9546
MPLS	0x1	MPLS Generic Associated Channel	RFC 5586
PW	0x1	PW Associated Channel	RFC 4385
NSH	0x2	NSH Base Header, OAM	RFC 8300
	0x3	Unassigned	
	0x4	Reserved, not to be assigned	
BIER	0x5	BIER Header	RFC 8296
	0x6	Reserved, not to be assigned	
	0x7 - 0xF	Unassigned	

Table 1: Post-stack First Nibble Values

4. Security Considerations

This document creates a new IANA registry for and specifies changes to the treatment in the data plane of packets based on the first nibble of data beyond the MPLS label stack. One intent of this is to reduce or eliminate errors in determining whether a packet being transported by MPLS is IP or not. While such errors have primarily caused unbalanced and, thus, inefficient multi-pathing, they have the potential to cause more severe security problems.

For general MPLS label stack security considerations, see [RFC3032].

5. Acknowledgements

The authors express their appreciation and gratitude to Donald E. Eastlake 3rd for the review, insightful questions, and helpful comments. Also, the authors are gateful to Amanda Baber for helping organize the IANA registry in clear and consise manner.

6. References

6.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[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>.

[RFC2780] Bradner, S. and V. Paxson, "IANA Allocation Guidelines For Values In the Internet Protocol and Related Headers", BCP 37, RFC 2780, DOI 10.17487/RFC2780, March 2000, <https://www.rfc-editor.org/info/rfc2780>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <https://www.rfc-editor.org/info/rfc3032>.

[RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <https://www.rfc-editor.org/info/rfc4385>.

[RFC4928] Swallow, G., Bryant, S., and L. Andersson, "Avoiding Equal Cost Multipath Treatment in MPLS Networks", BCP 128, RFC 4928, DOI 10.17487/RFC4928, June 2007, <https://www.rfc-editor.org/info/rfc4928>.

[RFC6391] Bryant, S., Ed., Filsfils, C., Drafz, U., Kompella, V., Regan, J., and S. Amante, "Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network", RFC 6391, DOI 10.17487/RFC6391, November 2011, <https://www.rfc-editor.org/info/rfc6391>.

[RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <https://www.rfc-editor.org/info/rfc6790>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

[RFC8469] Bryant, S., Malis, A., and I. Bagdonas, "Recommendation to Use the Ethernet Control Word", RFC 8469, DOI 10.17487/RFC8469, November 2018, <https://www.rfc-editor.org/info/rfc8469>.

[RFC8964] Varga, B., Ed., Farkas, J., Berger, L., Malis, A., Bryant, S., and J. Korhonen, "Deterministic Networking (DetNet) Data Plane: MPLS", RFC 8964, DOI 10.17487/RFC8964, January 2021, <https://www.rfc-editor.org/info/rfc8964>.

6.2. Informative References

[I-D.ietf-mpls-mna-fwk] Andersson, L., Bryant, S., Bocci, M., and T. Li, "MPLS Network Actions (MNA) Framework", Work in Progress, Internet-Draft, draft-ietf-mpls-mna-fwk-12, 3 November 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-mpls-mna-fwk-12>.

[RFC4446] Martini, L., "IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)", BCP 116, RFC 4446, DOI 10.17487/RFC4446, April 2006, <https://www.rfc-editor.org/info/rfc4446>.

[RFC4761] Kompella, K., Ed. and Y. Rekhter, Ed., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, DOI 10.17487/RFC4761, January 2007, <https://www.rfc-editor.org/info/rfc4761>.

[RFC4762] Lasserre, M., Ed. and V. Kompella, Ed., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, DOI 10.17487/RFC4762, January 2007, <https://www.rfc-editor.org/info/rfc4762>.

[RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009, <https://www.rfc-editor.org/info/rfc5586>.

[RFC7274] Kompella, K., Andersson, L., and A. Farrel, "Allocating and Retiring Special-Purpose MPLS Labels", RFC 7274, DOI 10.17487/RFC7274, June 2014, <https://www.rfc-editor.org/info/rfc7274>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

[RFC9546] Mirsky, G., Chen, M., and B. Varga, "Operations, Administration, and Maintenance (OAM) for Deterministic Networking (DetNet) with the MPLS Data Plane", RFC 9546, DOI 10.17487/RFC9546, February 2024, <https://www.rfc-editor.org/info/rfc9546>.

Authors' Addresses

Kireeti Kompella
Juniper Networks
1133 Innovation Way
Sunnyvale, 94089
United States of America
Email: kireeti.ietf@gmail.com

Stewart Bryant
University of Surrey 5GIC

Email: sb@stewartbryant.com

Matthew Bocci
Nokia
Email: matthew.bocci@nokia.com

Greg Mirsky (editor)
Ericsson
Email: gregimirsky@gmail.com

Loa Andersson
Huawei Technologies
Email: loa@pi.nu

Jie Dong
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing, 100095
China
Email: jie.dong@huawei.com

MPLS Working Group
Internet-Draft
Intended status: Informational
Expires: 22 May 2025

L. Andersson
Huawei Technologies
S. Bryant
University of Surrey 5GIC
M. Bocci
Nokia
T. Li
Juniper Networks

18 November 2024

MPLS Network Actions (MNA) Framework
draft-ietf-mpls-mna-fwk-13

Abstract

This document specifies an architectural framework for the MPLS Network Actions (MNA) technologies. MNA technologies are used to indicate actions that impact the forwarding or other processing (such as monitoring) of the packet along the Label Switched Path (LSP) of the packet and to transfer any additional data needed for these actions.

The document provides the foundation for the development of a common set of network actions and information elements supporting additional operational models and capabilities of MPLS networks.

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 22 May 2025.

Copyright Notice

Table of Contents

1. Introduction

This document specifies an architectural framework for the MPLS Network Actions (MNA) technologies. MNA technologies are used to indicate actions for Label Switched Paths (LSPs) and/or MPLS packets and to transfer data needed for these actions.

The document provides the foundation for the development of a common set of network actions and information elements supporting additional operational models and capabilities of MPLS networks. MNA solutions derived from this framework are intended to address the requirements found in [RFC9613]. In addition, MNA may support actions that overlap existing MPLS functionality. This may be beneficial for numerous reasons, such as making it more efficient to combine existing functionality and new functions in the same MPLS packet.

MPLS forwarding actions are instructions to MPLS routers to apply additional actions when forwarding a packet. These might include load-balancing a packet given its entropy, whether or not to perform fast-reroute on a failure, and whether or not a packet has metadata relevant to the forwarding actions along the path.

This document generalizes the concept of MPLS "forwarding actions" into "network actions" to include any action that an MPLS router is requested to take on the packet. That includes any MPLS forwarding action, but may include other operations (such as security functions, OAM procedures, etc.) that are not directly related to forwarding of the packet. MPLS network actions are always triggered by an MNA packet but may have implications for subsequent traffic, including non-MNA packets, as discussed in Section 2.4.

MNA technologies may redefine the semantics of the Label, Traffic Class (TC), and Time to Live (TTL) fields in an MPLS Label Stack Entry (LSE) within a Network Action Sub-Stack (NAS).

1.1. Requirement Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

Although this is an Informational document, these conventions are applied to achieve clarity in the requirements that are presented.

1.2. Terminology

1.2.1. Normative Definitions

This document adopts the definitions of the following terms and abbreviations from [RFC9613] as normative: "Network Action", "Network Action Indication (NAI)", "Ancillary Data (AD)", and "Scope".

In addition, this document also defines the following terms:

* Network Action Sub-Stack (NAS): A set of related, contiguous LSEs in the MPLS label stack for carrying information related to network actions. The Label, TC, and TTL values in the LSEs in the NAS may be redefined, but the meaning of the S bit is unchanged.

* Network Action Sub-Stack Indicator (NSI): The first LSE in the NAS contains a special label that indicates the start of the NAS.

1.2.2. Abbreviations

Abbreviation	Meaning	Reference
AD	Ancillary Data	[RFC9613]
BIER	Bit Index Explicit Replication	[RFC8279]
BoS	Bottom of Stack	[RFC6790]
bSPL	Base Special Purpose Label	[RFC9017]
ECMP	Equal Cost Multipath	[RFC9522]
EL	Entropy Label	[RFC6790]
ERLD	Entropy Readable Label Depth	[RFC8662]
eSPL	Extended Special Purpose Label	[RFC9017]
HBH	Hop by hop	In the MNA context,

		this document.
I2E	Ingress to Egress	In the MNA context, this document.
IGP	Interior Gateway Protocol	
ISD	In-stack data	[RFC9613]
LSE	Label Stack Entry	[RFC3032]
MNA	MPLS Network Actions	[RFC9613]
MSD	Maximum SID Depth	[RFC8491]
NAI	Network Action Indicator	[RFC9613]
NAS	Network Action Sub-Stack	This document
NSI	Network Action Sub-Stack Indicator	This document
PSD	Post-stack data	[RFC9613] and Section 3.6
RLD	Readable Label Depth	This document
SID	Segment Identifier	[RFC8402]
SPL	Special Purpose Label	[RFC9017]

Table 1: Abbreviations

2. Structure

An MNA solution specifies one or more network actions to apply to an MPLS packet. These network actions and their ancillary data may be carried in sub-stacks within the MPLS label stack and/or post-stack data. A solution must specify where in the label stack the network actions sub-stacks occur, if and how frequently they should be replicated within the label stack, and how the network action sub-

stack and post-stack data are encoded.

It seems highly likely that some ancillary data will be needed at many points along an LSP. Replication of ancillary data throughout the label stack would be highly inefficient, as would a full rewrite of the label stack at each hop, so MNA allows encoding of network actions and ancillary data deeper in the label stack, requiring implementations to look past the first LSE. Processing of the label stack past the top of stack LSE was first introduced with the Entropy Label [RFC6790].

A network action sub-stack contains:

* Network Action Sub-Stack Indicator (NSI): The first LSE in the NAS contains a special purpose label, called the MNA label, which is used to indicate the start of a network action sub-stack.

* Network Action Indicators (NAI): Optionally, a set of indicators that describes the set of network actions. If the set of indicators is not in the sub-stack, a solution could encode them in post-stack data. A network action is said to be present if there is an indicator in the packet that invokes the action.

* In-Stack Data (ISD): A set of zero or more LSEs that carry ancillary data for the network actions that are present. Network action indicators are not considered ancillary data.

Each network action present in the network action sub-stack may have zero or more LSEs of in-stack data. The ordering of the in-stack data LSEs corresponds to the ordering of the network action indicators. The encoding of the in-stack data, if any, for a network action must be specified in the document that defines the network action. In-stack data may be referenced by multiple network actions.

As an example, in-stack data might look like the following label stack with an embedded NAS:

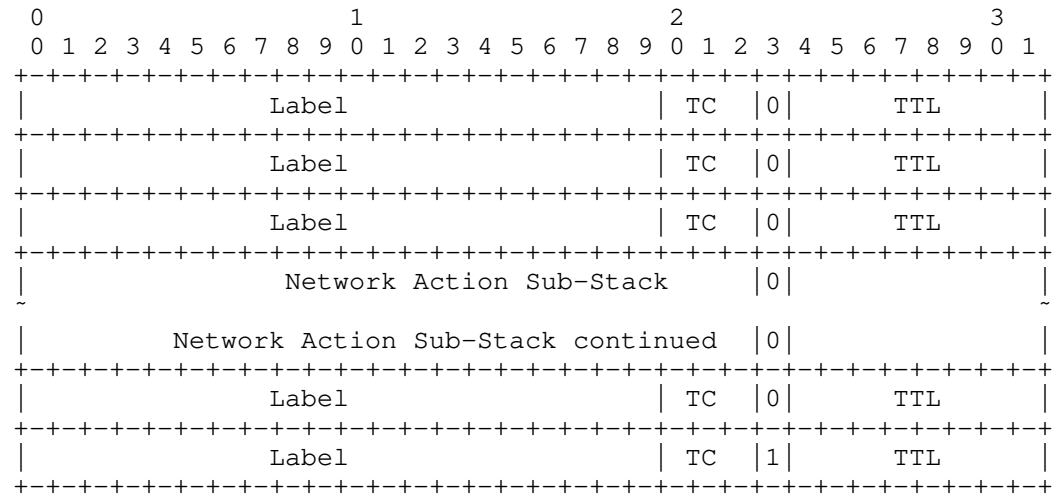


Figure 1: A label stack with an embedded Network Action Sub-Stack

Certain network actions may also specify that data is carried after the label stack. This is called post-stack data. The encoding of the post-stack data, if any, for a network action must be specified in the document that defines the network action. If multiple network actions are present and have post-stack data, the ordering of their post-stack data corresponds to the ordering of the network action indicators.

As an example, post-stack data might appear as a label stack followed by post-stack data, followed by the payload:

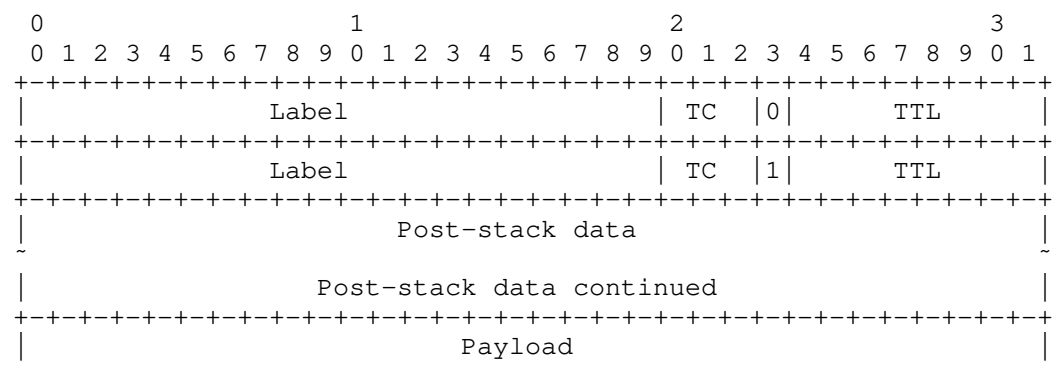


Figure 2: A label stack followed by post-stack data

A solution must specify the order for network actions to be applied to the packet for the actions to have consistent semantics. Since there are many possible orderings, especially with bit catalogs (Section 3.5.1), the solution must provide an unambiguous specification. The precise semantics of an action are dependent on the contents of the packet, including any ancillary data, and the state of the router.

This document assumes that the MPLS WG will select not more than one solution for the encoding of ISD and not more than one solution for the encoding of PSD.

2.1. Scopes

A network action may need to be processed by every node along the path, or some subset of the nodes along its path. Some of the scopes that an action may have are:

* Hop-by-hop (HBH): Every node along the path will perform the action.

* Ingress-to-Egress (I2E): Only the last node on the path will perform the action.

* Select: Only specific nodes along the path will perform the action.

If a solution supports the select scope, it must describe how it specifies the set of nodes to perform the actions.

This framework does not place any constraints on the scope of, or the ancillary data for, a network action. Any network action may appear in any scope or combination of scopes, may have no ancillary data, and may require in-stack data, and/or post-stack data. Some combinations may be sub-optimal, but this framework does not restrict the combinations in an MNA solution. A specific MNA solution may define such constraints.

2.2. Partial Processing

As described in [RFC3031], legacy devices that do not recognize the MNA label will discard the packet if the top label is the MNA label.

Devices that do recognize the MNA label might not implement all of the network actions that are present. A solution must specify how unrecognized network actions that are present should be handled.

One alternative is that an implementation should stop processing network actions when it encounters an unrecognized network action. Subsequent present network actions would not be applied. The result is dependent on the solution's order of operations.

Another alternative is that an implementation should drop any packet that contains any unrecognized present network actions.

A third alternative is that an implementation should perform all recognized present network actions, but ignore all unrecognized present network actions.

Other alternatives may also be possible. The solution should specify the alternative adopted.

In some solutions, an indication may be provided in the packet or in the action as to how the forwarder should proceed if it does not recognize the action. Where an action needs to be processed at every hop, it is recommended that care be taken not to construct an LSP that traverses nodes that do not support that action. It is recognised that in some circumstances it may not be possible to construct an LSP that avoids such nodes, such as when a network is re-converging following a failure or when IPFRR [RFC5714] is taking place.

2.3. Signaling

A node that wishes to make use of MNA and apply network actions to a packet must understand the nodes that the packet will transit, whether or not the nodes support MNA, and the network actions that are to be invoked. These capabilities are presumed to be signaled by protocols that are out-of-scope for this document and are presumed to have per-network action granularity. If a solution requires alternate signaling, it must specify that explicitly.

If a node does not support MNA, then the node will simply ignore any network actions in the packet.

2.3.1. Readable Label Depth

Readable Label Depth (RLD) is defined as the number of LSEs, starting from the top of the stack, that a router can read in an incoming MPLS packet with no performance impact. [RFC8662] introduced Entropy Readable Label Depth (ERLD). Readable Label Depth is the same concept, but generalized and not specifically associated with the Entropy Label (EL) or MNA.

ERLD is not redundant with RLD because ERLD specifically specifies a value of zero if a system does not support the Entropy Label. Since a system could reasonably support MNA or other MPLS functions and needs to advertise an RLD value but not support the Entropy Label, another advertised value is required.

A node that pushes an NAS onto the label stack is responsible for ensuring that all nodes that are expected to process the NAS will have the entire NAS within their RLD. A node SHOULD use signaling (e.g., [RFC9088], [RFC9089]) to determine this.

Per [RFC8662], a node that does not support EL will advertise a value of zero for its ERLD, so advertising ERLD alone does not suffice in all cases. A node MAY advertise both ERLD and RLD and SHOULD do so if its ERLD and RLD values are different. If a node's ERLD and RLD values are the same, it MAY only advertise ERLD for efficiency reasons. If a node supports MNA but does not support EL, then it SHOULD advertise RLD.

RLD is advertised by an IGP MSD-Type value of (TBA) and MAY be advertised as a Node Maximum Segment Identifier (SID) Depth (MSD), Link MSD, or both.

An MNA node MUST use the RLD determined by selecting the first advertised non-zero value from:

* The RLD advertised for the link.

* The RLD advertised for the node.

* The non-zero ERLD for the node.

A node's RLD is a function of its hardware capabilities and is not expected to depend on the specifics of the MNA solution.

2.4. State

A network action can affect the state stored in the network. This implies that a packet may affect how subsequent packets are handled. In particular, one packet may affect subsequent packets in the same LSP.

3. Encoding

Several possible ways to encode NAIs have been proposed. In this section, we summarize the proposals and some considerations for the various alternatives.

When network actions are carried in the MPLS label stack, then regardless of their type, they are represented by a set of LSEs termed a network action sub-stack (NAS). An NAS consists of a special label, optionally followed by LSEs that specify which network actions are to be performed on the packet and the in-stack ancillary data for each indicated network action. Different network actions may be placed together in one NAS or may be carried in different sub-stacks.

[RFC9613] requires that a solution not add unnecessary LSEs to the sub-stack (Section 3.1, requirement 9). Accordingly, solutions should also make efficient use of the bits within the sub-stack (except the S-bit), as inefficient use of the bits could result in the addition of unnecessary LSEs.

3.1. The MNA Label

The first LSE in a network action sub-stack contains a special label that indicates a network action sub-stack. A solution has several choices for this special label.

3.1.1. Existing Base SPL

A solution may reuse an existing Base SPL (bSPL). If it elects to do so, it must explain how the usage is backward compatible, including in the case where there is ISD.

If an existing inactive bSPL is selected that will not be backward compatible, then it must first be retired per [RFC7274] and then reallocated.

3.1.2. New Base SPL

A solution may select a new bSPL.

3.1.3. New Extended SPL

A solution may select a new Extended SPL (eSPL). If it elects to do so, it must address the requirement for the minimal number of LSEs.

3.1.4. User-Defined Label

A solution may allow the network operator to define the label that indicates the network action sub-stack. This creates management overhead for the network operator to coordinate the use of this label across all nodes on the path using management or signaling protocols. The user-defined label could be network-wide or LSP-specific. If a solution elects to use a user-defined label, the solution should

justify this overhead.

3.2. TC and TTL

In the first LSE of the network action sub-stack, only the 20 bits of Label Value and the Bottom of Stack bit are used by NSI; the TC field (3 bits) and the TTL (8 bits) are not used. This could leave 11 bits that could be used for MNA purposes.

3.2.1. TC and TTL retained

If the solution elects to retain the TC and TTL fields, then the first LSE of the network action sub-stack would appear as described in [RFC3032]:

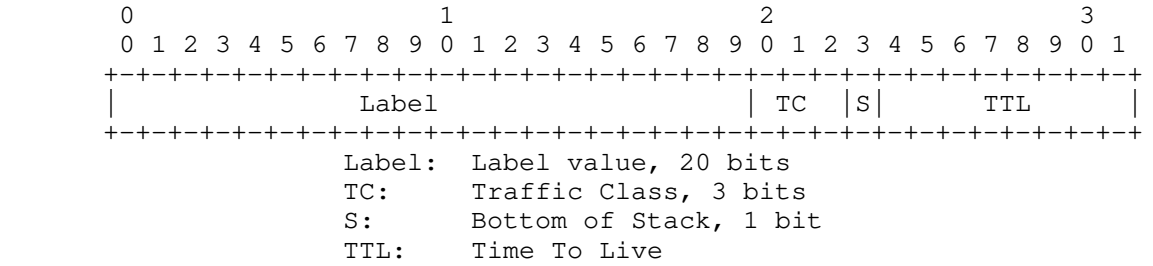
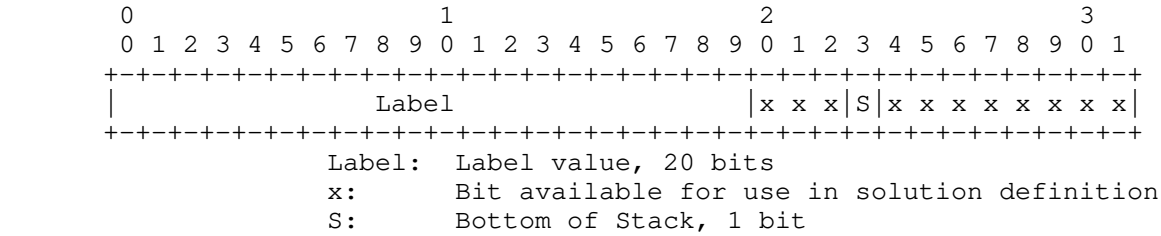


Figure 3: A Label Stack Entry

Further LSEs would be needed to encode NAIs. If a solution elects to retain these fields, it must address the requirement for the minimal number of LSEs.

3.2.2. TC and TTL Repurposed

If the solution elects to reuse the TC and TTL fields, then the first LSE of the network action sub-stack would appear as:



The solution may use more LSEs to contain NAIs. If a solution elects to use more LSEs it must address the requirement for the minimal number of LSEs.

3.3. Length of the NAS

A solution must have a mechanism (such as an indication of the length of the NAS) to enable an implementation to find the end of the NAS. This must be easily processed even by implementations that do not understand the full contents of the NAS. Two options are described below, other solutions may be possible.

3.3.1. Last/Continuation Bits

A solution may use a bit per LSE to indicate whether the NAS continues into the next LSE or not. The bit may indicate continuation by being set or by being clear. The overhead of this approach is one bit per LSE and has the advantage that it can effectively encode an arbitrarily sized NAS. This approach is efficient if the NAS is small.

3.3.2. Length Field

A solution may opt to have a fixed size length field at a fixed location within the NAS. The fixed size of the length field may not be large enough to support all possible NAS contents. This approach may be more efficient if the NAS is longer but not longer than can be described by the length field.

Advice from one hardware designer recommends a length field as this minimizes branching in the logic.

3.4. Encoding of Scopes

A solution may choose to explicitly encode the scope of each action contained in a network action sub-stack. For example, a NAS might contain Action A (HBH), Action B (HBH), and Action C (HBH). A solution may alternately choose to have the scope encoded implicitly, based on the actions present in the network action sub-stack. For example, a NAS might contain HBH scope actions: A, B, C. This choice may have performance implications as an implementation might have to parse the network actions that are present in a network action sub-stack only to discover that there are no actions for it to perform.

For example, suppose that an NAS is embedded in a label stack at a depth of 6 LSEs and that the NAS contains 3 actions, each with Select scope. These actions are not applicable at the current node and should be ignored. If the scope is encoded explicitly with each action, then an implementation must parse each action. However, if the scope is encoded as part of the NAS, then an implementation need only parse the start of the NAS and need not parse individual actions.

Solutions need to consider the order of scoped NAIs and their associated AD within individual sub-stacks and the order of per-scope sub-stacks so that network actions and the AD can be most readily found and need not be processed by nodes that are not required to handle those actions.

3.5. Encoding a Network Action

Two options for encoding NAIs are described below, other solutions may be possible. Any solution should allow the encoding of an arbitrary number of NAIs.

3.5.1. Bit Catalogs

A solution may opt to encode the set of network actions as a list of bits, sometimes known as a catalog. The solution must provide a mechanism to determine how many LSEs are devoted to the catalog when the NAIs are carried in-stack. A set bit in the catalog would indicate that the corresponding network action is present.

Catalogs are efficient if the number of present network actions is relatively high and if the size of the necessary catalog is small. For example, if the first 16 actions are all present, a catalog can encode this in 16 bits. However, if the number of possible actions is large, then a catalog can become inefficient. Selecting only one action that is the 256th action would require a catalog of 256 bits, which would require more than one LSE when the NAIs are carried in-stack.

A solution may include a bit remapping mechanism so that a given domain may optimize for its commonly used actions.

3.5.2. Operation Codes

A solution may opt to encode the set of present network actions as a list of operation codes (opcodes). Each opcode is a fixed number of bits. The size of the opcode bounds the number of network actions that the solution can support.

Opcodes are efficient if there are only one or two active network actions. For example, if an opcode is 8 bits, then two active network actions could be encoded in 16 bits. However, if 16 actions are required, then opcodes would consume 128 bits. Opcodes are efficient at encoding a large number of possible actions. If only the 256th action is to be selected, that still requires 8 bits.

3.6. Encoding of Post-Stack Data

A solution may carry some NAI and AD as PSD. For ease of parsing, all AD should be co-located with its NAI.

If there are multiple instances of post-stack data, they should occur in the same order as their relevant network action sub-stacks and then in the same order as their relevant network actions occur within the network action sub-stacks.

3.6.1. First Nibble Considerations

The first nibble after the label stack has been used to convey information in certain cases [RFC4385]. A consolidated view of first nibble uses is provided in [I-D.ietf-mpls-1stnibble].

For example, in [RFC4928] this nibble is investigated to find out if it has the value "4" or "6". If it is not, it is assumed that the packet payload is not IPv4 or IPv6, and Equal Cost Multipath (ECMP) is not performed.

It should be noted that this is an inexact method. For example, an Ethernet Pseudowire without a control word might have "4" or "6" in the first nibble and thus will be ECMP'ed.

Nevertheless, the method is implemented and deployed, it is used today and will be for the foreseeable future.

The use of the first nibble for Bit Index Explicit Replication (BIER) is specified in [RFC8296]. BIER sets the first nibble to 5. The same is true for a BIER payload as for any use of the first nibble: it is not possible to conclude that the payload is BIER even if the first nibble is set to 5 because an Ethernet pseudowire without a control word might begin with a 5. However, the BIER approach meets the design goal of [RFC8296] to determine that the payload is IPv4, IPv6 or with the header of a pseudowire packet with a control word, rather than being a payload belonging to a BIER or some other type of packet.

[RFC4385] allocates 0b0000 for the pseudowire control word and 0b0001 as the control word for the pseudowire Associated Channel Header (ACH).

A PSD solution should specify the contents of the first nibble, the actions to be taken for the value, and the interaction with post-stack data used concurrently by other MPLS applications.

4. Semantics

For MNA to be consistent across implementations and predictable in operational environments, its semantics need to be entirely predictable. An MNA solution MUST specify a deterministic order for processing each of the Network Actions in a packet. Each network action must specify how it interacts with all other previously defined network actions. Private network actions are network actions that are not publicly documented. Private network actions MUST be included in the ordering of network actions, but the interactions of private actions with other actions are outside of the scope of this document.

5. Definition of a Network Action

Network actions should be defined in a document that must contain:

* Name: The name of the network action.

* Network Action Indicator: The bit position or opcode that indicates that the network action is active.

* Scope: The document should specify which nodes should perform the network action as described in Section 2.1.

* State: The document should specify if the network action can modify state in the network, and if so, the state that may be modified and its side effects.

* Required/Optional: The document should specify whether a node is required to perform the network action.

* In-Stack Data: The number of LSEs of in-stack data, if any, and its encoding. If this is of a variable length, then the solution must specify how an implementation can determine this length without implementing the network action.

* Post-Stack Data: The encoding of post-stack data, if any. If this is of a variable length, then the solution must specify how an implementation can determine this length without implementing the network action.

A solution should create an IANA registry for network actions.

6. Management Considerations

Network operators will need to be cognizant of which network actions are supported by which nodes and will need to ensure that this is signaled. Some solutions may require network-wide configuration to synchronize the use of the labels that indicate the start of an NAS. Solution documents must make clear what management considerations apply to the solutions they are describing. Solutions documents must describe mechanisms for performing network diagnostics in the presence of MNAs.

7. Security Considerations

An analysis of the security of MPLS systems is provided in [RFC5920], which also notes that the MPLS forwarding plane has no built-in security mechanisms.

Central to the security of MPLS networks is operational security of the network; something that operators of MPLS networks are well versed in. The deployment of link-level security (e.g., [MACsec]) prevents link traffic observation covertly acquiring the label stack for an attack. This is particularly important in the case of a network deploying MNA, because the MNA information may be sensitive. Thus the confidentiality and authentication achieved through the use of link-level security is particularly advantageous.

Some additional proposals to add encryption to the MPLS forwarding plane have been suggested [I-D.ietf-mpls-opportunistic-encrypt], but no mechanisms have been agreed upon at the time of publication of this document. [I-D.ietf-mpls-opportunistic-encrypt] offers hop-by-hop security that encrypts the label stack and is functionally equivalent to that provided by [MACsec]. Alternatively, it also offers end-to-end encryption of the MPLS payload with no cryptographic integrity protection of the MPLS label stack.

Particular care would be needed when introducing any end-to-end security mechanism to allow an in-stack MNA solution that needed to employ on-path modification of the MNA data, or where post-stack MNA data needed to be examined on-path.

A cornerstone of MPLS security is to protect the network from processing MPLS labels originated outside the network.

Operators have considerable experience in excluding MPLS-encoded packets at the network boundaries for example, by excluding all MPLS packets and all packets that are revealed to be carrying an MPLS packet as the payload of IP tunnels. Where such packets are accepted into an MPLS network from an untrusted third party, non-MPLS packets

are immediately encapsulated in an MPLS label stack specified by the MPLS network operator and MPLS packets have additional label stack entries imported as specified by the MPLS network operator. Thus, it is difficult for an attacker to pass an MPLS-encoded packet into a network or to present any instructions to the network forwarding system.

Within a single well-managed domain, an adjacent domain may be considered to be trusted provided that it is sufficiently shielded from third-party traffic ingress and third-party traffic observation. In such a situation, no new security vulnerabilities are introduced by MNA.

In some inter-domain applications (including carrier's carrier) where a first network's MPLS traffic is encapsulated directly over a second MPLS network by simply pushing additional MPLS LSEs, the contents of the first network's payload and label stack may be visible to the forwarders in the second network. Historically this has been benign, and indeed useful for ECMP. However, if the first network's traffic has MNA information this may be exposed to MNA-capable forwarders causing unpredictable behavior or modification of the customer MPLS label stack or MPLS payload. This is an increased vulnerability introduced by MNA that SHOULD be addressed in any MNA solution.

Several mitigations are available to an operator:

a) Reject all incoming packets containing MNA information that do not come from a trusted network. Note that it may be acceptable to accept and process MNA information from a trusted network.

b) Fully encapsulate the inbound packet in a new additional MPLS label stack such that the forwarder finds a Bottom of Stack (BoS) bit imposed by the carrier network and only finds MNA information added by the carrier network.

A mitigation that we reject as unsafe is having the ingress LSR push sufficient additional labels such that any MNA information received in packets entering the network from a third-party network is made inaccessible due to it being below the RLD. This is unsafe in the presence of an overly conservative RLD value which can result in the third-party MNA information becoming visible to and acted on by an MNA forwarder in the carrier network.

8. IANA Considerations

This document requests that IANA allocate a code point from the "IGP MSD-Types" registry in the "Interior Gateway Protocol (IGP) Parameters" namespace for "Readable Label Depth", referencing this document.

9. Acknowledgements

This document is the result of work started in MPLS Open Design Team, with participation by the MPLS, PALS, and DETNET working groups.

The authors would like to thank Adrian Farrel for his contributions and to John Drake, Toerless Eckert, and Jie Dong for their comments.

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>.

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <https://www.rfc-editor.org/info/rfc3031>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <https://www.rfc-editor.org/info/rfc3032>.

[RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <https://www.rfc-editor.org/info/rfc4385>.

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <https://www.rfc-editor.org/info/rfc5920>.

[RFC7274] Kompella, K., Andersson, L., and A. Farrel, "Allocating and Retiring Special-Purpose MPLS Labels", RFC 7274, DOI 10.17487/RFC7274, June 2014, <https://www.rfc-editor.org/info/rfc7274>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9017] Andersson, L., Kompella, K., and A. Farrel, "Special-Purpose Label Terminology", RFC 9017, DOI 10.17487/RFC9017, April 2021, <https://www.rfc-editor.org/info/rfc9017>.

[RFC9613] Bocci, M., Ed., Bryant, S., and J. Drake, "Requirements for Solutions that Support MPLS Network Actions (MNAs)", RFC 9613, DOI 10.17487/RFC9613, August 2024, <https://www.rfc-editor.org/info/rfc9613>.

10.2. Informative References

[I-D.ietf-mpls-opportunistic-encrypt]
Farrel, A. and S. Farrell, "Opportunistic Security in MPLS Networks", Work in Progress, Internet-Draft, draft-ietf-mpls-opportunistic-encrypt-03, 28 March 2017, <https://datatracker.ietf.org/doc/html/draft-ietf-mpls-opportunistic-encrypt-03>.

[I-D.ietf-mpls-1stnibble]
Kompella, K., Bryant, S., Bocci, M., Mirsky, G., Andersson, L., and J. Dong, "IANA Registry for the First Nibble Following a Label Stack", Work in Progress, Internet-Draft, draft-ietf-mpls-1stnibble-11, 12 November 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-mpls-1stnibble-11>.

[RFC4928] Swallow, G., Bryant, S., and L. Andersson, "Avoiding Equal Cost Multipath Treatment in MPLS Networks", BCP 128, RFC 4928, DOI 10.17487/RFC4928, June 2007, <https://www.rfc-editor.org/info/rfc4928>.

[RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, DOI 10.17487/RFC5714, January 2010, <https://www.rfc-editor.org/info/rfc5714>.

[RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <https://www.rfc-editor.org/info/rfc6790>.

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491, DOI 10.17487/RFC8491, November 2018, <https://www.rfc-editor.org/info/rfc8491>.

[RFC8662] Kini, S., Kompella, K., Sivabalan, S., Litkowski, S., Shakir, R., and J. Tantsura, "Entropy Label for Source Packet Routing in Networking (SPRING) Tunnels", RFC 8662, DOI 10.17487/RFC8662, December 2019, <https://www.rfc-editor.org/info/rfc8662>.

[RFC9088] Xu, X., Kini, S., Psenak, P., Filsfils, C., Litkowski, S., and M. Bocci, "Signaling Entropy Label Capability and Entropy Readable Label Depth Using IS-IS", RFC 9088, DOI 10.17487/RFC9088, August 2021, <https://www.rfc-editor.org/info/rfc9088>.

[RFC9089] Xu, X., Kini, S., Psenak, P., Filsfils, C., Litkowski, S., and M. Bocci, "Signaling Entropy Label Capability and Entropy Readable Label Depth Using OSPF", RFC 9089, DOI 10.17487/RFC9089, August 2021, <https://www.rfc-editor.org/info/rfc9089>.

[RFC9522] Farrel, A., Ed., "Overview and Principles of Internet Traffic Engineering", RFC 9522, DOI 10.17487/RFC9522, January 2024, <https://www.rfc-editor.org/info/rfc9522>.

[MACsec] IEEE Computer Society, "IEEE 802.1AE Media Access Control (MAC) Security", August 2006.

Authors' Addresses

Loa Andersson
Huawei Technologies
Email: loa@pi.nu

Stewart Bryant
University of Surrey 5GIC
Email: sb@stewartbryant.com

Matthew Bocci
Nokia
Email: matthew.bocci@nokia.com

Tony Li
Juniper Networks
Email: tony.li@tony.li

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 30 May 2025

S. Gringeri
J. Whittaker
Verizon
N. Leymann
Deutsche Telekom
C. Schmutzer, Ed.
Cisco Systems, Inc.
C. Brown
Ciena Corporation
26 November 2024

Private Line Emulation over Packet Switched Networks
draft-ietf-pals-ple-12

Abstract

This document describes methods and requirements for implementing the encapsulation of high-speed bit-streams into virtual private wire services (VPWS) over packet switched networks (PSN) providing complete signal transport transparency.

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 30 May 2025.

Table of Contents

1. Introduction and Motivation

This document describes a method called Private Line Emulation (PLE) for encapsulating high-speed bit-streams as Virtual Private Wire Service (VPWS) over Packet Switched Networks (PSN).

This emulation suits applications, where carrying Protocol Data Units (PDUs) as defined in [RFC4906] or [RFC4448] is not enough, physical layer signal transparency is required and data or framing structure interpretation of the PE would be counterproductive.

One example of such case is two Ethernet connected Customer Edge (CE) devices and the need for Synchronous Ethernet operation between them without the intermediate Provider Edge (PE) devices interfering or addressing concerns about Ethernet control protocol transparency for PDU based carrier Ethernet services, beyond the behavior definitions of Metro Ethernet Forum (MEF) specifications.

Another example would be a Storage Area Networking (SAN) extension between two data centers. Operating at a bit-stream level allows for a connection between Fibre Channel switches without interfering with any of the Fibre Channel protocol mechanisms.

Also, SONET/SDH add/drop multiplexers or cross-connects can be interconnected without interfering with the multiplexing structures and networks mechanisms. This is a key distinction to Circuit Emulation over Packet (CEP) defined in [RFC4842] where demultiplexing and multiplexing is desired in order to operate per SONET Synchronous Payload Envelope (SPE) and Virtual Tributary (VT) or SDH Virtual Container (VC). Said in another way, PLE does provide an independent layer network underneath the SONET/SDH layer network, whereas CEP does operate at the same level and peer with the SONET/SDH layer network.

The mechanisms described in this document follow principles similar to Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP) defined in [RFC4553]. The applicability is expanded beyond the narrow set of PDH interfaces (T1, E1, T3 and E3) to allow the transport of signals from many different technologies such as Ethernet, Fibre Channel, SONET/SDH [GR253]/[G.707] and OTN [G.709] at gigabit speeds. The signals are treated as bit-stream payload which was defined in the Pseudo Wire Emulation Edge-to-Edge (PWE3) architecture in [RFC3985] sections 3.3.3 and 3.3.4.

2. Requirements Notation

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 RFC 2119.

3. Terminology and Reference Model

3.1. Terminology

* ACH – Associated Channel Header [RFC7212]
* AIS – Alarm Indication Signal
* AIS-L – Line AIS
* AS – Autonomous System
* ASBR – Autonomous System Border Router
* MS-AIS – Multiplex Section AIS
* BITS – Building Integrated Timing Supply
* CBR – Constant Bit Rate
* CE – Customer Edge
* CEP – Circuit Emulation over Packet [RFC4842]
* CSRC – Contributing SouRCe [RFC3550]
* DEG – Degradation
* ES – Errored Second
* FEC – Forward Error Correction
* ICMP – Internet Control Message Protocol [RFC4443]
* IEEE – Institute of Electrical and Electronics Engineers
* INCITS – InterNational Committee for Information Technology Standards
* IWF – InterWorking Function
* LDP – Label Distribution Protocol [RFC5036], [RFC8077]

* LF – Local Fault
* LOF – Loss Of Frame
* LOM – Loss Of Multiframe
* LOS – Loss Of Signal
* LPI – Low Power Idle
* LSP – Label Switched Path
* MEF – Metro Ethernet Forum
* MPLS – Multi Protocol Label Switching [RFC3031]
* NOS – Not Operational
* NSP – Native Service Processor [RFC3985]
* ODUk – Optical Data Unit k
* OTN – Optical Transport Network
* OTUk – Optical Transport Unit k
* PCS – Physical Coding Sublayer
* PDH – Plesiochronous Digital Hierarchy
* PDV – Packet Delay Variation
* PE – Provider Edge
* PLE – Private Line Emulation
* PLOS – Packet Loss Of Signal
* PLR – Packet Loss Ratio
* PMA – Physical Medium Attachment
* PMD – Physical Medium Dependent
* PSN – Packet Switched Network
* PTP – Precision Time Protocol

* PW - Pseudowire [RFC3985]
* PWE3 - Pseudo Wire Emulation Edge-to-Edge [RFC3985]
* P2P - Point-to-Point
* QOS - Quality Of Service
* RDI - Remote Defect Indication
* RSVP-TE - Resource Reservation Protocol Traffic Engineering [RFC4875]
* RTCP - RTP Control Protocol [RFC3550]
* RTP - Realtime Transport Protocol [RFC3550]
* SAN - Storage Area Network
* SAToP - Structure-Agnostic Time Division Multiplexing (TDM) over Packet [RFC4553]
* SD - Signal Degrade
* SES - Severely Errored Second
* SDH - Synchronous Digital Hierarchy
* SID - Segment Identifier [RFC8402]
* SPE - Synchronous Payload Envelope
* SR - Segment Routing [RFC8402]
* SRH - Segment Routing Header [RFC8402]
* SR-TE - Segment Routing Traffic Engineering [RFC9256]
* SRTP - Secure Realtime Transport Protocol [RFC3711]
* SRv6 - Segment Routing over IPv6 Dataplane [RFC8986]
* SSRC - Synchronization SouRCe [RFC3550]
* SONET - Synchronous Optical Network
* TCP - Transmission Control Protocol [RFC9293]

* TDM - Time Division Multiplexing

* TTS - Transmitter Training Signal

* UAS - Unavailable Second

* VPWS - Virtual Private Wire Service [RFC3985]

* VC - Virtual Circuit

* VT - Virtual Tributary

The term Interworking Function (IWF) is used to describe the functional block that encapsulates bit streams into PLE packets and in the reverse direction decapsulates PLE packets and reconstructs bit streams.

3.2. Reference Models

The reference model for PLE is illustrated in Figure 1 and is inline with the reference model defined in Section 4.1 of [RFC3985]. PLE does rely on PWE3 pre-processing, in particular the concept of a Native Service Processing (NSP) function defined in Section 4.2.2 of [RFC3985].

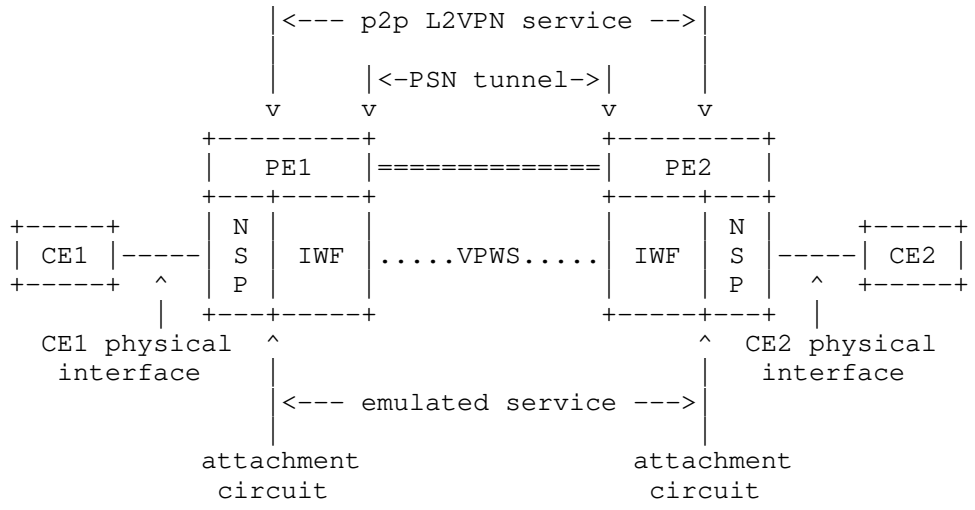


Figure 1: PLE Reference Model

PLE embraces the minimum intervention principle outlined in Section 3.3.5 of [RFC3985] whereas the data is flowing through the PLE encapsulation layer as received without modifications.

For some service types the NSP function is responsible for performing operations on the native data received from the CE. Examples are terminating Forward Error Correction (FEC), terminating the OTUk layer for OTN or dealing with multi-lane processing. After the NSP, the IWF is generating the payload of the VPWS which is carried via a PSN tunnel.

To allow the clock of the transported signal to be carried across the PLE domain in a transparent way the relative network synchronization reference model and deployment scenario outlined in Section 4.3.2 of [RFC4197] are applicable and are shown in Figure 2.

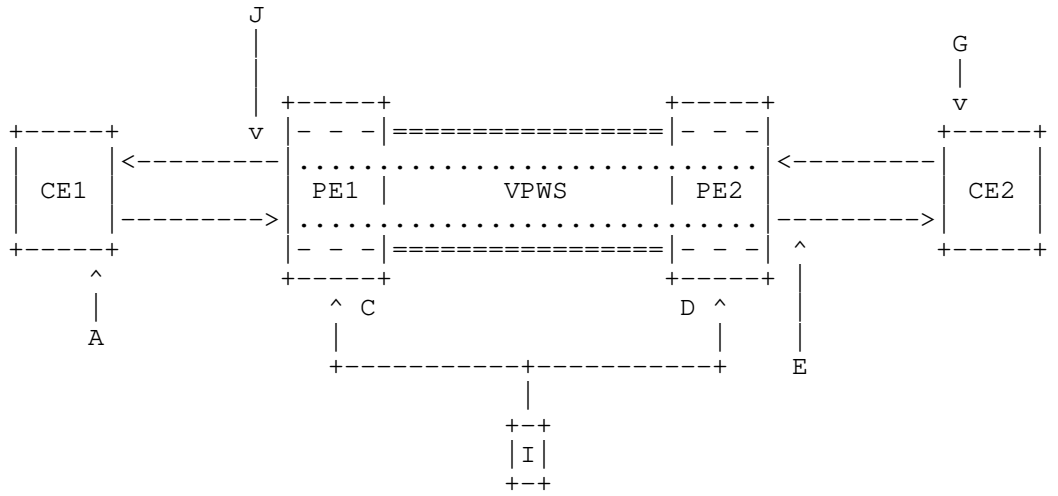


Figure 2: Relative Network Scenario Timing

The local oscillators C of PE1 and D of PE2 are locked to a common clock I.

The attachment circuit clock E is generated by PE2 via a differential clock recovery method in reference to the common clock I. For this to work the difference between clock A and clock C (locked to I) MUST be explicitly transferred from PE1 to PE2 using the timestamp inside the RTP header.

For the reverse direction PE1 does generate the attachment circuit clock J and the clock difference between G and D (locked to I) transferred from PE2 to PE1.

The method used to lock clocks C and D to the common clock I is out of scope of this document, but there are already several well established concepts for achieving frequency synchronization available.

While using external timing inputs (aka BITS) or synchronous Ethernet as defined in [G.8261] the characteristics and limits defined in [G.8262] have to be considered.

While relying on precision time protocol (PTP) as defined in [G.8265.1], the network limits defined in [G.8261.1] have to be considered.

4. Emulated Services

This specification describes the emulation of services from a wide range of technologies, such as TDM, Ethernet, Fibre Channel, or OTN, as bit streams or structured bit streams, as defined in Section 3.3.3 and Section 3.3.4 of [RFC3985].

4.1. Generic PLE Service

The generic PLE service is an example of the bit stream defined in Section 3.3.3 of [RFC3985].

Under the assumption that the CE-bound IWF is not responsible for any service specific operation, a bit stream of any rate can be carried using the generic PLE payload.

There is no NSP function present for this service.

4.2. Ethernet services

Ethernet services are special cases of the structured bit stream defined in Section 3.3.4 of [RFC3985].

IEEE has defined several layers for Ethernet in [IEEE802.3]. Emulation is operating at the physical (PHY) layer, more precisely at the Physical Coding Sublayer (PCS).

Over time many different Ethernet interface types have been specified in [IEEE802.3] with a varying set of characteristics such as optional vs mandatory FEC and single-lane vs multi-lane transmission.

Ethernet interface types with backplane physical media dependent (PMD) variants and ethernet interface types mandating auto-negotiation (except 1000Base-X) are out of scope for this document.

All Ethernet services are leveraging the basic PLE payload and interface specific mechanisms are confined to the respective service specific NSP functions.

4.2.1. 1000BASE-X

The PCS layer of 1000BASE-X defined in clause 36 of [IEEE802.3] is based on 8B/10B code.

The PSN-bound NSP function does not modify the received data and is transparent to auto-negotiation but is responsible to detect 1000BASE-X specific attachment circuit faults such as LOS and sync loss.

When the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set, the CE-bound NSP function MAY disable its transmitter as no appropriate maintenance signal was defined for 1000BASE-X by IEEE.

4.2.2. 10GBASE-R and 25GBASE-R

The PCS layers of 10GBASE-R defined in clause 49 and 25GBASE-R defined in clause 107 of [IEEE802.3] are based on a 64B/66B code.

[IEEE802.3] clauses 74 and 108 do define an optional FEC layer, if present the PSN-bound NSP function MUST terminate the FEC and the CE-bound NSP function MUST generate the FEC.

The PSN-bound NSP function is also responsible to detect 10GBASE-R and 25GBASE-R specific attachment circuit faults such as LOS and sync loss.

The PSN-bound IWF is mapping the scrambled 64B/66B code stream into the basic PLE payload.

The CE-bound NSP function MUST perform

* PCS code sync

* descrambling

in order to properly

* transform invalid 66B code blocks into proper error control characters /E/

* insert Local Fault (LF) ordered sets when the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set

Note: Invalid 66B code blocks typically are a consequence of the CE-bound IWF inserting replacement data in case of lost PLE packets, or if the far-end PSN-bound NSP function did set sync headers to 11 due to uncorrectable FEC errors.

Before sending the bit stream to the CE, the CE-bound NSP function MUST also scramble the 64B/66B code stream.

4.2.3. 40GBASE-R, 50GBASE-R and 100GBASE-R

The PCS layers of 40GBASE-R and 100GBASE-R defined in clause 82 and of 50GBASE-R defined in clause 133 of [IEEE802.3] are based on a 64B/66B code transmitted over multiple lanes.

[IEEE802.3] clauses 74 and 91 do define an optional FEC layer, if present the PSN-bound NSP function MUST terminate the FEC and the CE-bound NSP function MUST generate the FEC.

To gain access to the scrambled 64B/66B code stream the PSN-bound NSP further MUST perform

* block synchronization

* PCS lane de-skew

* PCS lane reordering

The PSN-bound NSP function is also responsible to detect 40GBASE-R, 50GBASE-R and 100GBASE-R specific attachment circuit faults such as LOS and loss of alignment.

The PSN-bound IWF is mapping the serialized, scrambled 64B/66B code stream including the alignment markers into the basic PLE payload.

The CE-bound NSP function MUST perform

* PCS code sync

* alignment marker removal

* descrambling

in order to properly

* transform invalid 66B code blocks into proper error control characters /E/

* insert Local Fault (LF) ordered sets when the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set

Note: Invalid 66B code blocks typically are a consequence of the CE-bound IWF inserting replacement data in case of lost PLE packets, or if the far-end PSN-bound NSP function did set sync headers to 11 due to uncorrectable FEC errors.

When sending the bit stream to the CE, the CE-bound NSP function MUST also perform

* scrambling of the 64B/66B code

* block distribution

* alignment marker insertion

4.2.4. 200GBASE-R and 400GBASE-R

The PCS layers of 200GBASE-R and 400GBASE-R defined in clause 119 of [IEEE802.3] are based on a 64B/66B code transcoded to a 256B/257B code to reduce the overhead and make room for a mandatory FEC.

To gain access to the 64B/66B code stream the PSN-bound NSP further MUST perform

* alignment lock and de-skew

* PCS Lane reordering and de-interleaving

* FEC decoding

* post-FEC interleaving

* alignment marker removal

* descrambling

* reverse transcoding from 256B/257B to 64B/66B

Further the PSN-bound NSP MUST perform rate compensation and scrambling before the PSN-bound IWF is mapping the same into the basic PLE payload.

Rate compensation is applied so that the rate of the 66B encoded bit stream carried by PLE is 528/544 times the nominal bitrate of the 200GBASE-R or 400GBASE-R at the PMA service interface. X number of 66 byte long rate compensation blocks are inserted every X*20479 number of 66B client blocks. For 200GBASE-R the value of X is 16 and for 400GBASE-R the value of X is 32. Rate compensation blocks are special 66B control characters of type 0x00 that can easily be searched for by the CE-bound IWF in order to remove them.

The PSN-bound NSP function is also responsible to detect 200GBASE-R and 400GBASE-R specific attachment circuit faults such as LOS and loss of alignment.

The CE-bound NSP function MUST perform

* PCS code sync

* descrambling

* rate compensation block removal

in order to properly

* transform invalid 66B code blocks into proper error control characters /E/

* insert Local Fault (LF) ordered sets when the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set

Note: Invalid 66B code blocks typically are a consequence of the CE-bound IWF inserting replacement data in case of lost PLE packets, or if the far-end PSN-bound NSP function did set sync headers to 11 due to uncorrectable FEC errors.

When sending the bit stream to the CE, the CE-bound NSP function MUST also perform

* transcoding from 64B/66B to 256B/257B

* scrambling

* alignment marker insertion

* pre-FEC distribution

* FEC encoding

* PCS Lane distribution

4.2.5. Energy Efficient Ethernet (EEE)

Section 78 of [IEEE802.3] does define the optional Low Power Idle (LPI) capability for Ethernet. Two modes are defined

* deep sleep

* fast wake

Deep sleep mode is not compatible with PLE due to the CE ceasing transmission. Hence there is no support for LPI for 10GBASE-R services across PLE.

When in fast wake mode the CE transmits /LI/ control code blocks instead of /I/ control code blocks and therefore PLE is agnostic to it. For 25GBASE-R and higher services across PLE, LPI is supported as only fast wake mode is applicable.

4.3. SONET/SDH Services

SONET/SDH services are special cases of the structured bit stream defined in Section 3.3.4 of [RFC3985].

SDH interfaces are defined in [G.707] and SONET interfaces are defined in [GR253].

The PSN-bound NSP function does not modify the received data but is responsible to detect SONET/SDH interface specific attachment circuit faults such as LOS, LOF and OOF.

Data received by the PSN-bound IWF is mapped into the basic PLE payload without any awareness of SONET/SDH frames.

When the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set, the CE-bound NSP function is responsible for generating the

* MS-AIS maintenance signal defined in clause 6.2.4.1.1 of [G.707] for SDH services

* AIS-L maintenance signal defined in clause 6.2.1.2 of [GR253] for SONET services

at client frame boundaries.

4.4. Fibre Channel Services

Fibre Channel services are special cases of the structured bit stream defined in Section 3.3.4 of [RFC3985].

The T11 technical committee of INCITS has defined several layers for Fibre Channel. Emulation is operating at the FC-1 layer.

Over time many different Fibre Channel interface types have been specified with a varying set of characteristics such as optional vs mandatory FEC and single-lane vs multi-lane transmission.

Speed negotiation is out of scope for this document.

All Fibre Channel services are leveraging the basic PLE payload and interface specific mechanisms are confined to the respective service specific NSP functions.

4.4.1. 1GFC, 2GFC, 4GFC and 8GFC

[FC-PI-2] specifies 1GFC and 2GFC. [FC-PI-5] and [FC-PI-5am1] do define 4GFC and 8GFC.

The PSN-bound NSP function is responsible to detect Fibre Channel specific attachment circuit faults such as LOS and sync loss.

The PSN-bound IWF is mapping the received 8B/10B code stream as is directly into the basic PLE payload.

The CE-bound NSP function MUST perform transmission word sync in order to properly

* replace invalid transmission words with the special character K30.7

* insert Not Operational (NOS) ordered sets when the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set

Note: Invalid transmission words typically are a consequence of the CE-bound IWF inserting replacement data in case of lost PLE packets.

[FC-PI-5am1] does define the use of scrambling for 8GFC, in this case the CE-bound NSP MUST also perform descrambling before replacing invalid transmission words or inserting NOS ordered sets. And before sending the bit stream to the, the CE-bound NSP function MUST scramble the 8B/10B code stream.

4.4.2. 16GFC and 32GFC

[FC-PI-5] and [FC-PI-5am1] specify 16GFC and define a optional FEC layer. [FC-PI-6] specifies 32GFC with the FEC layer and transmitter training signal (TTS) support being mandatory.

If FEC is present it must be indicated via TTS during attachment circuit bring up. Further the PSN-bound NSP function MUST terminate the FEC and the CE-bound NSP function must generate the FEC.

The PSN-bound NSP function is responsible to detect Fibre Channel specific attachment circuit faults such as LOS and sync loss.

The PSN-bound IWF is mapping the received 64B/66B code stream as is into the basic PLE payload.

The CE-bound NSP function MUST perform

* transmission word sync

* descrambling

in order to properly

* replace invalid transmission words with the error transmission word 1Eh

* insert Not Operational (NOS) ordered sets when the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set

Note: Invalid transmission words typically are a consequence of the CE-bound IWF inserting replacement data in case of lost PLE packets, or if the far-end PSN-bound NSP function did set sync headers to 11 due to uncorrectable FEC errors.

Before sending the bit stream to the CE, the CE-bound NSP function MUST also scramble the 64B/66B code stream.

4.4.3. 64GFC and 4-lane 128GFC

[FC-PI-7] specifies 64GFC and [FC-PI-6P] specifies 4-lane 128GFC. Both specify a mandatory FEC layer. The PSN-bound NSP function MUST terminate the FEC and the CE-bound NSP function must generate the FEC.

To gain access to the 64B/66B code stream the PSN-bound NSP further MUST perform

* alignment lock and de-skew
* Lane reordering and de-interleaving
* FEC decoding
* post-FEC interleaving
* alignment marker removal
* descrambling
* reverse transcoding from 256B/257B to 64B/66B

Further the PSN-bound NSP MUST perform scrambling before the PSN-bound IWF is mapping the same into the basic PLE payload.

Note : The use of rate compensation is for further study and out of scope for this document.

The PSN-bound NSP function is also responsible to detect Fibre Channel specific attachment circuit faults such as LOS and sync loss.

The CE-bound NSP function MUST perform

* transmission word sync
* descrambling

in order to properly

* replace invalid transmission words with the error transmission word 1Eh
* insert Not Operational (NOS) ordered sets when the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set

Note: Invalid transmission words typically are a consequence of the CE-bound IWF inserting replacement data in case of lost PLE packets, or if the far-end PSN-bound NSP function did set sync headers to 11 due to uncorrectable FEC errors.

When sending the bit stream to the CE, the CE-bound NSP function MUST also perform

* transcoding from 64B/66B to 256B/257B

* scrambling

* alignment marker insertion

* pre-FEC distribution

* FEC encoding

* Lane distribution

4.5. OTN Services

OTN services are special cases of the structured bit stream defined in Section 3.3.4 of [RFC3985].

OTN interfaces are defined in [G.709].

The PSN-bound NSP function MUST terminate the FEC and replace the OTUk overhead in row 1 columns 8-14 with all-0s fixed stuff which results in a extended ODUk frame as illustrated in Figure 3. The frame alignment overhead (FA OH) in row 1 columns 1-7 is kept as it is.

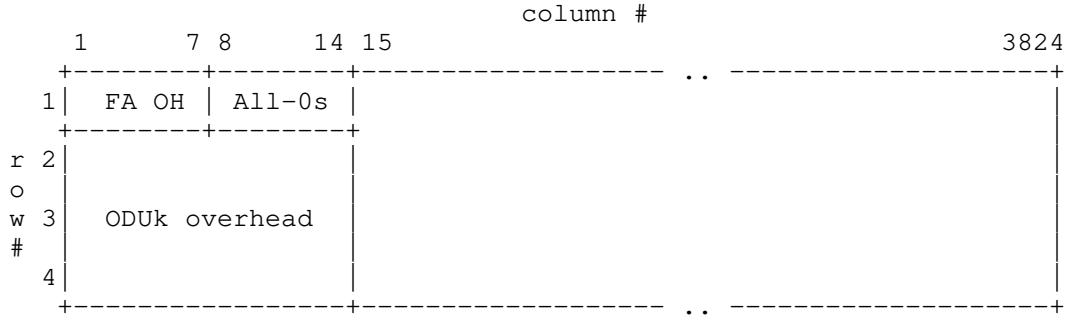


Figure 3: Extended ODUk Frame

The PSN-bound NSP function is also responsible to detect OTUk specific attachment circuit faults such as LOS, LOF, LOM and AIS.

The PSN-bound IWF is mapping the extended ODUk frame into the byte aligned PLE payload.

The CE-bound NSP function will recover the ODUk by searching for the frame alignment overhead in the extended ODUk received from the CE-bound IWF and generates the FEC.

When the CE-bound IWF is in PLOS state or when PLE packets are received with the L-bit being set, the CE-bound NSP function is responsible for generating the ODUk-AIS maintenance signal defined in clause 16.5.1 of [G.709] at client frame boundaries.

5. PLE Encapsulation Layer

The basic packet format used by PLE is shown in the Figure 4.

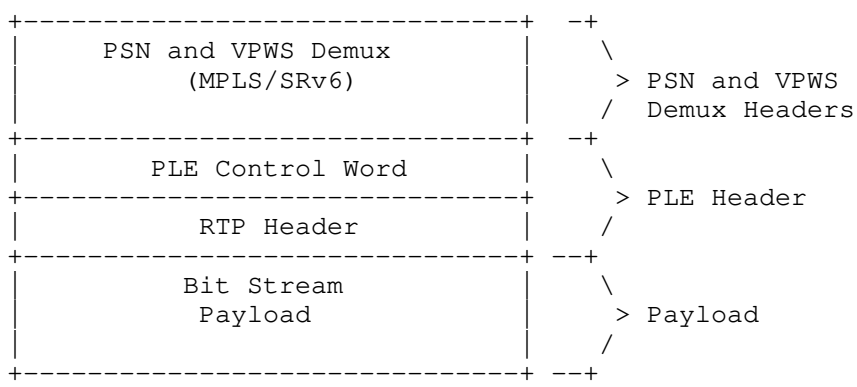


Figure 4: PLE Encapsulation Layer

5.1. PSN and VPWS Demultiplexing Headers

This document does not imply any specific technology to be used for implementing the VPWS demultiplexing and PSN layers.

The total size of a PLE packet for a specific PW MUST NOT exceed the path MTU between the pair of PEs terminating this PW.

When a MPLS PSN layer is used, a VPWS label provides the demultiplexing mechanism as described in Section 5.4.2 of [RFC3985]. The PSN tunnel can be a simple best path Label Switched Path (LSP) established using LDP [RFC5036] or Segment Routing [RFC8402] or a traffic engineered LSP established using RSVP-TE [RFC3209] or SR-TE [RFC9256].

When a SRv6 PSN layer is used, a SRv6 service segment identifier (SID) as defined in [RFC8402] does provide the demultiplexing mechanism and definitions of Section 6 of [RFC9252] do apply. Both SRv6 service SIDs with the full IPv6 address format defined in [RFC8986] and compressed SIDs (C-SIDs) with format defined in [I-D.draft-ietf-spring-srv6-srh-compression] can be used.

Two new encapsulation behaviors H.Encaps.L1 and H.Encaps.L1.Red are defined in this document. The behavior procedures are applicable to both SIDs and C-SIDs.

The H.Encaps.L1 behavior encapsulates a frame received from an IWF in a IPv6 packet with an segment routing header (SRH). The received frame becomes the payload of the new IPv6 packet.

* The next header field of the SRH or last extension header present MUST be set to TBA1.

* The push of the SRH MAY be omitted when the SRv6 policy only contains one segment and there is no need to use any flag, tag, or TLV.

The H.Encaps.L1.Red behavior is an optimization of the H.Encaps.L1 behavior.

* H.Encaps.L1.Red reduces the length of the SRH by excluding the first SID in the SRH of the pushed IPv6 header. The first SID is only placed in the destination address field of the pushed IPv6 header.

* The push of the SRH MAY be omitted when the SRv6 policy only contains one segment and there is no need to use any flag, tag, or TLV.

Three new "Endpoint with decapsulation and bit-stream cross-connect" behaviors called End.DX1, End.DX1 with NEXT-CSID and End.DX1 with REPLACE-CSID are defined in this document. These new behaviors are variants of End.DX2 defined in [RFC8986] and all have the following procedures in common.

The End.DX1 SID MUST be the last segment in an SR Policy, and it is associated with a CE-bound IWF I. When N receives a packet destined to S and S is a local End.DX1 SID, N does the following:

```
S01. When an SRH is processed {
S02.   If (Segments Left != 0) {
S03.     Send an ICMP Parameter Problem to the Source Address
         with Code 0 (Erroneous header field encountered)
         and Pointer set to the Segments Left field,
         interrupt packet processing, and discard the packet.
S04.   }
S05.   Proceed to process the next header in the packet
S06. }
```

When processing the next (Upper-Layer) header of a packet matching a FIB entry locally instantiated as an End.DX1 SID, N does the following:

```
S01. If (Upper-Layer header type == TBA1 (bit-stream) ) {
S02.    Remove the outer IPv6 header with all its extension headers
S03.    Forward the remaining frame to the IWF I
S04. } Else {
S05.    Process as per {{Section 4.1.1 of RFC8986}}
S06. }
```

5.2. PLE Header

The PLE header MUST contain the PLE control word (4 bytes) and MUST include a fixed size RTP header [RFC3550]. The RTP header MUST immediately follow the PLE control word.

5.2.1. PLE Control Word

The format of the PLE control word is in line with the guidance in [RFC4385] and is shown in Figure 5.

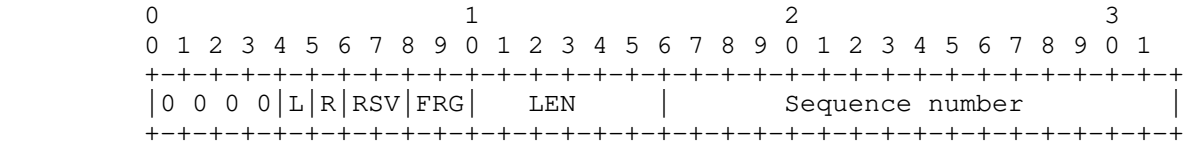


Figure 5: PLE Control Word

The bits 0..3 of the first nibble are set to 0 to differentiate a control word or Associated Channel Header (ACH) from an IP packet or Ethernet frame. The first nibble MUST be set to 0000b to indicate that this header is a control word as defined in Section 3 of [RFC4385].

The other fields in the control word are used as defined below:

* L

  Set by the PE to indicate that data carried in the payload is invalid due to an attachment circuit fault. The downstream PE MUST send appropriate replacement data. The NSP MAY inject an appropriate native fault propagation signal.

* R

Set by the downstream PE to indicate that the IWF experiences packet loss from the PSN or a server layer backward fault indication is present in the NSP. The R bit MUST be cleared by the PE once the packet loss state or fault indication has cleared.

* RSV

  These bits are reserved for future use. This field MUST be set to zero by the sender and ignored by the receiver.

* FRG

  These bits MUST be set to zero by the sender and ignored by the receiver as PLE does not use payload fragmentation.

* LEN

  In accordance to Section 3 of [RFC4385] the length field MUST always be set to zero as there is no padding added to the PLE packet. To detect malformed packets the default, preconfigured or signaled payload size MUST be assumed.

* Sequence number

  The sequence number field is used to provide a common PW sequencing function as well as detection of lost packets. It MUST be generated in accordance with the rules defined in Section 5.1 of [RFC3550] and MUST be incremented with every PLE packet being sent.

5.2.2. RTP Header

The RTP header MUST be included and is used for explicit transfer of timing information. The RTP header is purely a formal reuse and RTP mechanisms, such as header extensions, contributing source (CSRC) list, padding, RTP Control Protocol (RTCP), RTP header compression, Secure Realtime Transport Protocol (SRTP), etc., are not applicable to PLE VPWS.

The format of the RTP header is as shown in Figure 6.

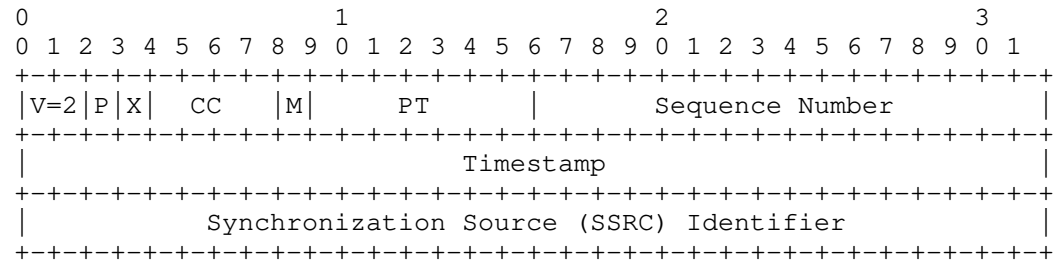


Figure 6: RTP Header

* V: Version

  The version field MUST be set to 2.

* P: Padding

  The padding flag MUST be set to zero by the sender and ignored by the receiver.

* X: Header extension

  The X bit MUST be set to zero by sender and ignored by receiver.

* CC: CSRC count

  The CC field MUST be set to zero by the sender and ignored by the receiver.

* M: Marker

  The M bit MUST be set to zero by the sender and ignored by the receiver.

* PT: Payload type

  A PT value MUST be allocated from the range of dynamic values defined in Section 6 of [RFC3551] for each direction of the VPWS. The same PT value MAY be reused both for direction and between different PLE VPWS.

* Sequence number

When using a 16 bit sequence number space, the sequence number in the RTP header MUST be equal to the sequence number in the PLE control word. When using a sequence number space of 32 bit, the initial value of the RTP sequence number MUST be 0 and incremented whenever the PLE control word sequence number cycles through from 0xFFFF to 0x0000.

* Timestamp

  Timestamp values are used in accordance with the rules established in [RFC3550]. For bit-streams up to 200 Gbps the frequency of the clock used for generating timestamps MUST be 125 MHz based on a the common clock I. For bit-streams above 200 Gbps the frequency MUST be 250 MHz.

* SSRC: Synchronization source

  The SSRC field MAY be used for detection of misconnections.

6. PLE Payload Layer

A bit-stream is mapped into a PLE packet with a fixed payload size which MUST be defined during VPWS setup, MUST be the same in both directions of the VPWS and MUST remain unchanged for the lifetime of the VPWS.

All PLE implementations MUST be capable of supporting the default payload size of 1024 bytes.

6.1. Basic Payload

The PLE payload is filled with incoming bits of the bit-stream starting from the most significant to the least significant bit without considering any structure of the bit-stream.

6.2. Byte aligned Payload

The PLE payload is filled in a byte aligned manner, where the order of the payload bytes corresponds to their order on the attachment circuit. Consecutive bits coming from the attachment circuit fill each payload byte starting from most significant bit to least significant. The PLE payload size MUST be an integer number of bytes.

7. PLE Operation

7.1. Common Considerations

A PLE VPWS can be established using manual configuration or leveraging mechanisms of a signaling protocol.

Furthermore emulation of bit-stream signals using PLE is only possible when the two attachment circuits of the VPWS are of the same service type (OC192, 10GBASE-R, ODU2, etc) and are using the same PLE payload type and payload size. This can be ensured via manual configuration or via the mechanisms of a signaling protocol.

PLE related control protocol extensions to LDP [RFC8077] or EVPN-VPWS [RFC8214] are out of scope for this document.

Extensions for EVPN-VPWS are proposed in [I-D.draft-schmutzer-bess-bitstream-vpws-signalling] and for LDP in [I-D.draft-schmutzer-pals-ple-signaling].

7.2. PLE IWF Operation

7.2.1. PSN-bound Encapsulation Behavior

After the VPWS is set up, the PSN-bound IWF does perform the following steps:

* Packetize the data received from the CE is into a fixed size PLE payloads

* Add PLE control word and RTP header with sequence numbers, flags and timestamps properly set

* Add the VPWS demultiplexer and PSN headers

* Transmit the resulting packets over the PSN

* Set L bit in the PLE control word whenever attachment circuit detects a fault

* Set R bit in the PLE control word whenever the local CE-bound IWF is in packet loss state

7.2.2. CE-bound Decapsulation Behavior

The CE-bound IWF is responsible for removing the PSN and VPWS demultiplexing headers, PLE control word and RTP header from the received packet stream and sending the bit-stream out via the local attachment circuit.

A de-jitter buffer MUST be implemented where the PLE packets are stored upon arrival. The size of this buffer SHOULD be locally configurable to allow accommodation of specific PSN packet delay variation expected.

The CE-bound IWF SHOULD use the sequence number in the control word to detect lost and misordered packets. It MAY use the sequence number in the RTP header for the same purposes. The CE-bound IWF MAY support re-ordering of packets received out of order. If the CE-bound IWF does not support re-ordering it MUST drop the misordered packets.

The payload of a lost or dropped packet MUST be replaced with equivalent amount of replacement data. The contents of the replacement data MAY be locally configurable. By default, all PLE implementations MUST support generation of "0xAA" as replacement data. The alternating sequence of 0s and 1s of the "0xAA" pattern does ensure clock synchronization is maintained and for 64B/66B code based services no invalid sync headers are generated. While sending out the replacement data, the IWF will apply a holdover mechanism to maintain the clock.

Whenever the VPWS is not operationally up, the CE-bound NSP function MUST inject the appropriate native downstream fault indication signal.

Whenever a VPWS comes up, the CE-bound IWF enters the intermediate state, will start receiving PLE packets and will store them in the jitter buffer. The CE-bound NSP function will continue to inject the appropriate native downstream fault indication signal until a pre-configured number of payload s stored in the jitter buffer.

After the pre-configured amount of payload is present in the jitter buffer the CE-bound IWF transitions to the normal operation state and the content of the jitter buffer is streamed out to the CE in accordance with the required clock. In this state the CE-bound IWF MUST perform egress clock recovery.

The recovered clock MUST comply with the jitter and wander requirements applicable to the type of attachment circuit, specified in:

* [G.825] and [G.823] for SDH

* [GR253] for SONET

* [G.8261] for synchronous Ethernet

* [G.8251] for OTN

Whenever the L bit is set in the PLE control word of a received PLE packet the CE-bound NSP function SHOULD inject the appropriate native downstream fault indication signal instead of streaming out the payload.

If the CE-bound IWF detects loss of consecutive packets for a pre-configured amount of time (default is 1 millisecond), it enters packet loss (PLOS) state and a corresponding defect is declared.

If the CE-bound IWF detects a packet loss ratio (PLR) above a configurable signal-degrade (SD) threshold for a configurable amount of consecutive 1-second intervals, it enters the degradation (DEG) state and a corresponding defect is declared. The SD-PLR threshold can be defined as percentage with the default being 15% or absolute packet count for finer granularity for higher rate interfaces. Possible values for consecutive intervals are 2..10 with the default 7.

While the PLOS defect is declared the CE-bound NSP function SHOULD inject the appropriate native downstream fault indication signal. Also the PSN-bound IWF SHOULD set the R bit in the PLE control word of every packet transmitted.

The CE-bound IWF does change from the PLOS to normal state after the pre-configured amount of payload has been received similarly to the transition from intermediate to normal state.

Whenever the R bit is set in the PLE control word of a received PLE packet the PLE performance monitoring statistics SHOULD get updated.

7.3. PLE Performance Monitoring

Attachment circuit performance monitoring SHOULD be provided by the NSP. The performance monitors are service specific, documented in related specifications and beyond the scope of this document.

The PLE IWF SHOULD provide functions to monitor the network performance to be inline with expectations of transport network operators.

The near-end performance monitors defined for PLE are as follows:

* ES-PLE : PLE Errored Seconds

* SES-PLE : PLE Severely Errored Seconds

* UAS-PLE : PLE Unavailable Seconds

Each second with at least one packet lost or a PLOS/DEG defect SHALL be counted as ES-PLE. Each second with a PLR greater than 15% or a PLOS/DEG defect SHALL be counted as SES-PLE.

UAS-PLE SHALL be counted after a configurable number of consecutive SES-PLE have been observed, and no longer counted after a configurable number of consecutive seconds without SES-PLE have been observed. Default value for each is 10 seconds.

Once unavailability is detected, ES and SES counts SHALL be inhibited up to the point where the unavailability was started. Once unavailability is removed, ES and SES that occurred along the clearing period SHALL be added to the ES and SES counts.

A PLE far-end performance monitor is providing insight into the CE-bound IWF at the far end of the PSN. The statistics are based on the PLE-RDI indication carried in the PLE control word via the R bit.

The PLE VPWS performance monitors are derived from the definitions in accordance with [G.826]

Performance monitoring data MUST be provided by the management interface and SHOULD be provided by a YANG model. The YANG model specification is out of scope for this document.

7.4. PLE Fault Management

Attachment circuit faults applicable to PLE are detected by the NSP, are service specific and are documented in relevant section of Section 4.

The two PLE faults, PLOS and DEG are detected by the IWF.

Faults MUST be time stamped as they are declared and cleared and fault related information MUST be provided by the management interface and SHOULD be provided by a YANG model. The YANG model specification is out of scope for this document.

8. QoS and Congestion Control

The PSN carrying PLE VPWS may be subject to congestion. Congestion considerations for PWs are described in Section 6.5 of [RFC3985].

PLE VPWS represent inelastic constant bit-rate (CBR) flows that cannot respond to congestion in a TCP-friendly manner as described in [RFC2914] and are sensitive to jitter, packet loss and packets received out of order.

The PSN providing connectivity between PE devices of a PLE VPWS has to ensure low jitter and low loss. The exact mechanisms used are beyond the scope of this document and may evolve over time. Possible options, but not exhaustively, are a Diffserv-enabled [RFC2475] PSN with a per domain behavior [RFC3086] supporting Expedited Forwarding [RFC3246]. Traffic-engineered paths through the PSN with bandwidth reservation and admission control applied. Or capacity over-provisioning.

9. Security Considerations

As PLE is leveraging VPWS as transport mechanism, the security considerations described [RFC3985] are applicable.

PLE does not enhance or detract from the security performance of the underlying PSN. It relies upon the PSN mechanisms for encryption, integrity, and authentication whenever required.

The PSN is assumed to be trusted and secure. Considerations about the MPLS core network outlined in [RFC4381] are applicable.

For MPLS based PSNs, one of the requirements for protecting the data plane is that the MPLS packets be accepted only from valid interfaces. For a PE, valid interfaces comprise links from other routers in the PE's own AS. For an ASBR, valid interfaces comprise links from other routers in the ASBR's own AS, and links from other ASBRs in ASes that have instances of a given PLE PWs. It is especially important in the case of multi-AS PLE PWs that one accepts PLE packets only from valid interfaces.

When a Segment Routing (SR) based PSN is used (MPLS or SRv6) the considerations in Section 8 of [RFC8402] and Section 9.3 of [RFC9252] are applicable.

PLE PWs share susceptibility to a number of pseudowire-layer attacks and will use whatever mechanisms for confidentiality, integrity, and authentication that are developed for general PWs. These methods are beyond the scope of this document.

Random initialization of sequence numbers, in both the control word and the RTP header, makes known-plaintext attacks more difficult.

Misconnection detection using the SSRC of the RTP header can increase the resilience to misconfiguration and some types of denial-of-service (DoS) attacks. A randomly chosen expected SSRC value does decrease the chance of a spoofing attack being successful. Control plane mechanisms for signaling the expected SSRC value are described in [I-D.draft-schmutzer-bess-bitstream-vpws-signalling] and [I-D.draft-schmutzer-pals-ple-signaling].

A data plane attack may force PLE packets to be dropped, re-ordered or delayed beyond the limit of the CE-bound IWF's dejitter buffer leading to either degradation or service disruption. Considerations outlined in [RFC9055] are a good reference.

Clock synchronization leveraging PTP is sensitive to Packet Delay Variation (PDV) and vulnerable to various threads and attack vectors. Considerations outlined in [RFC7384] should be taken into account.

10. IANA Considerations

10.1. Bit-stream Next Header Type

This document introduces a new value to be used in the next header field of an IPv6 header or any extension header indicating that the payload is a emulated bit-stream. IANA is requested to assign the following from the "Assigned Internet Protocol Numbers" registry (see https://www.iana.org/assignments/protocol-numbers/).

Decimal	Keyword	Protocol	IPv6 Extension Header	Reference
TBA1	BIT-EMU	Bit-stream Emulation	Y	this document

Table 1

10.2. SRv6 Endpoint Behaviors

This document introduces three new SRv6 Endpoint behaviors. IANA is requested to assign identifier values in the "SRv6 Endpoint Behaviors" sub-registry under "Segment Routing Parameters" registry.

Value	Hex	Endpoint Behavior	Reference
158	0x009E	End.DX1	this document
159	0x009F	End.DX1 with NEXT-CSID	this document
160	0x00A0	End.DX1 with REPLACE-CSID	this document

Table 2

11. Acknowledgements

The authors would like to thank all reviewers, contributors and the working group for reviewing this document and providing useful comments and suggestions.

12. References

12.1. Normative References

[I-D.draft-ietf-spring-srv6-srh-compression]
Cheng, W., Filsfils, C., Li, Z., Decraene, B., and F. Clad, "Compressed SRv6 Segment List Encoding", Work in Progress, Internet-Draft, draft-ietf-spring-srv6-srh-compression-19, 3 November 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-spring-srv6-srh-compression-19>.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, DOI 10.17487/RFC3551, July 2003, <https://www.rfc-editor.org/rfc/rfc3551>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/rfc/rfc8402>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/rfc/rfc8986>.

[RFC9252] Dawra, G., Ed., Talaulikar, K., Ed., Raszuk, R., Decraene, B., Zhuang, S., and J. Rabadan, "BGP Overlay Services Based on Segment Routing over IPv6 (SRv6)", RFC 9252, DOI 10.17487/RFC9252, July 2022, <https://www.rfc-editor.org/rfc/rfc9252>.

12.2. Informative References

[FC-PI-2] INCITS, "Information Technology - Fibre Channel Physical Interfaces - 2 (FC-PI-2)", 2006, <https://webstore.ansi.org/standards/incits/incits4042006>.

[FC-PI-5] INCITS, "Information Technology - Fibre Channel - Physical Interface-5 (FC-PI-5)", 2011, <https://webstore.ansi.org/standards/incits/incits4792011>.

[FC-PI-5am1] INCITS, "Information Technology - Fibre Channel - Physical Interface - 5/Amendment 1 (FC-PI-5/AM1)", 2016, <https://webstore.ansi.org/standards/incits/incits4792011am12016>.

[FC-PI-6] INCITS, "Information Technology - Fibre Channel - Physical Interface - 6 (FC-PI-6)", 2015, <https://webstore.ansi.org/standards/incits/incits5122015>.

[FC-PI-6P] INCITS, "Information Technology - Fibre Channel - Physical Interface - 6P (FC-PI-6P)", 2016, <https://webstore.ansi.org/standards/incits/incits5332016>.

[FC-PI-7] INCITS, "Information Technology â\200\223 Fibre Channel - Physical Interfaces - 7 (FC-PI-7)", 2021, <https://webstore.ansi.org/standards/iso/isoiec141651472021>.

[G.707] International Telecommunication Union (ITU), "Network node interface for the synchronous digital hierarchy (SDH)", January 2007, <https://www.itu.int/rec/T-REC-G.707>.

[G.709] International Telecommunication Union (ITU), "Interfaces for the optical transport network", June 2020, <https://www.itu.int/rec/T-REC-G.709>.

[G.823] International Telecommunication Union (ITU), "The control of jitter and wander within digital networks which are based on the 2048 kbit/s hierarchy", March 2000, <https://www.itu.int/rec/T-REC-G.823>.

[G.825] International Telecommunication Union (ITU), "The control of jitter and wander within digital networks which are based on the synchronous digital hierarchy (SDH)", March 2000, <https://www.itu.int/rec/T-REC-G.825>.

[G.8251] International Telecommunication Union (ITU), "The control of jitter and wander within the optical transport network (OTN)", November 2022, <https://www.itu.int/rec/T-REC-G.8251>.

[G.826] International Telecommunication Union (ITU), "End-to-end error performance parameters and objectives for international, constant bit-rate digital paths and connections", December 2002, <https://www.itu.int/rec/T-REC-G.826>.

[G.8261] International Telecommunication Union (ITU), "Timing and synchronization aspects in packet networks", August 2019, <https://www.itu.int/rec/T-REC-G.8261>.

[G.8261.1] International Telecommunication Union (ITU), "Packet delay variation network limits applicable to packet-based methods (Frequency synchronization)", February 2012, <https://www.itu.int/rec/T-REC-G.8261.1>.

[G.8262] International Telecommunication Union (ITU), "Timing characteristics of synchronous equipment slave clock", November 2018, <https://www.itu.int/rec/T-REC-G.8262>.

[G.8265.1] International Telecommunication Union (ITU), "Precision time protocol telecom profile for frequency synchronization", November 2022, <https://www.itu.int/rec/T-REC-G.8265.1>.

[GR253] Telcordia, "SONET Transport Systems - Common Generic Criteria", October 2009.

[I-D.draft-schmutzer-bess-bitstream-vpws-signalling]
Gringeri, S., Whittaker, J., Schmutzer, C., Vasudevan, B., and P. Brissette, "Ethernet VPN Signalling Extensions for Bit-stream VPWS", Work in Progress, Internet-Draft, draft-schmutzer-bess-bitstream-vpws-signalling-02, 18 October 2024, <https://datatracker.ietf.org/doc/html/draft-schmutzer-bess-bitstream-vpws-signalling-02>.

[I-D.draft-schmutzer-pals-ple-signaling]
Schmutzer, C., "LDP Extensions to Support Private Line Emulation (PLE)", Work in Progress, Internet-Draft, draft-schmutzer-pals-ple-signaling-02, 20 October 2024, <https://datatracker.ietf.org/doc/html/draft-schmutzer-pals-ple-signaling-02>.

[IEEE802.3]
IEEE, "IEEE Standard for Ethernet", May 2022, <https://standards.ieee.org/ieee/802.3/10422/>.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <https://www.rfc-editor.org/rfc/rfc2475>.

[RFC2914] Floyd, S., "Congestion Control Principles", BCP 41, RFC 2914, DOI 10.17487/RFC2914, September 2000, <https://www.rfc-editor.org/rfc/rfc2914>.

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <https://www.rfc-editor.org/rfc/rfc3031>.

[RFC3086] Nichols, K. and B. Carpenter, "Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification", RFC 3086, DOI 10.17487/RFC3086, April 2001, <https://www.rfc-editor.org/rfc/rfc3086>.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/rfc/rfc3209>.

[RFC3246] Davie, B., Charny, A., Bennet, J.C.R., Benson, K., Le Boudec, J.Y., Courtney, W., Davari, S., Firoiu, V., and D. Stiliadis, "An Expedited Forwarding PHB (Per-Hop Behavior)", RFC 3246, DOI 10.17487/RFC3246, March 2002, <https://www.rfc-editor.org/rfc/rfc3246>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/rfc/rfc3550>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/rfc/rfc3711>.

[RFC3985] Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <https://www.rfc-editor.org/rfc/rfc3985>.

[RFC4197] Riegel, M., Ed., "Requirements for Edge-to-Edge Emulation of Time Division Multiplexed (TDM) Circuits over Packet Switching Networks", RFC 4197, DOI 10.17487/RFC4197, October 2005, <https://www.rfc-editor.org/rfc/rfc4197>.

[RFC4381] Behringer, M., "Analysis of the Security of BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4381, DOI 10.17487/RFC4381, February 2006, <https://www.rfc-editor.org/rfc/rfc4381>.

[RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <https://www.rfc-editor.org/rfc/rfc4385>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/rfc/rfc4443>.

[RFC4448] Martini, L., Ed., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, DOI 10.17487/RFC4448, April 2006, <https://www.rfc-editor.org/rfc/rfc4448>.

[RFC4553] Vainshtein, A., Ed. and YJ. Stein, Ed., "Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)", RFC 4553, DOI 10.17487/RFC4553, June 2006, <https://www.rfc-editor.org/rfc/rfc4553>.

[RFC4842] Malis, A., Pate, P., Cohen, R., Ed., and D. Zelig, "Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) Circuit Emulation over Packet (CEP)", RFC 4842, DOI 10.17487/RFC4842, April 2007, <https://www.rfc-editor.org/rfc/rfc4842>.

[RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <https://www.rfc-editor.org/rfc/rfc4875>.

[RFC4906] Martini, L., Ed., Rosen, E., Ed., and N. El-Aawar, Ed., "Transport of Layer 2 Frames Over MPLS", RFC 4906, DOI 10.17487/RFC4906, June 2007, <https://www.rfc-editor.org/rfc/rfc4906>.

[RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed., "LDP Specification", RFC 5036, DOI 10.17487/RFC5036, October 2007, <https://www.rfc-editor.org/rfc/rfc5036>.

[RFC7212] Frost, D., Bryant, S., and M. Bocci, "MPLS Generic Associated Channel (G-ACh) Advertisement Protocol", RFC 7212, DOI 10.17487/RFC7212, June 2014, <https://www.rfc-editor.org/rfc/rfc7212>.

[RFC7384] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/RFC7384, October 2014, <https://www.rfc-editor.org/rfc/rfc7384>.

[RFC8077] Martini, L., Ed. and G. Heron, Ed., "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", STD 84, RFC 8077, DOI 10.17487/RFC8077, February 2017, <https://www.rfc-editor.org/rfc/rfc8077>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/rfc/rfc8214>.

[RFC9055] Grossman, E., Ed., Mizrahi, T., and A. Hacker, "Deterministic Networking (DetNet) Security Considerations", RFC 9055, DOI 10.17487/RFC9055, June 2021, <https://www.rfc-editor.org/rfc/rfc9055>.

[RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <https://www.rfc-editor.org/rfc/rfc9256>.

[RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <https://www.rfc-editor.org/rfc/rfc9293>.

Contributors

Andreas Burk
1&1 Versatel
Email: andreas.burk@magenta.de

Faisal Dada
AMD
Email: faisal.dada@amd.com

Gerald Smallegange
Ciena Corporation
Email: gsmalleg@ciena.com

Erik van Veelen
Aimvalley
Email: erik.vanveelen@aimvalley.com

Luca Della Chiesa
Cisco Systems, Inc.
Email: ldellach@cisco.com

Nagendra Kumar Nainar
Cisco Systems, Inc.
Email: naikumar@cisco.com

Carlos Pignataro
North Carolina State University
Email: cmpignat@ncsu.edu

Authors' Addresses

Steven Gringeri
Verizon
Email: steven.gringeri@verizon.com

Jeremy Whittaker
Verizon
Email: jeremy.whittaker@verizon.com

Nicolai Leymann
Deutsche Telekom
Email: N.Leymann@telekom.de

Christian Schmutzer (editor)
Cisco Systems, Inc.
Email: cschmutz@cisco.com

Chris Brown
Ciena Corporation
Email: cbrown@ciena.com

Path Computation Element
Internet-Draft
Updates: 5440, 8231, 8233, 8281, 8623, 8664,
8685, 8697, 8745, 8733, 8779, 8780,
8800, 8934, 9050, 9059, 9168, 9357,
9504, 9603, 9604 (if approved)
Intended status: Standards Track
Expires: 16 May 2025

D. Dhody
Huawei
A. Farrel
Old Dog Consulting

12 November 2024

Update to the IANA PCE Communication Protocol (PCEP) Registration Procedures and Allowing Experimental Error Codes
draft-ietf-pce-iana-update-03

Abstract

This document updates the registration procedure within the IANA "Path Computation Element Protocol (PCEP) Numbers" group of registries. This specification changes some of the registries with Standards Action to IETF Review as defined in RFC 8126. This memo updates RFCs 8231, 8233, 8281, 8623, 8664, 8685, 8697, 8733, 8745, 8779, 8780, 8800, 8934, 9050, 9059, 9168, 9357, 9504, 9603, and 9604 for the same.

Designating "experimental use" sub-ranges within code point registries is often beneficial for protocol experimentation in controlled environments. Although the registries for PCEP messages, objects, and TLV types have sub-ranges assigned for Experimental Use, the registry for PCEP Error-Types and Error-values currently does not. This document updates RFC 5440 by designating a specific range of PCEP Error-Types for Experimental Use.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Path Computation Element Working Group mailing list (pce@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/pce/.

Source for this draft and an issue tracker can be found at https://github.com/ietf-wg-pce/draft-ietf-pce-iana-update.

Status of This Memo

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

This Internet-Draft will expire on 16 May 2025.

Table of Contents

1. Introduction

The IANA "Path Computation Element Protocol (PCEP) Numbers" registry group was populated by several RFCs produced by the Path Computation Element (PCE) working group. Most of the registries include the "IETF Review" [RFC8126] as registration procedures. There are a few registries that use "Standards Action". Thus, the values in those registries can be assigned only through the Standards Track or Best Current Practice RFCs in the IETF Stream. This memo changes the policy from Standards Action to IETF Review to allow any type of RFC under the IETF stream to make the allocation request.

Further, in Section 9 of [RFC5440], IANA assigns values to the PCEP parameters. The allocation policy for each of these parameters specified in RFC 5440 is IETF Review [RFC8126]. In consideration of the benefits of conducting experiments with PCEP and the utility of experimental codepoints [RFC3692], codepoint ranges for PCEP messages, objects, and TLV types for Experimental Use [RFC8126] are designated in [RFC8356]. However, protocol experiments may also need to return protocol error messages indicating experiment-specific error cases. It will often be the case that previously assigned error codes (in the PCEP-ERROR Object Error Types and Values sub-registry) can be used to indicate the error cases within an experiment, but there may also be cases where new, experimental error codes are needed. In order to run experiments, it is important that the codepoint values used in the experiments do not collide with existing codepoints or any future allocations. This document updates [RFC5440] by changing the allocation policy for the registry of PCEP Error-Types to mark some of the codepoints as assigned for Experimental Use. As stated in [RFC3692], experiments using these codepoints are not intended to be used in general deployments, and due care must be taken to ensure that two experiments using the same codepoints are not run in the same environment.

2. Standards Action PCEP Registries Affected

The following table lists the "Path Computation Element Protocol (PCEP) Numbers" registries whose registration policy will be changed from Standards Action to IETF Review. Affected registries will list this document as a reference. Where this change is applied to a specific range of values within the particular registry, that range is given in the Remarks column.

Registry	RFC	Remarks
BU Object Type Field	[RFC8233]	
LSP Object Flag Field	[RFC8231]	
STATEFUL-PCE-CAPABILITY TLV Flag Field	[RFC8231]	
LSP-ERROR-CODE TLV Error Code Field	[RFC8231]	
SRP Object Flag Field	[RFC8281]	
SR-ERO Flag Field	[RFC8664]	
PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators	[RFC8664]	
SR Capability Flag Field	[RFC8664]	
WA Object Flag Field	[RFC8780]	
Wavelength Restriction TLV Action Values	[RFC8780]	
Wavelength Allocation TLV Flag Field	[RFC8780]	
S2LS Object Flag Field	[RFC8623]	
H-PCE-CAPABILITY TLV Flag Field	[RFC8685]	
H-PCE-FLAG TLV Flag Field	[RFC8685]	
ASSOCIATION Flag Field	[RFC8697]	
ASSOCIATION Type Field	[RFC8697]	
AUTO-BANDWIDTH-CAPABILITY TLV Flag Field	[RFC8733]	
Path Protection Association Group TLV Flag Field	[RFC8745]	
Generalized Endpoint Types	[RFC8779]	0-244
GMPLS-CAPABILITY TLV Flag Field	[RFC8779]	
DISJOINTNESS-CONFIGURATION TLV Flag	[RFC8800]	

Field		
SCHED-PD-LSP-ATTRIBUTE TLV Opt Field	[RFC8934]	
Schedule TLVs Flag Field	[RFC8934]	
FLOWSPEC Object Flag Field	[RFC9168]	
Bidirectional LSP Association Group TLV Flag Field	[RFC9059]	
PCECC-CAPABILITY sub-TLV	[RFC9050]	
CCI Object Flag Field for MPLS Label	[RFC9050]	
TE-PATH-BINDING TLV BT Field	[RFC9050]	
TE-PATH-BINDING TLV Flag Field	[RFC9604]	
LSP-EXTENDED-FLAG TLV Flag Field	[RFC9357]	
LSP Exclusion Subobject Flag Field	[RFC9504]	
SRv6-ERO Flag Field	[RFC9603]	
SRv6 Capability Flag Field	[RFC9603]	

Table 1: PCEP Registries Affected

Future registries in the "Path Computation Element Protocol (PCEP) Numbers" registry group should prefer to use "IETF Review" over "Standards Action".

3. Experimental Error-Types

This document requests IANA for the designation of four PCEP Error-Type codepoints (252-255) for Experimental Use.

IANA maintains a registry group called "Path Computation Element Protocol (PCEP) Numbers" with a registry named "PCEP-ERROR Object Error Types and Values". IANA is requested to change the assignment policy for this registry to read:

* Error-Types
  - 0-251 : IETF Review

- 252-255 : Experimental Use

* Error-value

  - For all IETF Review Error-Types : IETF Review

  - For all Experimental Use Error-Types : Experimental Use

Additionally, IANA is requested to make an entry in the table as follows:

Error-Type	Meaning	Error-value	Reference
252-255	Experimental Use	0-255 Experimental Use	This I-D

Table 2

3.1. Advice on Experimentation

An experiment that wishes to return experimental error codes should use one of the experimental Error-Type values as defined in this document. The experiment should agree, between all participating parties, on which Error-Type to use and which Error-values to use within that Error-Type. The experiment will describe what the meanings of those Error-Type / Error-value pairs are. Those Error-Type and Error-values should not be recorded in any public (especially any IETF) documentation. Textual or symbolic names for the Error-Types and Error-values may be used to help keep the documentation clear.

If multiple experiments are taking place at the same time using the same implementations, care must be taken to keep the sets of Error-Type / Error-value distinct.

Note that there is no scope for experimental Error-values within existing non-experimental Error-Types. This reduces the complexity of the registry and implementations. Experiments should place all experimental Error-values under the chosen experimental Error-Types.

If, at some future time, the experiment is declared a success and moved to IETF work targeting publication on the Standards Track, each pair of Error-Type / Error-value will need to be assigned by IANA from the registry. In some cases, this will involve assigning a new Error-Type with its subtended Error-values. In other cases, use may be made of an existing Error-Type with new subtended Error-values

being assigned. The resulting change to code in an implementation is as simple as changing the numeric values of the Error-Types and Error-values.

3.2. Handling of Unknown Experimentation

A PCEP implementation that receives an experimental Error-Type in a PCEP message and does not recognize the Error-Type (i.e., is not part of the experiment) will treat the error as it would treat any other unknown Error-Type (such as from a new protocol extension). An implementation that is notified of a PCEP error will normally close the PCEP session (see [RFC5440]). In general, PCEP implementations are not required to take specific action based on Error-Types but may log the errors for diagnostic purposes.

An implementation that is part of an experiment may receive an experimental Error-Type, but not recognize the Error-value. This could happen because of any of:

* A faulty implementation.

* Two implementations not being synchronized with respect to which Error-values to use in the experiment.

* More than one experiment being run at the same time.

As with unknown Error-Types, an implementation receiving an unknown Error-value is not expected to do more than log the received error and may close the PCEP session.

4. IANA Considerations

This memo is entirely about updating the IANA "Path Computation Element Protocol (PCEP) Numbers" registry.

5. Security Considerations

This memo does not change the Security Considerations for any of the updated RFCs. Refer to [RFC5440] and [I-D.ietf-pce-pceps-tls13] for further details of the specific security measures applicable to PCEP.

[RFC3692] asserts that the existence of experimental codepoints introduces no new security considerations. However, implementations accepting experimental error codepoints need to consider how they parse and process them in case they come, accidentally, from another experiment. Further, an implementation accepting experimental codepoints needs to consider the security aspects of the experimental extensions. [RFC6709] provides various design considerations for protocol extensions (including those designated as experimental).

6. References

6.1. Normative References

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/rfc/rfc5440>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/rfc/rfc8126>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/rfc/rfc8231>.

[RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki, "Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September 2017, <https://www.rfc-editor.org/rfc/rfc8233>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/rfc/rfc8281>.

[RFC8356] Dhody, D., King, D., and A. Farrel, "Experimental Codepoint Allocation for the Path Computation Element Communication Protocol (PCEP)", RFC 8356, DOI 10.17487/RFC8356, March 2018, <https://www.rfc-editor.org/rfc/rfc8356>.

[RFC8623] Palle, U., Dhody, D., Tanaka, Y., and V. Beeram, "Stateful Path Computation Element (PCE) Protocol Extensions for Usage with Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 8623, DOI 10.17487/RFC8623, June 2019, <https://www.rfc-editor.org/rfc/rfc8623>.

[RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing", RFC 8664, DOI 10.17487/RFC8664, December 2019, <https://www.rfc-editor.org/rfc/rfc8664>.

[RFC8685] Zhang, F., Zhao, Q., Gonzalez de Dios, O., Casellas, R., and D. King, "Path Computation Element Communication Protocol (PCEP) Extensions for the Hierarchical Path Computation Element (H-PCE) Architecture", RFC 8685, DOI 10.17487/RFC8685, December 2019, <https://www.rfc-editor.org/rfc/rfc8685>.

[RFC8697] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "Path Computation Element Communication Protocol (PCEP) Extensions for Establishing Relationships between Sets of Label Switched Paths (LSPs)", RFC 8697, DOI 10.17487/RFC8697, January 2020, <https://www.rfc-editor.org/rfc/rfc8697>.

[RFC8733] Dhody, D., Ed., Gandhi, R., Ed., Palle, U., Singh, R., and L. Fang, "Path Computation Element Communication Protocol (PCEP) Extensions for MPLS-TE Label Switched Path (LSP) Auto-Bandwidth Adjustment with Stateful PCE", RFC 8733, DOI 10.17487/RFC8733, February 2020, <https://www.rfc-editor.org/rfc/rfc8733>.

[RFC8745] Ananthakrishnan, H., Sivabalan, S., Barth, C., Minei, I., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extensions for Associating Working and Protection Label Switched Paths (LSPs) with Stateful PCE", RFC 8745, DOI 10.17487/RFC8745, March 2020, <https://www.rfc-editor.org/rfc/rfc8745>.

[RFC8779] Margaria, C., Ed., Gonzalez de Dios, O., Ed., and F. Zhang, Ed., "Path Computation Element Communication Protocol (PCEP) Extensions for GMPLS", RFC 8779, DOI 10.17487/RFC8779, July 2020, <https://www.rfc-editor.org/rfc/rfc8779>.

[RFC8780] Lee, Y., Ed. and R. Casellas, Ed., "The Path Computation Element Communication Protocol (PCEP) Extension for Wavelength Switched Optical Network (WSON) Routing and Wavelength Assignment (RWA)", RFC 8780, DOI 10.17487/RFC8780, July 2020, <https://www.rfc-editor.org/rfc/rfc8780>.

[RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <https://www.rfc-editor.org/rfc/rfc8800>.

[RFC8934] Chen, H., Ed., Zhuang, Y., Ed., Wu, Q., and D. Ceccarelli, "PCE Communication Protocol (PCEP) Extensions for Label Switched Path (LSP) Scheduling with Stateful PCE", RFC 8934, DOI 10.17487/RFC8934, October 2020, <https://www.rfc-editor.org/rfc/rfc8934>.

[RFC9050] Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) Procedures and Extensions for Using the PCE as a Central Controller (PCECC) of LSPs", RFC 9050, DOI 10.17487/RFC9050, July 2021, <https://www.rfc-editor.org/rfc/rfc9050>.

[RFC9059] Gandhi, R., Ed., Barth, C., and B. Wen, "Path Computation Element Communication Protocol (PCEP) Extensions for Associated Bidirectional Label Switched Paths (LSPs)", RFC 9059, DOI 10.17487/RFC9059, June 2021, <https://www.rfc-editor.org/rfc/rfc9059>.

[RFC9168] Dhody, D., Farrel, A., and Z. Li, "Path Computation Element Communication Protocol (PCEP) Extension for Flow Specification", RFC 9168, DOI 10.17487/RFC9168, January 2022, <https://www.rfc-editor.org/rfc/rfc9168>.

[RFC9357] Xiong, Q., "Label Switched Path (LSP) Object Flag Extension for Stateful PCE", RFC 9357, DOI 10.17487/RFC9357, February 2023, <https://www.rfc-editor.org/rfc/rfc9357>.

[RFC9504] Lee, Y., Zheng, H., Gonzalez de Dios, O., Lopez, V., and Z. Ali, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE Usage in GMPLS-Controlled Networks", RFC 9504, DOI 10.17487/RFC9504, December 2023, <https://www.rfc-editor.org/rfc/rfc9504>.

[RFC9603] Li, C., Ed., Kaladharan, P., Sivabalan, S., Koldychev, M., and Y. Zhu, "Path Computation Element Communication Protocol (PCEP) Extensions for IPv6 Segment Routing", RFC 9603, DOI 10.17487/RFC9603, July 2024, <https://www.rfc-editor.org/rfc/rfc9603>.

[RFC9604] Sivabalan, S., Filsfils, C., Tantsura, J., Previdi, S., and C. Li, Ed., "Carrying Binding Label/SID in PCE-Based Networks", RFC 9604, DOI 10.17487/RFC9604, August 2024, <https://www.rfc-editor.org/rfc/rfc9604>.

6.2. Informative References

[I-D.ietf-pce-pceps-tls13] Dhody, D., Turner, S., and R. Housley, "Updates for PCEPS: TLS Connection Establishment Restrictions", Work in Progress, Internet-Draft, draft-ietf-pce-pceps-tls13-04, 9 January 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-pceps-tls13-04>.

[RFC3692] Narten, T., "Assigning Experimental and Testing Numbers Considered Useful", BCP 82, RFC 3692, DOI 10.17487/RFC3692, January 2004, <https://www.rfc-editor.org/rfc/rfc3692>.

[RFC6709] Carpenter, B., Aboba, B., Ed., and S. Cheshire, "Design Considerations for Protocol Extensions", RFC 6709, DOI 10.17487/RFC6709, September 2012, <https://www.rfc-editor.org/rfc/rfc6709>.

Appendix A. Acknowledgments

Thanks to John Scudder for the initial discussion behind this document. Thanks to Ketan Talaulikar, Andrew Stone, Samuel Sidor, Quan Xiong, Cheng Li, and Aijun Wang for the review comments. Thanks to Carlos Pignataro for the OPSDIR review. Thanks to Meral Shirazipour for GENART review. Thanks to Paul Kyzivat for ArtArt review. Thanks to Alexey Melnikov for SECDIR review.

Appendix B. Rationale for updating all registries with Standards Action

This specification updates all the registries with the "Standards Action" policy. WG considered keeping "Standards Action" for some registries such as flag fields with limited bits, where the space is tight but decided against it. The WG's last call and IETF's last call process should be enough to handle the case of frivolous experiments taking over the few code points. The working group could also create a new protocol field and registry for future use as done

in the past (see [RFC9357]).

Appendix C. Consideration of RFC 8356

It is worth noting that [RFC8356] deliberately chose to make experimental codepoints available only in the PCEP messages, objects, and TLV type registries. Appendix A of that document gives a brief explanation of why that decision was taken stating that:

> The justification for this decision is that, if an experiment finds that it wants to use a new codepoint in another PCEP sub-registry, it can implement the same function using a new experimental object or TLV instead.

While it is true that an experimental implementation could assign an experimental PCEP object and designate it the "experimental errors object", using it to carry arbitrary contents including experimental error codes, such an approach would cause unnecessary divergence in the code. The allowance of experimental Error-Types is a better approach that will more easily enable the migration of successful experiments onto the Standards Track.

Appendix D. Contributor

Haomian Zheng
Huawei Technologies
Email: zhenghaomian@huawei.com

Authors' Addresses

Dhruv Dhody
Huawei
India
Email: dhruv.ietf@gmail.com

Adrian Farrel
Old Dog Consulting
Email: adrian@olddog.co.uk

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 30 May 2025

H. Bidgoli, Ed.
Nokia
S. Venaas
Cisco System, Inc.
M. Mishra
Cisco System
Z. Zhang
Juniper Networks
M. McBride
Futurewei Technologies Inc.

26 November 2024

Protocol Independent Multicast Light (PIM Light)
draft-ietf-pim-light-10

Abstract

This document specifies Protocol Independent Multicast Light (PIM Light) and PIM Light Interface (PLI) which does not need PIM Hello message to accept PIM Join/Prune messages. PLI can signal multicast states over networks that can not support full PIM neighbor discovery, as an example BIER networks that are connecting two or more PIM domains. This document outlines the PIM Light protocol and procedures to ensure loop-free multicast traffic between two or more PIM Light routers.

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 30 May 2025.

Copyright Notice

Table of Contents

1. Introduction

This document specifies the Protocol Independent Multicast Light (PIM Light) and PIM Light Interface (PLI) procedures. PLI is a new type of PIM interface that allows signaling of PIM Join/Prune packets without full PIM neighbor discovery. PLI is useful in scenarios where multicast states needs to be signalled over networks or media that cannot support full PIM neighborship between routers or alternatively full PIM neighborship is not desired. These type of networks or medias are addressed as a PIM Light Domain within this document. Lack of full PIM neighborship will remove some PIM functionality as explained in section 3.2 of this document. PIM Light only supports Protocol Independent Multicast Sparse Mode (PIM-SM) protocol including PIM Source-Specific Multicast (PIM-SSM) as per [RFC7761]. The document details procedures and considerations needed for PIM Light and PLI to ensure efficient routing of multicast groups

for specific deployment environments.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2.1. Definitions

This document uses definitions used in Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification [RFC7761]

3. PIM Light Interface

RFC [RFC7761] section 4.3.1 describes the PIM neighbor discovery via Hello messages. In section 4.5 it describes that if a router receives a Join/Prune message from a particular IP source address and it has not seen a PIM Hello message from that source address, then the Join/Prune message SHOULD be discarded without further processing.

In certain scenarios, it is desirable to establish multicast states between two layer-3 adjacent routers without forming a PIM neighborship. This can be necessary for various reasons, such as signaling multicast states upstream between multiple PIM domains over a network that is not optimized for PIM or does not necessitate PIM Neighbor establishment. For example, in a Bit Index Explicit Replication (BIER) [RFC8279] networks connecting multiple PIM domains, where PIM Join/Prune messages are tunneled via BIER as specified in [draft-ietf-bier-pim-signaling].

A PIM Light Interface (PLI) accepts Join/Prune messages from an unknown PIM router without requiring a PIM Hello message from the router. The absence of Hello messages on a PLI means there is no mechanism to discover neighboring PIM routers or their capabilities, nor to execute basic algorithms such as Designated Router (DR) election [RFC7761]. Consequently, the PIM Light router does not create any general-purpose state for neighboring PIM routers and only processes Join/Prune messages from downstream routers in its multicast routing table. Processing these Join/Prune messages will introduce multicast states in a PIM Light router.

Due to these constraints, a PLI should be deployed in very specific scenarios where PIM-SM is not suitable. The applications or the networks that PLIs are deployed on MUST ensure there is no multicast

packet duplication, such as multiple upstream routers sending the same multicast stream to a single downstream router. As an example the implementation should ensure that DR election is done on upstream Redundant PIM routers that are at the edge of the PIM Light Domain to ensure a single Designated Router to forward the PIM Join message from reviver to the Source.

3.1. PLI supported Messages

IANA [iana_pim-parameters_message-types], lists the PIM supported message types. PIM Light only supports the following message types from the table "PIM Message Types"

1. type 3 (Join/Prune) from the ALL-PIM-ROUTERS message types listed in [RFC7761].

2. type 1 (Register)

3. type 2 (Register Stop)

4. type 8 (Candidate RP Advertisement)

5. type 13 (PIM Packed Null-Register)

6. type 13.1 (PIM Packed Register-Stop)

7. Any future PIM message types that use unicast destination IP.

No other message types are supported for PIM Light and SHOULD NOT be process if received on a PLI.

3.2. Absence of Hello Message consideration

In a PIM Light domain, the following considerations should be taken into account due to the lack of processing Hello messages.

3.2.1. Join Attribute

Since a PLI does not process PIM Hello messages, it also does not support the join attributes option in PIM Hello as specified in [RFC5384]. As such, PIM Light is unaware of its neighbor’s capability to process join attributes and it SHOULD NOT process a join message containing it.

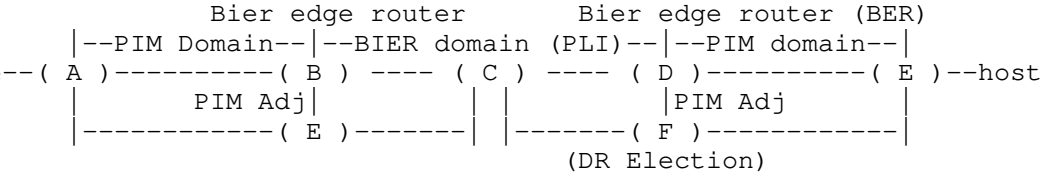
For a PLI to send and process a join attributes there can be two cases:

1. It should be configured with appropriate join attribute type that the PLI is capable of processing as per [iana_pim-parameters_join-attribute-types] table.

2. Separate IETF drafts or RFCs may dictate that certain join attributes are allowed to be used without explicit configuration of the PLI in certain scenarios. The details are left to those drafts or RFCs.

3.2.2. DR Election

Due to the absence of Hello messages, DR Election is not supported on a PIM Light router. The network design must ensure DR Election occurs within the PIM domain, assuming the PIM Light domain interconnects PIM domains.



For instance, in a BIER domain connecting two PIM networks, a PLI can be used between BIER edge routers solely for multicast state communication and transmit only PIM Join/Prune messages. To prevent multicast stream duplication, PIM routers on either side of the BIER domain SHOULD establish PIM adjacency as per [RFC7761] to ensure DR election at the edge of BIER domain. An example DR election could be DR election between router D and F in above figure. When the Join or Prune message arrives from a PIM domain to the down stream BIER edge router, it can be send over the BIER tunnel to the upstream BIER edge router only via the designated router.

3.2.3. PIM Assert

In scenarios where multiple PIM routers peer over a shared LAN or a Point-to-Multipoint medium, more than one upstream router may have valid forwarding state for a packet, potentially causing packet duplication. PIM Assert is used to select a single transmitter when such duplication is detected. According to [RFC7761], PIM Assert should only be accepted from a known PIM neighbor.

In PIM Light implementations, care must be taken to avoid duplicate streams arriving from multiple upstream PIM Light routers to a single downstream PIM Light router. If network design constraints prevent this, the implemented network architecture should take measures to avoid traffic duplication. For example, in a PIM Light over a BIER

domain scenario, downstream IBBR (Ingress BIER Border Router) in a BIER domain can identify the nearest EBBRs (Egress BIER Border Routers) to the source using the Shortest Path First (SPF) algorithm with a post-processing as described in [draft-ietf-bier-pim-signaling] Appendix A.1. If the downstream IBBR identifies two EBBRs, it can select one using a unique IP selection algorithm, such as choosing the EBBR with the lowest or highest IP address. If the selected EBBR goes offline, the downstream router can use the next EBBR based on the IP selection algorithm, which is beyond the scope of this document.

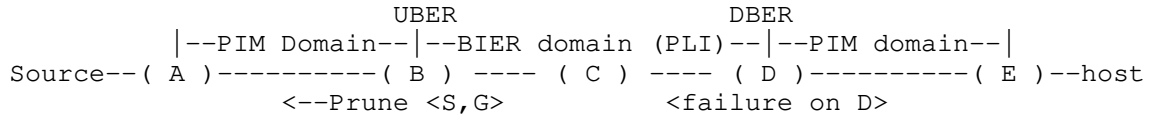
3.3. PLI Configuration

Since a PLI doesn't require PIM Hello Messages and PIM neighbor adjacency is not checked for arriving Join/Prune messages, there needs to be a mechanism to enable PLI on interfaces. Only when PLI is enabled on an interface, arriving Join/Prune messages SHOULD be processed, otherwise they SHOULD be dropped. While on some logical interfaces PLI maybe enabled automatically or via an underlying mechanism, as an example the logical interface connecting two or more BIER edge routers in a BIER sub-domain [draft-ietf-bier-pim-signaling].

3.4. Failures in PLR domain

Because the Hello messages are not processed on the PLI, PIM Light Interface failures may not be discovered in a PIM Light domain and multicast routes will not be pruned toward the source on the PIM Light domain, leaving the upstream routers continuously sending multicast streams until the out going interface (OIF) expires.

Other protocols can be used to detect these failures in the PIM Light domain and they can be implementation specific. As an example, the interface that PIM Light is configured on can be protected via Bidirectional Forwarding Detection (BFD) or similar technology. If BFD to the far-end PLI goes down, and the PIM Light Router is upstream and has an OIF for a multicast route <S,G>, PIM should remove that PLI from its OIF list.



In another example, where the PLI is configured automatically between the BIER Edge Routers (BER), when the downstream BIER Edge Router (DBER) is no longer reachable on the upstream BIER Edge Router (UBER), the UBER which is also a PIM Light Router can prune the <S,G> advertised toward the source on the PIM domain to stop the transmission of the multicast stream.

3.5. Reliable Transport Mechanism for PIM LIGHT

[RFC6559] defines a reliable transport mechanism for PIM transmission of Join/Prune messages, using either TCP or SCTP as transport protocol. For TCP, PIM over reliable transport (PORT) uses port 8471 which is assigned by IANA. SCTP is explained in [RFC9260], and it is used as a second option for PORT. [RFC6559] mentions that when a router is configured to use PIM over TCP on a given interface, it MUST include the PIM-over-TCP-Capable Hello Option in its Hello messages for that interface. The same is true for SCTP and the router must include PIM-over-SCTP-Capable Hello Option in its Hello messsage on that interface.

These Hello options contain a Connection ID which is an IPv4 or IPv6 address used to establish the SCTP or TCP connection. For PORT using TCP, the connection ID is used for determining which peer is doing a active transport open to the neighbor and which peer is doing passive transport open, as per section 4 of [RFC6559]

When the router is using SCTP, the Connection ID IP address comparison need not be done since the SCTP protocol can handle call collision.

PIM Light lacks Hello messages, the PLI can be configured with the Connection ID IPv4 or IPv6 addresses used to establish the SCTP or TCP connection. For PIM Light using TCP PORT option each end of the PLI must be explicitly and correct configured as being active transport open or passive transport open to ensure handle call collision is avoided.

3.6. PIM Variants not supported

The following PIM variants are not supported with PIM Light and not covered by this document:

1. Protocol Independent Multicast - Dense Mode (PIM-DM)[RFC3973]

2. Bidirectional Protocol Independent Multicast (BIDIR-PIM) [RFC5015]

4. IANA Considerations

There are no new IANA considerations for this document.

5. Security Considerations

Since PIM Light does not require PIM Hello messages and does not verify PIM neighbor adjacency for incoming Join/Prune messages, it is crucial for security reasons, that the implementation ensures only Join/Prune messages arriving on a configured PLI are processed. Any Join/Prune messages received on an interface that is not configured as a PLI MUST be discarded and not processed. Additionally, as a secondary line of defense, route policies SHOULD be implemented to process only the Join/Prune messages associated with the desired (S,G) pairs, while all other (S,G) pairs MUST be discarded and not processed.

Furthermore, because PIM Light can be used for signaling Source-Specific and Sparse Mode Join/Prune messages, the security considerations outlined in [RFC7761] and [RFC4607] SHOULD be considered where appropriate.

In section 6.1.1 of [RFC7761], only forged join/prune message should be considered as a potential attack vector, as PIM Light does not process Hello or Assert messages. In addition, as detailed in Section 6.3, the authentication mechanisms described in [RFC5796] can be applied to PIM Light via IPsec Encapsulating Security Payload (ESP) or, optionally, the Authentication Header (AH).

6. Acknowledgments

Would like to thank Sandy <Zhang Zheng> and Tanmoy Kundu for their suggestions and contribution to this document.

7. References

7.1. Normative References

[iana_pim-parameters_join-attribute-types]
"", January 2022, <https://www.iana.org/assignments/pim-parameters/pim-parameters.xhtml#pim-parameters-2>.

[iana_pim-parameters_message-types]
"", January 2022, <https://www.iana.org/assignments/pim-parameters/pim-parameters.xhtml#message-types>.

[RFC2119] "S. Brandner, "Key words for use in RFCs to Indicate Requirement Levels"", March 1997.

[RFC4607] "H. Holbrook, B. Cain "Source-Specific Multicast for IP"".

[RFC5015] "M. Handley, I. Kouvelas, T. Speakman, L. Vicisano "Bidirectional Protocol Independent Multicast"".

[RFC5384] "A. Boers, I. Wijnands, E. Rosen "PIM Join Attribute Format"", March 2016.

[RFC5796] "W. Atwood, S. Islam, M. Siami "Authentication and Confidentiality in PIM-SM"".

[RFC6559] "D. Farinacci, I. Wijnands, S. Venaas, M. Napierala "A reliable Transport Mechanism for PIM"".

[RFC7761] "B.Fenner, M.Handley, H. Holbrook, I. Kouvelas, R. Parekh, Z.Zhang "PIM Sparse Mode"", March 2016.

[RFC8174] "B. Leiba, "ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words"", May 2017.

[RFC8279] "Wijnands, IJ., Rosen, E., Dolganow, A., Przygienda, T. and S. Aldrin, "Multicast using Bit Index Explicit Replication"", October 2016.

[RFC9260] "R. Stewart, M. Tuxen, K. Nielsen, "Stream Control Transmission Protocol"", June 2022.

7.2. Informative References

[draft-ietf-bier-pim-signaling] "H.Bidgoli, F.XU, J. Kotalwar, I. Wijnands, M.Mishra, Z. Zhang, "PIM Signaling Through BIER Core"", July 2021.

[RFC3973] "A. Adams, J. Nicholas, W. Siadak, "Protocol Independent Multicast - Dense Mode"".

Authors' Addresses

Hooman Bidgoli (editor)
Nokia
March Road
Ottawa Ontario K2K 2T6
Canada
Email: hooman.bidgoli@nokia.com

Stig Venaas
Cisco System, Inc.
Tasman Drive
San Jose, California 95134
United States of America
Email: stig@cisco.com

Mankamana Mishra
Cisco System
Tasman Drive
San Jose, California 95134
United States of America
Email: mankamis@cisco.com

Zhaohui Zhang
Juniper Networks
Boston,
United States of America
Email: zzhang@juniper.com

Mike McBride
Futurewei Technologies Inc.
Santa Clara,
United States of America
Email: michael.mcbride@futurewei.com

IPv6 Operations (v6ops) Working Group
Internet Draft
Intended status: Informational
Expires: May 2025

X. Xiao
E. Vasilenko
Huawei Technologies
E. Metz
KPN
G. Mishra
Verizon Inc.
N. Buraglio
Energy Sciences Network

November 25, 2024

Neighbor Discovery Considerations in IPv6 Deployments
draft-ietf-v6ops-nd-considerations-07

Abstract

Neighbor Discovery (ND) is a critical part of IPv6. ND uses multicast extensively and trusts all hosts. In some scenarios, such as wireless networks, multicast can be inefficient. In other scenarios, such as public access networks, hosts may not be trustworthy. Consequently, ND can have issues in some scenarios. The issues and mitigation solutions are documented in more than 20 RFCs, making it challenging to track all these issues and solutions. Therefore, an overview document is helpful.

This document first summarizes the published ND issues and the solutions. This provides a one-stop reference. This document then analyzes these mitigation solutions to reveal that isolating hosts into different subnets or links can help prevent ND issues. Three isolation methods and their applicability are described. A simple guideline is provided for selecting a suitable isolation method to prevent potential ND issues.

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 in May 2025.

Table of Contents

1. Introduction

Neighbor Discovery [ND] is specified in RFC 4861. It defines how hosts and routers on the link interact with each other. ND contains eight main procedures:

1. Host's Duplicate Address Detection (DAD): hosts generate Link-Local Addresses (LLAs) and use multicast Neighbor Solicitations (NSs) for DAD.
2. Router Discovery: hosts send multicast Router Solicitations (RSs) to discover the routers. Routers respond with unicast Router Advertisements (RAs) with subnet prefixes for the link and other information. Routers also send unsolicited multicast RAs from time to time.
3. Host's Global Unicast Address (GUA) DAD: hosts form GUA and use multicast NSs for DAD.
4. Router's Neighbor Discovery: When a router is to forward a packet to an on-link host for the first time, the router uses multicast NSs to perform address resolution for the host.
5. Host's Neighbor Discovery: When a host is to send a packet to another on-link host, the source host uses multicast NSs to perform address resolution for the destination host.
6. Host/router's Node Unreachability Detection (NUD): hosts/routers use unicast NSs for NUD.

7. Host's link-layer address change announcement: hosts may use multicast NAs to announce link-layer address changes.
8. Router's Redirect: Routers send Redirect packets to inform a host of a better router or that the destination host is on-link.

ND can have issues in some scenarios due to the use of multicast, trusting all hosts, or installing Neighbor Cache Entry (NCE) on demand. Various ND issues and mitigation solutions have been published in more than 20 RFCs, including:

. ND Trust Models and Threats [RFC3756],
. Secure ND [SeND],
. Cryptographically Generated Addresses [CGA],
. ND Proxy [RFC4389],
. Optimistic ND [RFC4429],
. ND for mobile broadband [RFC6459][RFC7066],
. ND for fixed broadband [TR177],
. ND Mediation [RFC6575],
. Operational ND Problems [RFC6583],
. Wireless ND (WiND) [RFC6775][RFC8505][RFC8928][RFC8929][SND],
. DAD Proxy [RFC6957],
. Source Address Validation Improvement [SAVI],
. Router Advertisement Guard [RA-Guard][RA-Guard+],
. Enhanced Duplicate Address Detection [RFC7527],
. Scalable ARP [RFC7586],
. Reducing Router Advertisements [RFC7772],
. Unique Prefix Per Host [RFC8273],
. ND Optimization for TRILL [RFC8302],
. Gratuitous Neighbor Discovery [GRAND],
. Proxy ARP/ND for EVPN [RFC9161].

Because of the number of RFCs involved, it can become difficult to track issues and solutions. This document summarizes these RFCs into a one-stop reference to better inform network administrators about potential ND issues and mitigation solutions. This document also identifies three host isolation methods that are useful for preventing potential ND issues and provides a simple guideline for selecting one of them.

1.1. Terminology

Some important terms are defined in this section.

MAC - To avoid confusion with link-local addresses, link-layer addresses are referred to as MAC addresses in this document.

Host Isolation – separating hosts into different subnets or links.

Subnet & Link Isolation – assigning a unique prefix to each host, and connecting each host in a P2P link to the router. Every host is therefore in its subnet and its link. This is also called L3 & L2 Isolation.

Subnet Isolation – assigning a unique prefix per host so that each host is in its subnet. The hosts may be in the same link or different links. This is also called L3 Isolation.

Proxy Isolation – using a routing proxy device to represent the hosts behind it and separating hosts in a subnet into multiple multicast domains. This is also called Partial L2 Isolation.

2. Review of ND Issues

2.1. Multicast May Cause Performance and Reliability Issues

ND uses multicast for Node Solicitations (NSs), Node Advertisements (NAs), Router Solicitations (RSs) and Router Advertisements (RAs). Multicast can be inefficient in some scenarios, e.g. large L2 networks and wireless networks.

In large L2 networks, ND multicast can create a large amount of protocol traffic. This can consume network bandwidth, create a processing burden, and reduce network performance [RFC7342].

In Wi-Fi networks, mobile devices often employ power-saving modes, where they may sleep through multiple beacon intervals and wake up only for Delivery Traffic Indication Message (DTIM) beacons [RFC9119]. Since DTIM beacons occur less frequently, this can introduce delays in receiving multicast traffic. Moreover, multicast frames are transmitted at lower data rates without retransmissions. As a result, multicast traffic over Wi-Fi can suffer from reduced performance and reliability. DAD uses the lack of a response as an indication that the address is not currently in use. If the DAD multicast messages are lost, DAD will not work properly.

ND uses multicast in the following messages. Multicast impact on performance and reliability is summarized below:

. Hosts' LLA, GUA DAD: may cause performance issues in both wired and wireless networks, and possibly reliability issues in wireless networks.

- Router's periodic unsolicited RAs: multicast RAs are generally limited to one packet every 3s, and there are usually only one or two routers on the link, so it is unlikely to cause a performance issue. However, for battery-powered hosts, such messages may wake them up and create battery life issues [RFC7772]. Additionally, three RAs in a row may be lost in wireless which depreciates the router on the host.
- Router's address resolution for hosts: in an L2 network of N hosts, there can be N such multicast messages. This may cause performance issues when N is large.
- Hosts address resolution for hosts: in an L2 network of N hosts, there can be N-square such multicast messages. This may cause performance issues when N is large.
- Hosts' MAC address change NAs: this type of multicast message is rare and will not cause a performance or reliability issue. It will not be further discussed.

Multicast originated from hosts and routers will be called host multicast and router multicast hereafter.

2.2. Trusting-all-hosts May Cause On-link Security Issues

ND trusts all hosts. In some scenarios, such as public access networks, some hosts may not be trustworthy. An attacker on the link can cause the following security issues [RFC3756][RFC9099]:

- Source IP address spoofing: an attacker can use a victim host's IP address as the source address of its ND message to pretend to be the victim. The attacker can then launch Redirect or Denial of Service (DoS) attacks on the victim.
- DAD denial: an attacker can repeatedly reply to a victim's DAD messages, causing the victim's address configuration procedure to fail, resulting in a denial of service to the victim host.
- Forged RAs: an attacker can send RAs to other hosts to claim to be a router and preempt the real router, resulting in a Redirect attack [RA-Guard].
- Forged Redirects: an attacker can pretend to be the router and send Redirects to other hosts to redirect their traffic from the router to itself, resulting in a Redirect attack.
- Replay attacks: an attacker can capture valid ND messages and replay them later.

2.3. Router-NCE-on-Demand May Cause Forwarding Delay, NCE Exhaustion, and Address Accountability Issues

In ND, a router does not maintain (IP, MAC address) binding (i.e. Neighbor Cache Entry or NCE) for a host until it is needed. This is called Router-NCE-on-Demand in this document. When a router is to forward a packet to a host, it will perform address resolution to find the MAC address of the host. This can cause multiple issues:

- The packet has to be buffered before the router finds out the MAC address of the host. This delays forwarding, and depending on the router's buffer size, may also cause packet loss. This is called "Router-NCE-on-Demand Forwarding Delay" in this document.
- The way ND performs address resolution is the source node will create an NCE entry first and set its state to INCOMPLETE, then the node will multicast NSs to all the nodes and wait for the destination node to reply with its MAC address. This creates a security vulnerability. If an attacker sends a large number of packets destined to non-existing IP addresses, the router will create a large number of NCEs in INCOMPLETE state while trying to resolve the MAC addresses. The router may run out of resources and stop functioning. This is called "NCE Exhaustion" in this document. Note that in this case, the attacker can be off-link.
- With SLAAC, a host forms its own IP address. A router does not know the host's IP address until an NCE entry is installed. In a service provider network, subscribers are typically managed by their IP addresses. Consequently, if the router does not know a host's IP address, the service provider cannot manage the subscriber. In other words, there is a lack of address accountability. This is an issue for public access networks. In addition, after the NCE entries are installed on the router, there is no clearly defined method to retrieve them for management purpose [RFC9099, Section 2.6.1.4].

2.4. Summary of ND Issue

The ND issues discussed in Sections 2.1 to 2.3 are summarized below. It is worth noting that these issues originate from three causes: multicast, trusting all hosts, and Router-NCE-on-Demand. If a cause can be eliminated, the corresponding issues will also be eliminated. This points out the directions for preventing ND issues.

- Performance issues caused by multicast
  - I1: LLA DAD degrading performance

- o I2: Unsolicited RA draining hosts' battery
- o I3: GUA DAD degrading performance
- o I4: Router address resolution for hosts degrading performance
- o I5: Host Address resolution for other hosts degrading performance

. Reliability issues caused by multicast
- o I6: LLA DAD not reliable for wireless networks
- o I7: GUA DAD not reliable for wireless networks

. On-link security issues caused by trusting all hosts
- o I8: Source IP address spoofing
- o I9: DAD denial
- o I10: Forged RAs
- o I11: Forged Redirects
- o I12: Replay attacks

. Router-NCE-on-Demand related issues
- o I13: Router NCE exhaustion
- o I14: Router forwarding delay
- o I15: Lack of address accountability with SLAAC

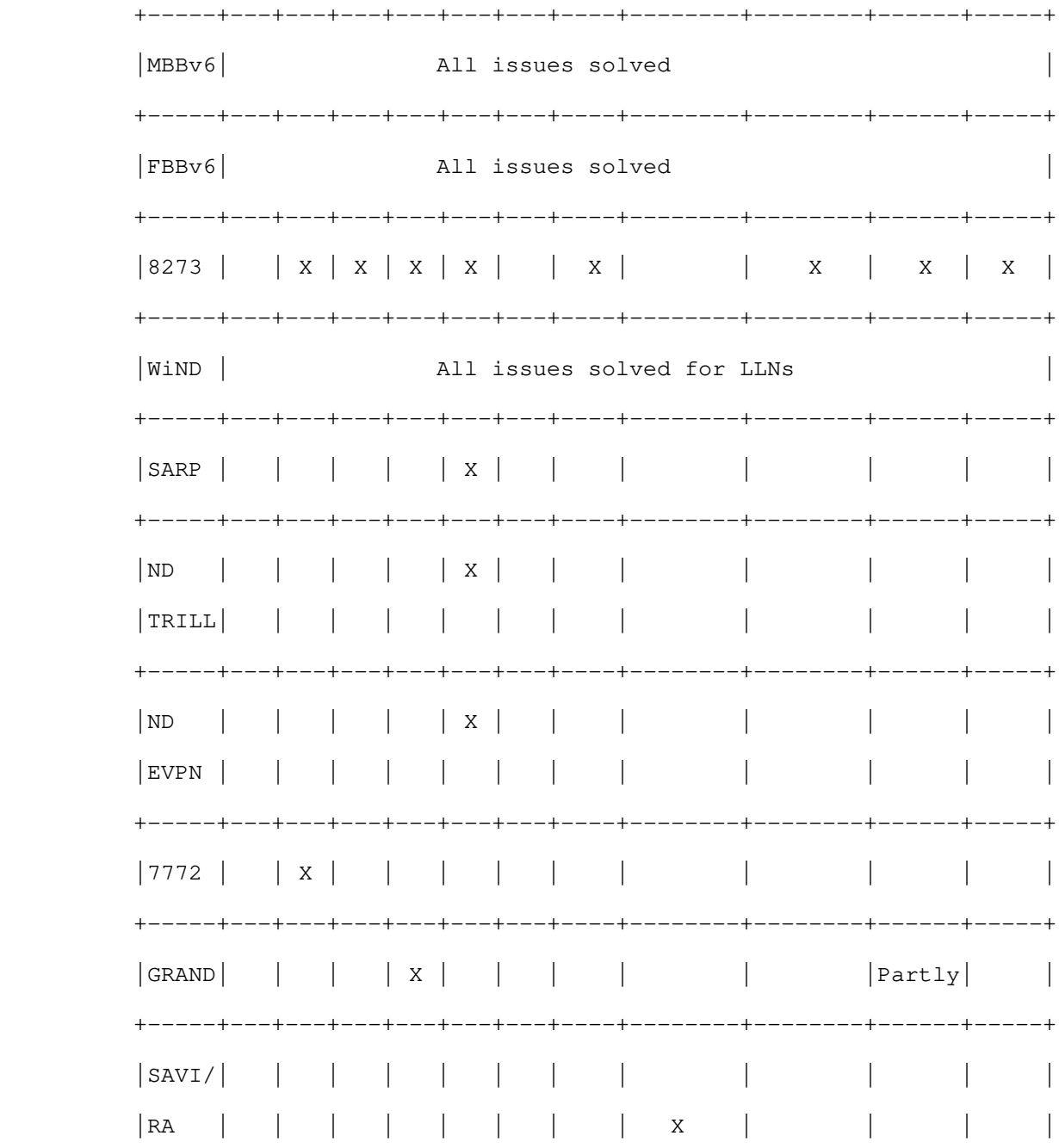
It is worth noting that these are just potential issues. Depending on the usage scenarios, they may not actually occur.

When the above issues can happen, it is advisable to be aware of the mitigation solutions available for them, as described in the next section.

3. Review of ND Mitigation Solutions

This section reviews the ND mitigation solutions developed over the years so that network administrators can get an idea of what solutions are available for which issues. They are summarized in Table 1 below for easy reference. The solutions are reviewed in an order that helps to reveal a methodology that can be useful for preventing ND issues, which will be discussed in Section 4.

	Multicast performance	Reli-ability	On-link security	R NCE Exhaust.	Fwd. Delay	NoAdr Acct.
Issue	1	2	3	4	5	6	7	8-12	13	14	15


MBBv6 All issues solved
FBBv6 All issues solved
8273 X X X X X X X
WiND All issues solved for LLNs
SARP X
ND TRILL X
ND EVPN X
7772 X
GRAND X Partly
SAVI/RA X

G/G+											
6583								X			
AddrR										X	

Table 1. Solutions for identified issues

3.1. ND Solution in Mobile Broadband IPv6

Mobile Broadband IPv6 (MBBv6) is defined in "IPv6 in 3GPP EPS" [RFC6459], "IPv6 for 3GPP Cellular Hosts" [RFC7066], and "Extending an IPv6 /64 Prefix from a Third Generation Partnership Project (3GPP) Mobile Interface to a LAN Link" [RFC7278]. The solution key points are:

- Putting every host, i.e. the mobile User Equipment (UE), in a P2P link with the router, i.e. the mobile gateway. MBBv6 also simplifies ND to take advantage of this P2P architecture. As a result:
  - All multicast is effectively turned into unicast.
  - The P2P links in MBB do not have MAC address. Therefore, Router-NCE-on-Demand is not needed.
  - Trusting-all-host is only relevant to the router. By applying some filtering at the router, e.g. dropping RAs from the host, even malicious hosts cannot cause security harm.
- Assigning a unique /64 prefix to each host. Together with the P2P link, this puts each host in a separate link and subnet.
- Maintaining (prefix, interface) binding at the router for forwarding purpose.

Since all the three causes of ND issues are addressed, all the ND issues discussed in Section 2.4 are also addressed.

3.2. ND Solution in Fixed Broadband IPv6

FBBv6 is defined in "IPv6 in the context of TR-101" [TR177]. FBBv6 has two flavors:

- P2P: every host, i.e. the Residential Gateway (RG), is in a P2P link with the router, i.e. the Broadband Network Gateway (BNG). In this case, the solution is essentially the same as MBBv6. All ND issues discussed in Section 2.4 are solved.
- P2MP: all hosts on an access device are in a P2MP link with the router. This is implemented by aggregating all hosts into a single VLAN at the router and implementing Split Horizon at the access device to prevent direct host communication.

The key points of FBBv6-P2MP [TR177] are:

- Implementing DAD Proxy [RFC6957]: P2MP architecture with Split Horizon breaks normal ND's DAD procedure. Because all hosts are in the same interface from the router's perspective, the router must ensure that the hosts have different LLAs and GUAs. Otherwise, the router will not be able to distinguish them. However, because hosts cannot reach each other, normal DAD will not function as expected. Therefore, the router must participate in the hosts' DAD process and help hosts resolve duplication. With P2MP link and DAD Proxy:
  - All host multicast to the router is effectively turned into unicast, as every host can only reach the router.
  - Trusting-all-host is only relevant to the router. By applying some simple filtering at the router, e.g. dropping RAs from the host, even malicious hosts cannot cause security harm.
- Results of assigning a unique /64 prefix to each host:
  - The router can proactively create (IP prefix, MAC address) binding and use it for forwarding. There is no Router-NCE-on-Demand.
  - Since different hosts are in different subnets, hosts will send traffic to other hosts via the router. There is no address resolution for other hosts.
  - Without address resolution, router multicast to hosts consists only of unsolicited RAs. Because every host is in its subnet, unsolicited RAs will be sent individually to each host with the "host's MAC address replacing the multicast MAC address" approach specified in [RFC6085]. Therefore, router multicast is turned into unicast.

Since all three causes of ND issues are addressed, all the ND issues discussed in Section 2.4 are also addressed.

3.3. Unique IPv6 Prefix per Host

Unique IPv6 Prefix per Host is specified in [RFC8273]. The purpose is to "improve host isolation and enhanced subscriber management on shared network segments" such as Wi-Fi or Ethernet. The solution key points are:

- Assigning a unique prefix to each host with SLAAC. As a result:
  - When a prefix is assigned to the host, the router can proactively create (Prefix, MAC address) binding and use it for forwarding. There is no more Router-NCE-on-Demand.
  - Since different hosts are in different subnets, hosts will send traffic to other hosts via the router. There is no host-to-host address resolution.
  - Without address resolution, downstream multicast to hosts consists only of unsolicited RAs. They will be sent to hosts one by one in unicast because the prefix for every host is different.

Therefore, ND issues caused by NCE-on-Demand and router multicast are avoided.

RFC 8273 believes that "A network implementing a unique IPv6 prefix per host can simply ensure that devices cannot send packets to each other except through the first-hop router". But this may not be true when hosts are on a shared medium like Ethernet. In this case, hosts may still reach each other in L2 with their LLAs via multicast. So, issues caused by host multicast and Trusting-all-hosts may happen.

3.4. Wireless ND and Subnet ND

Wireless ND (WiND) is specified in a series of RFCs [RFC6775][RFC8505][RFC8928][RFC8929]. WiND defines a fundamentally different ND solution for Low-Power and Lossy Networks (LLNs) [RFC7102]. WiND changes host and router behaviors to use multicast only for router discovery. The solution key points are:

- Hosts use unicast to proactively register their addresses at the routers. Routers use unicast to communicate with hosts and become an abstract registrar and arbitrator for address ownership.
- The router also proactively installs Neighbor Cache Entries (NCEs) for the hosts. This avoids the need for address resolution for the hosts.
- The router sets PIO L-bit to 0. Each host communicates only with the router.

. Other functionalities that are relevant only to LLNs.

WiND addresses all ND issues discussed in Section 2.4 in LLNs. If it is used outside LLNs, it avoids ND issues caused by NCE-on-Demand and router multicast.

Subnet Neighbor Discovery [SND] generalizes the solutions defined in WiND and defines a new protocol named Subnet Gateway Protocol (SGP). It is being discussed in the IPv6 Maintenance (6man) WG.

3.5. Scalable Address Resolution Protocol

Scalable Address Resolution Protocol (SARP) is an Experimental solution specified in [RFC7586]. The usage scenario is DCs where large L2 domains span across multiple sites. In each site, multiple hosts are connected to a switch. The hosts can be VMs so the number can be large. The switches are interconnected by a native or overlay L2 network.

The switch will snoop and install (IP, MAC address) proxy table for the local hosts. The switch will also reply to address resolution requests from other sites to its hosts with its own MAC address. This way, all hosts in a site will appear to have a single MAC address to other sites. Therefore, a switch only needs to build a MAC address table for the local hosts and the remote switches, not for all the hosts in the L2 domain. The MAC address table size of the switches is therefore significantly reduced. A switch will also add the (IP, MAC address) replies from remote switches to its proxy ND table so that it can reply to future address resolution requests for such IPs directly. This greatly reduces the number of address resolution multicast in the network.

Unlike MBBv6, FBBv6, and RFC 8372 which try to address all ND issues discussed in Section 2.4, SARP focuses on reducing address resolution multicast to improve the performance and scalability of large L2 domains in DCs.

3.6. ARP and ND Optimization for Transparent Interconnection of Lots of Links (TRILL):

ARP and ND Optimization for TRILL is specified in [RFC8302]. The solution is very similar to SARP discussed in Section 3.5. It can be considered as an application of SARP in the TRILL environment.

Like SARP, ARP, and ND Optimization for TRILL focuses on reducing multicast address resolution.

3.7. Proxy ARP/ND in EVPN

Proxy ARP/ND in EVPN is specified in [RFC9161]. The usage scenario is Data Centers (DCs) where large L2 domains span across multiple sites. In each site, multiple hosts are connected to a Provider Edge (PE) router acting as a switch. The PEs are interconnected by an overlay network.

PE of each site snoops the local address resolution NAs to build (IP, MAC address) Proxy ND table entries. PEs then propagate such Proxy ND entries to other PEs via BGP EVPN. Each PE also snoops address resolution NSs from its hosts. If an entry exists in its Proxy ND table for the specified destination IP address, the PE will reply directly. Consequently, the number of multicast address resolution messages is significantly reduced.

Like SARP, Proxy ARP/ND in EVPN also focuses on reducing address resolution multicast.

3.8. Gratuitous Neighbor Discovery

Gratuitous Neighbor Discovery is specified in [GRAND]. GRAND changes ND in the following ways:

- A node sends unsolicited NAs upon assigning a new IPv6 address to its interface.
- A router creates a new NCE for the host and sets its state to STALE.

Later, when the router receives traffic to the host, the existence of the NCE entry in the STALE state will cause the router to send unicast NS to the host to verify its reachability rather than sending multicast NS to resolve its MAC address. This can shorten the time the host's NCE entry reaches the REACHABLE state and improve forwarding performance. Therefore, GRAND provides an improvement but does not fully solve the Router-NCE-on-Demand issues. For example, NCE exhaustion can still happen.

3.9. Reducing Router Advertisements

[RFC7772] specifies a solution for reducing RAs:

- The router should respond to RS with unicast RA if the host's source IP address is specified (i.e. the RS is not the first RS before GUA DAD) and the host's MAC address is valid.
- The router should reduce multicast RA frequency.

- Sleeping hosts that process unicast packets while asleep must also process multicast RAs while asleep.
- Sleeping hosts that do not intend to maintain IPv6 connectivity while asleep should either disconnect from the network and clear all IPv6 configuration or perform Detecting Network Attachment in IPv6 (DNAv6) procedures [RFC6059] when waking up.

By reducing RAs, RFC 7772 reduces the energy consumption of battery-powered hosts that can be awakened by RAs.

3.10. Source Address Validation Improvement and Router Advertisement Guard

Source Address Validation Improvement [SAVI] binds an address to a port on an L2 switch and rejects claims from other ports for that address. Therefore, a node cannot spoof the IP address of another node.

[RA-Guard] and [RA-Guard+] only allow RAs from a port that a router is connected to. Therefore, nodes on other ports cannot pretend to be a router.

[SAVI], [RA-Guard], and [RA-Guard+] address the on-link security issues.

3.11. RFC 6583 Dealing with Operational Neighbor Discovery Problems

Router NCE Exhaustion handling is described in [RFC6583]. This is to deal with the off-link attack issue discussed in Section 2.3. The solution key points are:

- For operators:
  - Filtering of unused address space so that messages to such addresses can be dropped rather than triggering NCE creation;
  - Implement rate-limiting mechanisms for NDP (Neighbor Discovery Protocol) message processing to prevent CPU and memory resources from being overwhelmed.
- For vendors:
  - Prioritizing NDP processing for existing NCEs over creating new NCEs

RFC 6583 acknowledges that "some of these options are 'kludges', and can be operationally difficult to manage". RFC 6583 partially addresses the Router NCE Exhaustion issue. In the real world, network equipment vendors simply limit the number of NCE entries on

a router interface to prevent Router NCE Exhaustion. But this can have a side-effect. When a host uses more IPv6 addresses than the limit, irregular packet drops may result because the router does not maintain NCEs for all those IPv6 addresses [DHCP-PD].

3.12. Registering Self-generated IPv6 Addresses using DHCPv6

This document defines a method for informing a DHCPv6 server that a device has one or more self-generated or statically configured addresses [AddrReg]. This enables network administrators to retrieve the IPv6 addresses for each host from the DHCPv6 server. With IPv4, network administrators can retrieve a host's IP address from the DHCP server. With IPv6 and SLAAC, this is not possible, as discussed in Section 2.3. [AddrReg] makes this possible.

3.13. Enhanced DAD

Enhanced DAD is specified in [RFC7527]. Enhanced DAD addresses a DAD failure issue in a specific situation: looped back interface. DAD will fail in a looped-back interface because the sending host will receive the DAD message back and will interpret it as another host is trying to use the same address. The solution is to include a Nonce option (defined in [SeND]) in each DAD message so that the sending host can detect that the looped-back DAD message is sent by itself.

Enhanced DAD does not solve any ND issue discussed in Section 2. It extends ND to work in a new scenario: looped-back interface. It is reviewed here only for completeness.

3.14. ND Mediation for IP Interworking of Layer 2 VPNs

ND mediation is specified in [RFC6575]. When two Attachment Circuits (ACs) are interconnected by a Virtual Private Wired Service (VPWS), and the two ACs are of different media (e.g. one is Ethernet while the other is Frame Relay), the two Provider Edges (PEs) must interwork to provide mediation service so that a Customer Edge (CE) can resolve the MAC address of the remote end. RFC 6575 specifies such a solution.

ND Mediation does not address any ND issue discussed in Section 2. It extends ND to work in a new scenario: two ACs of different media interconnected by a VPWS. It is reviewed here only for completeness.

3.15. ND Solutions Defined before the Latest Versions of ND

The latest versions of [ND] and [SLAAC] are specified in RFCs 4861 and 4862. Several ND mitigation solutions are based on the older version of ND and SLAAC. They are reviewed in this section only for completeness.

3.15.1. SeND

Secure Neighbor Discovery [SeND] is specified in RFC 3971. The purpose is to ensure that hosts and routers are trustworthy. SeND defined three new ND options (i.e. Cryptographically Generated Addresses [CGA], RSA public-key cryptosystem, Timestamp/Nonce), an authorization delegation discovery process, an address ownership proof mechanism, and requirements for the use of these components in NDP.

SeND addresses the Trusting-all-hosts issues. But it has high requirements on the hosts and routers, especially to maintain the keys. It has very low market adoption.

3.15.2. Cryptographically Generated Addresses (CGA)

Cryptographically Generated Addresses [CGA] is specified in RFC 3972. The purpose is to associate a cryptographic public key with an IPv6 address in [SeND]. The solution key point is to generate the Interface Identifier (IID) of the IPv6 address by computing a cryptographic hash of the public key. The resulting IPv6 address is called a CGA. The corresponding private key can then be used to sign messages sent from the address.

CGA uses the fact that a legitimate host does not care about the bit combination of IID that would be created by some hash procedure. The attacker needs an exact IID to impersonate the legitimate hosts but then the attacker is challenged to do a reverse hash calculation that is a strong mathematical challenge.

CGA is part of SeND. It has low market adoption.

3.15.3. ND Proxy

ND Proxy is specified in [RFC4389]. It is an Experimental solution. The purpose is to enable multiple links joined by an ND-Proxy device to work as a single link. The ND-Proxy acts like a bridge:

. When it receives an ND request from a host in a link, it will "proxy" the message out the "best" outgoing interface. If there is no "best" interface, the ND-Proxy will "proxy" the message to all other links. Here "proxy" means acting as if the ND message originates from the ND-Proxy itself. That is, the ND-Proxy will change the ND message's source IP and source MAC address to the ND-Proxy's outgoing interface's IP and MAC address, and create an NCE entry at the outgoing interface accordingly.
. When ND-Proxy receives an ND reply, it will act as if the ND message is destined to itself, and update the NCE entry state at the receiving interface. Based on such state information, the ND-Proxy can determine the "best" outgoing interface for future ND requests. The ND-Proxy then "proxy" the ND message back to the requesting host.

ND Proxy does not solve any ND issue discussed in Section 2. It extends ND to work in a new scenario: multiple links joined by a device that is not a bridge but acting like a bridge.

The idea of ND Proxy is widely used in SARP, ND Optimization for TRILL, and Proxy ARP/ND in EVPN which are discussed in Sections 3.5 to 3.7.

3.15.4. Optimistic DAD

Optimistic DAD is specified in [RFC4429]. The purpose is to minimize address configuration delays in the successful case and to reduce disruption as far as possible in the failure case. That is, Optimistic DAD lets hosts immediately use the newly formed address to communicate before DAD actually completes, assuming that DAD will succeed anyway. If the address turns out to be duplicate, Optimistic DAD provides a set of mechanisms to minimize the impact. Optimistic DAD modified the original ND (RFC 2461) and SLAAC (RFC 2462) but the solution was not incorporated into the latest specification of [ND] and [SLAAC].

Optimistic DAD does not solve any ND issue discussed in Section 2. It is reviewed here only for completeness.

4. Guidelines for Prevention of Potential ND Issues

By knowing the potential ND issues and associated mitigation solutions, network administrators of existing IPv6 deployments can assess whether these issues may occur in their networks and, if so, whether to deploy the mitigation solutions proactively. Deploying

these solutions may take time and additional resources; therefore, it is advisable to plan ahead.

Network administrators who plan to start their IPv6 deployments can use the issue-solution information to help plan their deployments. Moreover, they can take proactive action to prevent potential ND issues.

4.1. Learning Host Isolation from the Existing Solutions

Although the various ND solutions look unrelated, dividing them into four groups helps to reveal an important point: isolating hosts can help to prevent ND issues.

The first group contains MBBv6 and FBBv6. These solutions isolate hosts in both L3 and L2 by putting each host in its subnet and its link. This isolation method is called "L3 & L2 Isolation" or "Subnet & Link Isolation". It prevents ND issues caused by multicast and Trusting-all-hosts as every host is in its subnet and link. Because a router can route packets to a host based on its unique prefix, there is no need for Router-NCE-on-Demand, therefore ND issues caused by Router-NCE-on-Demand are also prevented.

The second group contains Unique Prefix Per Host (UPPH) [RFC8273]. UPPH also isolates hosts into different subnets but may leave all hosts in the same shared medium. This isolation method is called "L3 Isolation" or "Subnet Isolation". As discussed in Section 3.3, this isolation method prevents ND issues caused by router multicast and Router-NCE-on-Demand.

The third group contains WiND, SARP, ND Optimization for TRILL, and Proxy ND in EVPN. They use a proxy device to represent the hosts behind it and effectively isolate such hosts into different multicast domains from other hosts. Multiple hosts are still in a multicast domain and all hosts are still in the same subnet. Therefore, this isolation method is called Partial L2 Isolation" or "Proxy Isolation". This isolation method alleviates ND issues from host multicast for address resolution.

The fourth group contains the remaining solutions. They do not isolate hosts. They do not prevent any ND issues but focus on solving a specific ND issue.

The above reveals that the stronger hosts are isolated, the more ND issues can be prevented. This is natural because isolating hosts

reduces multicast scope, the number of hosts to trust, and possibly the need for Router-NCE-on-Demand, the three causes of ND issues.

This understanding can be used to prevent ND issues.

4.2. Applicability of Various Isolation Methods and a Simple Guideline

4.2.1. Applicability of L3 & L2 Isolation

The benefits are:

- All ND issues discussed in Section 2.4 can be prevented.

The constraints or entry requirements are:

- The hosts must be able to set up P2P links with the router.
- Many prefixes will be needed, one per host.
    - This is unlikely to be an issue for IPv6. Today, any member of a Regional Internet Registry (RIR) can get a /29 [RIPE738]. This contains 32 billion /64 prefixes and should be sufficient for any scenario. MBBv6 assigning /64 prefixes to billions of mobile UEs [RFC6459] and FBBv6 assigning /56 prefixes to hundreds of millions of routed RGs [TR177] are evidence that this is doable.
- Each host is easily identifiable by its unique prefix. This theoretically reduces privacy. However, hosts can be identified by many other methods, e.g. by using cookies. Therefore, the real impact on privacy may be limited.
- The router must support a "Subnet Isolation with P2P Link" solution, e.g. MBBv6, as described in Section 3.1.
- Many interfaces will be needed at the router, one per host.
- All hosts will communicate through the router, and the router may become a bottleneck.
- Services relying on multicast communication among hosts, e.g. mDNS, will not work.

4.2.2. Applicability of L3 Isolation

The benefits are:

- All ND issues discussed in Section 2.4 are prevented except "LLA DAD multicast degrading performance", "LLA DAD not reliable for wireless networks", and "On-link security" issues. Depending on the shared medium, these remaining issues may not happen. For example, if the shared medium is Ethernet, "LLA DAD multicast degrading performance" and "LLA DAD not reliable for wireless networks" are non-issues. If the hosts can be trusted, e.g. in a private network, "On-link security" is also a non-issue.

- There is no new requirement on the hosts. Therefore, this method can be applied in many scenarios. It is practically the most usable host isolation method.

The constraints are:

- Many prefixes will be needed, one per host. However as explained above, this may not be an issue for organizations that can obtain sufficient IPv6 addresses from RIRs.

- The router must support a Subnet Isolation solution, e.g. [RFC8273] or [DHCP-PD].

- All host-to-host communication with GUA will go through the router, and the router may become a bottleneck.

- Each host is identifiable by its unique prefix. This might be a privacy issue as discussed previously.

4.2.3. Applicability of Partial L2 Isolation

The benefit is:

- Reduced multicast especially for address resolution, as the subnet is divided into multiple multicast domains.

The constraint is:

- The router must support Proxy Isolation.

4.2.4. A Simple Guideline

Given the applicability analysis above, network administrators can decide whether to apply any isolation method.

A simple guideline is to consider the isolation methods one by one in the order listed in the previous sections, that is, from the

strongest isolation to the weakest. With stronger isolation, more ND issues can be prevented but the entry requirements will also be higher. All things considered, L3 Isolation can be a good tradeoff because the benefits are clear while the entry requirements are manageable.

It is worth noting that, if a network administrator picks an isolation method that is too strong or too weak, there is no serious consequence. Picking an isolation method that is too strong means that the network administrator needs to meet more entry requirements upfront, while picking an isolation method that is too weak means that the network administrator may need to deploy more ND mitigation solutions to deal with ND issues. Either way, the resulting solution can still work.

5. Security Considerations

This document is a review of known ND issues and solutions. It does not introduce any new solutions. Therefore, it does not introduce new security issues.

6. IANA Considerations

This document has no request to IANA.

7. References

7.1. Informative References

[AddrReg] W. Kumari, S. Krishnan, R. Asati, L. Colitti, J. Linkova, S. Jiang, "Registering Self-generated IPv6 Addresses using DHCPv6", draft-ietf-dhc-addr-notification-13.

[CGA] T. Aura, "Cryptographically Generated Addresses (CGA)", RFC3972

[DHCP-PD] L. Colitti, J. Linkova, X. Ma, "Using DHCP-PD to Allocate Unique IPv6 Prefix per Host in Broadcast Networks", draft-ietf-v6ops-dhcp-pd-per-device-08.

[GRAND] J. Linkova, "Gratuitous Neighbor Discovery: Creating Neighbor Cache Entries on First-Hop Routers", RFC 9131

[mDNS] S. Cheshire, M. Krochmal, "Multicast DNS", RFC 6762.

[ND] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, RFC 4861.

[RA-Guard] E. Levy-Abegnoli, G. Van de Velde, C. Popoviciu, J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, DOI 10.17487/RFC6105, February 2011, RFC 6105.

[RA-Guard+]F. Gont, "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)", RFC 7113, DOI 10.17487/RFC7113, February 2014, RFC 7113.

[RFC3756] P. Nikander, J. Kempf, E. Nordmark, "IPv6 Neighbor Discovery (ND) Trust Models and Threats", RFC 3756.

[RFC4291] R. Hinden, S.Deering, "IP Version 6 Addressing Architecture", RFC 4291.

[RFC4389] D. Thaler, M. Talwar, C. Patel, "Neighbor Discovery Proxies (ND Proxy)", RFC 4389.

[RFC4429] N. Moore, "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429.

[RFC4903] D. Thaler, "Multi-Link Subnet Issues", RFC 4903.

[RFC6459] J. Korhonen, J. Soininen, B. Patil, T. Savolainen, G. Bajko, K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459.

[RFC6059] S. Krishnan, G. Daley, "Simple Procedures for Detecting Network Attachment in IPv6", RFC 6059.

[RFC6085] S. Gundavelli, M. Townsley, O. Troan, W. Dec, "Address Mapping of IPv6 Multicast Packets on Ethernet", RFC 6085.

[RFC6575] H. Shah, E. Rosen, G. Heron, V. Kompella, "Address Resolution Protocol (ARP) Mediation for IP Interworking of Layer 2 VPNs", RFC 6575.

[RFC6583] I. Gashinsky, J. Jaeggli, W. Kumari, "Operational Neighbor Discovery Problems", RFC 6583.

[RFC6775] Z. Shelby, S. Chakrabarti, E. Nordmark, C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775.

[RFC6957] F. Costa, J-M. Combes, X. Pougnard, H. Li, "Duplicate Address Detection Proxy", RFC 6957

[RFC7066] J. Korhonen, J. Arkko, T. Savolainen, S. Krishnan, "IPv6 for Third Generation Partnership Project (3GPP) Cellular Hosts", RFC 7066.

[RFC7102] JP. Vasseur, "Terms Used in Routing for Low-Power and Lossy Networks", RFC 7102.

[RFC7278] Extending an IPv6 /64 Prefix from a Third Generation Partnership Project (3GPP) Mobile Interface to a LAN Link", RFC7278.

[RFC7342] L. Dunbar, W. Kumari, I. Gashinsky, "Practices for Scaling ARP and Neighbor Discovery (ND) in Large Data Centers", RFC 7342.

[RFC7527] R. Asati, H. Singh, W. Beebee, C. Pignataro, E. Dart, W. George, "Enhanced Duplicate Address Detection", RFC 7527.

[RFC7586] Y. Nachum, L. Dunbar, I. Yerushalmi, T. Mizrahi, "The Scalable Address Resolution Protocol (SARP) for Large Data Centers", RFC7586.

[RFC7772] A. Yourtchenko, L. Colitti, "Reducing Energy Consumption of Router Advertisements", RFC 7772.

[RFC8273] J. Brzozowski, G. Van de Velde, "Unique IPv6 Prefix per Host", RFC 8273.

[RFC8302] Y. Li, D. Eastlake 3rd, L. Dunbar, R. Perlman, M. Umair, "Transparent Interconnection of Lots of Links (TRILL): ARP and Neighbor Discovery (ND) Optimization", RFC 8302.

[RFC8505] P. Thubert, E. Nordmark, S. Chakrabarti, C. Perkins, "Registration Extensions for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Neighbor Discovery", RFC 8505.

[RFC8928] P. Thubert, B. Sarikaya, M. Sethi, R. Struik, "Address-Protected Neighbor Discovery for Low-Power and Lossy Networks", RFC 8928.

[RFC8929] P. Thubert, C.E. Perkins, E. Levy-Abegnoli, "IPv6 Backbone Router", RFC 8929.

[RFC9099] E. Vyncke, K. Chittimaneni, M. Kaeo, E. Rey, "Operational Security Considerations for IPv6 Networks", RFC 9099.

[RFC9119] C. Perkins, M. McBride, D. Stanley, W. Kumari, JC. Zuniga, "Multicast Considerations over IEEE 802 Wireless Media", RFC 9119.

[RFC9161] J. Rabadan, S. Sathappan, K. Nagaraj, G. Hankins, T. King, "Operational Aspects of Proxy ARP/ND in Ethernet Virtual Private Networks", RFC 9161.

[RIPE738] IPv6 Address Allocation and Assignment Policy, https://www.ripe.net/publications/docs/ripe-738

[SAVI] J. Wu, J. Bi, M. Bagnulo, F. Baker, C. Vogt, "Source Address Validation Improvement (SAVI) Framework", RFC 7039.

[SeND] J. Arkko, J. Kempf, B. Zill, P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC3971.

[SLAAC] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862.

[SND] P. Thubert, M. Richardson, "Architecture and Framework for IPv6 over Non-Broadcast Access", Internet draft, June 2023.

[TR177] S. Ooghe, B. Varga, W. Dec, D. Allan, "IPv6 in the context of TR-101", Broadband Forum, TR-177.

8. Acknowledgments

The authors would like to thank Lorenzo Colitti, Warren Kumari, Pascal Thubert, Jen Linkova, Brian Carpenter, Eric Vyncke, Mike Ackermann, Nalini Elkins, Ed Horley, Ole Troan, David Thaler, Chongfeng Xie, Chris Cummings, Dale Carder, Tim Chown, Priyanka Sinha, Aijun Wang, Ines Robles, Magnus Westerlund, Barry Leiba for their reviews and comments. The authors would also like to thank Tim Winters for being the document shepherd.

Authors' Addresses

XiPeng Xiao
Huawei Technologies Dusseldorf
Hansaallee 205, 40549 Dusseldorf, Germany

Email: xipengxiao@huawei.com

Eduard Vasilenko
Huawei Technologies
17/4 Krylatskaya st, Moscow, Russia 121614

Email: vasilenko.eduard@huawei.com

Eduard Metz
KPN N.V.
Maanplein 55, 2516CK The Hague, The Netherlands

Email: eduard.metz@kpn.com

Gyan Mishra
Verizon Inc.

Email: gyan.s.mishra@verizon.com

Nick Buraglio
Energy Sciences Network

Email: buraglio@es.net

Network Working Group
Internet-Draft
Intended Status: Informational
Updates: 7932
Expires: May 26, 2025

J. Alakuijala
T. Duong
E. Kliuchnikov
Z. Szabadka
L. Vandevenne
Google, Inc

Nov 26, 2024

Shared Brotli Compressed Data Format
draft-vandevenne-shared-brotli-format-13

Abstract

This specification defines a data format for shared brotli compression, which adds support for shared dictionaries, large window and a container format to brotli (RFC 7932). Shared dictionaries and large window support allow significant compression gains compared to regular brotli. This document updates RFC 7932.

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 http://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 May 26, 2025.

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this specification is to extend the brotli compressed data format format ([RFC7932]) with new abilities that allow further compression gains:

* Shared dictionaries allow a static shared context between encoder and decoder for significant compression gains.

* Large window brotli allows much larger back reference distances to give compression gains for files over 16MiB.

* The framing format is a container format that allows storage of multiple resources and that reference dictionaries.

This document is the authoritative specification of shared brotli data formats and the backwards compatible changes to brotli, and defines:

* The data format of serialized shared dictionaries

* The data format of the framing format

* The encoding of window bits and distances for large window brotli in the brotli data format

* The encoding of shared dictionary references in the brotli data format

1.2. Intended audience

This specification is intended for use by software implementers to compress data into and/or decompress data from the shared brotli dictionary format.

The text of the specification assumes a basic background in programming at the level of bits and other primitive data representations. Familiarity with the technique of LZ77 coding [LZ77] is helpful but not required.

1.3. Scope

This specification defines a data format for shared brotli compression, which adds support for dictionaries and extended features to brotli [RFC7932].

1.4. Compliance

Unless otherwise indicated below, a compliant decompressor must be able to accept and decompress any data set that conforms to all the specifications presented here. A compliant compressor must produce data sets that conform to all the specifications presented here.

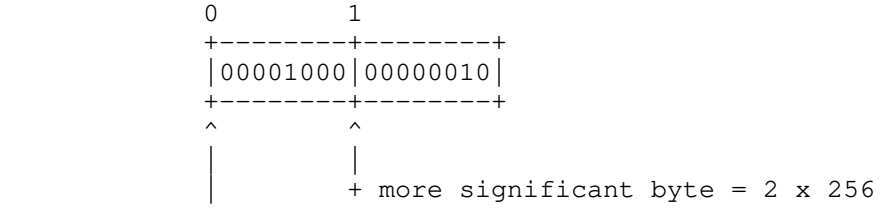
1.5. Definitions of terms and conventions used

Byte: 8 bits stored or transmitted as a unit (same as an octet). For this specification, a byte is exactly 8 bits, even on machines that store a character on a number of bits different from eight. See below for the numbering of bits within a byte.

String: a sequence of arbitrary bytes.

Bytes stored within a computer do not have a "bit order", since they are always treated as a unit. However, a byte considered as an integer between 0 and 255 does have a most- and least-significant bit, and since we write numbers with the most-significant digit on the left, we also write bytes with the most-significant bit on the left. In the diagrams below, we number the bits of a byte so that bit 0 is the least-significant bit, i.e., the bits are numbered:

Within a computer, a number may occupy multiple bytes. All multi-byte numbers in the format described here are unsigned and stored with the least-significant byte first (at the lower memory address). For example, the decimal 16-bit number 520 is stored as:



1.5.1. Packing into bytes

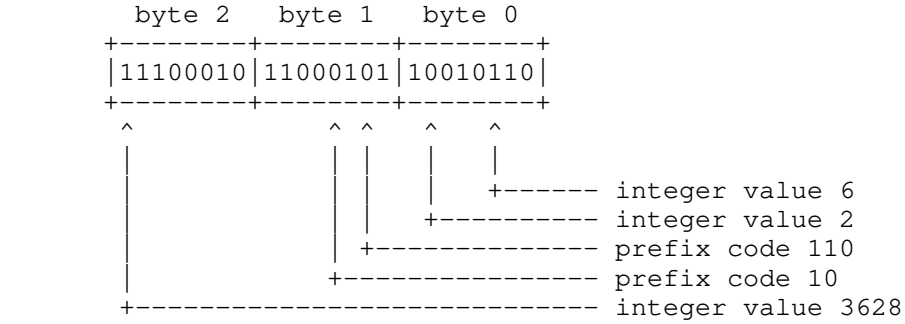
This document does not address the issue of the order in which bits of a byte are transmitted on a bit-sequential medium, since the final data format described here is byte- rather than bit-oriented. However, we describe the compressed block format below as a sequence

of data elements of various bit lengths, not a sequence of bytes. We must therefore specify how to pack these data elements into bytes to form the final compressed byte sequence:

* Data elements are packed into bytes in order of increasing bit number within the byte, i.e., starting with the least-significant bit of the byte.
* Data elements other than prefix codes are packed starting with the least-significant bit of the data element. These are referred to here as integer values and are considered unsigned.
* Prefix codes are packed starting with the most-significant bit of the code.

In other words, if one were to print out the compressed data as a sequence of bytes, starting with the first byte at the *right* margin and proceeding to the *left*, with the most-significant bit of each byte on the left as usual, one would be able to parse the result from right to left, with fixed-width elements in the correct MSB-to-LSB order and prefix codes in bit-reversed order (i.e., with the first bit of the code in the relative LSB position).

As an example, consider packing the following data elements into a sequence of 3 bytes: 3-bit integer value 6, 4-bit integer value 2, prefix code 110, prefix code 10, 12-bit integer value 3628.



2. Shared Brotli Overview

Shared brotli extends brotli [RFC7932] with support for shared dictionaries, larger LZ77 window and a framing format.

3. Shared Dictionaries

A shared dictionary is a piece of data shared by a compressor and decompressor. The compressor can take advantage of the dictionary

context to encode the input in a more compact manner. The compressor and the decompressor must use exactly the same dictionary. A shared dictionary is specially useful to compress short input sequences.

A shared brotli dictionary can use two methods of sharing context:

* An LZ77 dictionary. The encoder and decoder could refer to a given sequence of bytes. Multiple LZ77 dictionaries can be set.

* A custom static dictionary: a word list with transforms. The encoder and decoder will replace the static dictionary data with the data in the shared dictionary. The original static dictionary is described in Section 8 in [RFC7932]. The original data from Appendix A and Appendix B of [RFC7932] will be replaced. In addition, it is possible to dynamically switch this dictionary based on the data compression context, and/or to include a reference to the original dictionary in the custom dictionary.

If no shared dictionary is set the decoder behaves the same as in [RFC7932] on a brotli stream.

If a shared dictionary is set, then it can set any of: LZ77 dictionaries, overriding static dictionary words, and/or overriding transforms.

3.1. Custom Static Dictionaries

If a custom word list is set, then the following behavior of the RFC 7932 decoder [RFC7932] is overridden:

> Instead of the Static Dictionary Data from Appendix A of [RFC7932], one or more word lists from the custom static dictionary data are used.
>
> Instead of NDBITS at the end of Appendix A, a custom SIZE_BITS_BY_LENGTH per custom word list is used.
>
> The copy length for a static dictionary reference must be between 4 and 31 and may not be a value for which SIZE_BITS_BY_LENGTH of this dictionary is 0.

If a custom transforms list is set without context dependency, then the following behavior of the RFC 7932 decoder [RFC7932] is overridden:

> The "List of Word Transformations" from Appendix B is

overridden by one or more lists of custom prefixes, suffixes and transform operations.

The transform_id must be smaller than the number of transforms given in the custom transforms list.

If the dictionary is context dependent, it includes a lookup table of 64 word list and transform list combinations. When resolving a static dictionary word, the decoder computes the literal context id, as in section 7.1. of [RFC7932]. The literal context id is used as index in the lookup tables to select the word list and transforms to use. If the dictionary is not context dependent, this id is implicitely 0 instead.

If a distance goes beyond the dictionary for the current id and multiple word list / transform list combinations are defined, then a next dictionary is used in the following order: if not context dependent, the same order as defined in the shared dictionary. If context dependent, the index matching the current context is used first, the same order as defined in the shared dictionary excluding the current context are used next.

3.1.1. Transform Operations

A shared dictionary may include custom word transformations, to replace those specified in Section 8 and Appendix B of [RFC7932]. A transform consists of a possible prefix, a transform operation, for some operations a parameter, and a possible suffix. In the shared dictionary format, the transform operation is represented by a numerical ID, listed in the table below.

ID	Operation
0	Identity
1	OmitLast1
2	OmitLast2
3	OmitLast3
4	OmitLast4
5	OmitLast5
6	OmitLast6
7	OmitLast7
8	OmitLast8
9	OmitLast9
10	FermentFirst
11	FermentAll
12	OmitFirst1
13	OmitFirst2

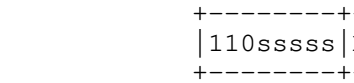
14	OmitFirst3
15	OmitFirst4
16	OmitFirst5
17	OmitFirst6
18	OmitFirst7
19	OmitFirst8
20	OmitFirst9
21	ShiftFirst (by PARAMETER)
22	ShiftAll (by PARAMETER)

Operations 0 to 20 are specified in Section 8 in [RFC7932]. ShiftFirst and ShiftAll transform specifically encoded SCALARs.

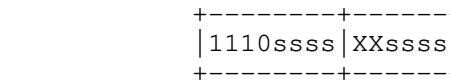
A SCALAR is a 7-, 11-, 16- or 21-bit unsigned integer encoded with 1, 2, 3 or 4 bytes respectively with following bit contents:

7-bit SCALAR:

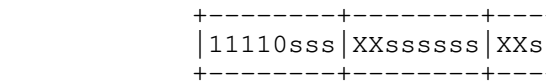
11-bit SCALAR:



16-bit SCALAR:



21-bit SCALAR:



Given the input bytes matching SCALAR encoding pattern, the SCALAR value is obtained by concatenation of the "s" bits, with the most significant bits coming from the earliest byte. The "X" bits could have arbitrary value.

An ADDEND is defined as the result of limited sign extension of 16-bit unsigned PARAMETER:

> At first the PARAMETER is zero-extended to 32 bits. After this, if the resulting value is greater or equal than 0x8000, then 0xFF0000 is added.

ShiftAll starts at the beginning of the word and repetitively applies the following transform until the whole word is transformed:

If the next untransformed byte matches the first byte of the 7-, 11-, 16- or 21-bit SCALAR pattern, then:

If the untransformed part of the word is not long enough to match the whole SCALAR pattern, then the whole word is marked as transformed.

Otherwise, let SHIFTED be the sum of the ADDEND and the encoded SCALAR. The lowest bits from SHIFTED are written back into the corresponding "s" bits. The "0", "1" and "X" bits remain unchanged. Next, 1, 2, 3 or 4 not transformed bytes marked as transformed, according to the SCALAR pattern length.

Otherwise, the next untransformed byte is marked as transformed.

ShiftFirst applies the same transform as ShiftAll, but does not iterate.

3.2. LZ77 Dictionaries

If an LZ77 dictionary is set, then the decoder treats this as a regular LZ77 copy, but behaves as if the bytes of this dictionary are accessible as the uncompressed bytes outside of the regular LZ77 window for backwards references.

Let LZ77_DICTIONARY_LENGTH be the length of the LZ77 dictionary. Then word_id, described in Section 8 in [RFC7932], is redefined as:

word_id = distance - (max allowed distance + 1 + LZ77_DICTIONARY_LENGTH)

For the case when LZ77_DICTIONARY_LENGTH is 0, word_id matches the [RFC7932] definition.

Let dictionary_address be

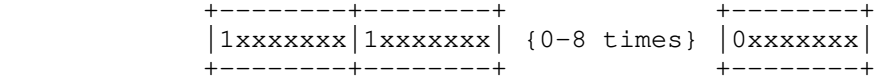
LZ77_DICTIONARY_LENGTH + max allowed distance - distance

Then distance values of <length, distance> pairs [RFC7932] in range (max allowed distance + 1)..(LZ77_DICTIONARY_LENGTH + max allowed distance) are interpreted as references starting in the LZ77 dictionary at the byte at dictionary_address. If length is longer than (LZ77_DICTIONARY_LENGTH - dictionary_address), then the reference continues to copy (length - LZ77_DICTIONARY_LENGTH +

dictionary_address) bytes from the regular LZ77 window starting at the beginning.

4. Varint Encoding

A varint is encoded in base 128 in one ore more bytes as follows:



where the "x" bits of the first byte are the least significant bits of the value and the "x" bits of the last byte are the most significant bits of the value. The last byte must have its MSB set to 0, all other bytes to 1 to indicate there is a next byte.

The maximum allowed amount of bits to read is 63 bits, if the 9th byte is present and has its MSB set then the stream must be considered as invalid.

5. Shared Dictionary Stream

The shared dictionary stream encodes a custom dictionary for brotli including custom words and/or custom transformations. A shared dictionary may appear standalone or as contents of a resource in a framing format container.

A compliant shared brotli dictionary stream must have the following format:

2 bytes: file signature, in hexadecimal the bytes 91, 0.

varint: LZ77_DICTIONARY_LENGTH, number of bytes for a LZ77 dictionary, or 0 if there is none.
The maximum allowed value is the maximum possible sliding window size of brotli or of large window brotli.

LZ77_DICTIONARY_LENGTH bytes: contents of the LZ77 dictionary.

1 byte: NUM_CUSTOM_WORD_LISTS, may have value 0 to 64

NUM_CUSTOM_WORD_LISTS times a word list, with the following format for each word list:

28 bytes: SIZE_BITS_BY_LENGTH, array of 28 unsigned 8-bit integers, indexed by word lengths 4 to 31. The value represents log2(number of words of this length),

with the exception of 0 meaning 0 words of this length. The max allowed length value is 15 bits. OFFSETS_BY_LENGTH is computed from this as OFFSETS_BY_LENGTH[i + 1] = OFFSETS_BY_LENGTH[i] + (SIZE_BITS_BY_LENGTH[i] ? (i << SIZE_BITS_BY_LENGTH[i]) : 0)

N bytes: words dictionary data, where N is OFFSETS_BY_LENGTH[31] + (SIZE_BITS_BY_LENGTH[31] ? (31 << SIZE_BITS_BY_LENGTH[31]) : 0), first all the words of shortest length, then all words of the next length, and so on, where for each length there are either 0 or a positive power of two amount of words.

1 byte: NUM_CUSTOM_TRANSFORM_LISTS, may have value 0 to 64

NUM_CUSTOM_TRANSFORM_LISTS times a transform list, with the following format for each transform list:

2 bytes: PREFIX_SUFFIX_LENGTH, the length of prefix/suffix data. Must be at least 1 because the list must always end with a zero-length stringlet even if empty.

NUM_PREFIX_SUFFIX times: prefix/suffix stringlet. NUM_PREFIX_SUFFIX is the amount of stringlets parsed and must be in range 1..256.

1 byte: STRING_LENGTH, the length of the entry contents. 0 for the last (terminating) entry of the transform list. For other entries STRING_LENGTH must be in range 1..255. The 0 entry must be present and must be the last byte of the PREFIX_SUFFIX_LENGTH bytes of prefix/suffix data, else the stream must be rejected as invalid.

STRING_LENGTH bytes: contents of the prefix/suffix.

1 byte: NTRANSFORMS, amount of transformation triplets.

NTRANSFORMS times: data for each transform:

1 byte: index of prefix in prefix/suffix data; must be less than NUM_PREFIX_SUFFIX.

1 byte: index of suffix in prefix/suffix data; must be less than NUM_PREFIX_SUFFIX.

1 byte: operation index, must be an index in the table of operations listed in the Section "Transform Operations".

If and only if at least one transform has operation index ShiftFirst or ShiftAll:

NTRANSFORMS times:

2 bytes: parameters for the transform. If the transform does not have type ShiftFirst or ShiftAll, the value must be 0. ShiftFirst and ShiftAll interpret these bytes as an unsigned 16-bit integer.

if NUM_CUSTOM_WORD_LISTS > 0 or NUM_CUSTOM_TRANSFORM_LISTS > 0 (else implicitly NUM_DICTIONARIES is 1 and points to the brotli built-in and there is no context map)

1 byte: NUM_DICTIONARIES, may have value 1 to 64. Each dictionary is a combination of a word list and a transform list. Each next dictionary is used when the distance goes beyond the previous. If a CONTEXT_MAP is enabled, then the dictionary matching the context is moved to the front in the order for this context.

NUM_DICTIONARIES times: the DICTIONARY_MAP:

1 byte: index into a custom word list, or value NUM_CUSTOM_WORD_LISTS to indicate to use the brotli [RFC7932] built-in default word list

1 byte: index into a custom transform list, or value NUM_CUSTOM_TRANSFORM_LISTS to indicate to use the brotli [RFC7932] built-in default transform list

1 byte: CONTEXT_ENABLED, if 0 there is no context map, if 1 a context map used to select the dictionary is encoded below

If CONTEXT_ENABLED is 1, a context map for the 64 brotli [RFC7932] literals contexts:

64 bytes: CONTEXT_MAP, index into the DICTIONARY_MAP for the first dictionary to use for this context

6. Large Window Brotli Compressed Data Stream

Large window brotli allows a sliding window beyond the 24-bit maximum of regular brotli [RFC7932].

The compressed data stream is backwards compatible to brotli [RFC7932], and may optionally have the following differences:

Encoding of WBITS in the stream header: the following new pattern of 14 bits is supported:

8 bits: value 00010001, to indicate a large window brotli stream

6 bits: WBITS, must have value in range 10 to 62

Distance alphabet: if the stream is a large window brotli stream, the maximum number of extra bits is 62 and the theoretical maximum size of the distance alphabet is (16 + NDIRECT + (124 << NPOSTFIX)). This overrides the value for the distance alphabet size given in Section 3.3. of [RFC7932] and affects the amount of bits in the encoding of the Simple Prefix Code for distances as described in Section 3.4 of [RFC7932].
An additional limitation to distances, despite the large allowed alphabet size, is that the alphabet is not allowed to contain a distance symbol able to represent a distance larger than ((1 << 63) - 4) when its extra bits have their maximum value. It depends on NPOSTFIX and NDIRECT when this can occur.

A decoder that does not support 64-bit integers may reject a stream if WBITS is higher than 30 or a distance symbol from the distance alphabet is able to encode a distance larger than 2147483644.

7. Shared Brotli Compressed Data Stream

The format of a shared brotli compressed data stream without framing format is backwards compatible with brotli [RFC7932], with the following optional differences:

*) LZ77 dictionaries as described above are supported

*) Custom static dictionaries replacing or extending the static dictionary of brotli [RFC7932] with different words or transforms are supported

*) The stream may have the format of regular brotli [RFC7932], or the format of large window brotli as described in section 6.

8. Shared Brotli Framing Format Stream

A compliant shared brotli framing format stream has the format described below.

8.1. Main Format

4 bytes: file signature, in hexadecimal the bytes 91, 0a, 42, 52. The first byte contains the invalid WBITS combination for brotli [RFC7932] and large window brotli.

1 byte: container flags, 8 bits with meanings:

bit 0 and 1: version indicator, must be 00

bit 2: if 0, the file contains no final footer, may not contain any metadata chunks, may not contain a central directory, and may encode only a single resource (using one or more data chunks). If 1, the file may contain one or more resources, metadata, central directory, and must contain a final footer.

multiple times: a chunk, each with the format specified in section 8.2

8.2. Chunk Format

varint: length of this chunk excluding this varint but including all next header bytes and data. If the value is 0, then the chunk type byte is not present and the chunk type is assumed to be 0.

1 byte: CHUNK_TYPE
0: padding chunk
1: metadata chunk
2: data chunk
3: first partial data chunk
4: middle partial data chunk
5: last partial data chunk
6: footer metadata chunk
7: global metadata chunk
8: repeat metadata chunk
9: central directory chunk
10: final footer

if CHUNK_TYPE is not padding chunk, central directory or final footer:

```
      1 byte: CODEC:

         0: uncompressed

         1: keep decoder

         2: brotli

         3: shared brotli

   if CODEC is not "uncompressed":

      varint: uncompressed size in bytes of the data contained
              within the compressed stream

   if CODEC is "shared brotli"

      1 byte: amount of dictionary references. Multiple dictionary
              references are possible with the following
              restrictions: there can be maximum 1 serialized
              dictionary, and maximum 15 prefix dictionaries (a
              serialized dictionary may already contain one of
              those). Circular references are not allowed (any
              dictionary reference that directly or indirectly
              uses this chunk itself as dictionary).

      per dictionary reference:

         1 byte: flags:

            bit 0 and 1: dictionary source:

               00: Internal dictionary reference to a full resource
                   by pointer, which can span one or more chunks.
                   Must point to a full data chunk or a first
                   partial data chunk.

               01: Internal dictionary reference to single chunk
                   contents by pointer. May point to any chunk with
                   content (data or metadata). If partial data
                   chunk, only this part is the dictionary. In this
                   case, the dictionary type is not allowed to be a
                   serialised dictionary.

               10: Reference to a dictionary by hash code of a
                   resource. The dictionary can come from an
                   external source such as a different container.
                   The user of the decoder must be able to provide
```

the dictionary contents given its hash code (even if it comes from this container itself), or treat it as an error when the user does not have it available.

11: invalid bit combination

bit 2 and 3: dictionary type:

00: prefix dictionary, set in front of the sliding window

01: serialized dictionary in the shared brotli format as specified in section 5.

10: invalid bit combination

11: invalid bit combination

bit 4-7: must be 0

if hash-based:

1 byte: type of hash used. Only supported value: 3, indicating 256-bit Highwayhash.

32 bytes: 256-bit Highwayhash checksum to refer to dictionary.

if pointer based: varint encoded pointer to its chunk in this container. The chunk must come earlier in the container than the current chunk.

X bytes: extra header bytes, depending on CHUNK_TYPE. If present, they are specified in the subsequent sections.

remaining bytes: the chunk contents. The uncompressed data in the chunk content depends on CHUNK_TYPE and is specified in the subsequent sections. The compressed data has following format depending on CODEC:

*) uncompressed: the raw bytes

*) if "keep decoder", the continuation of the compressed stream which was interrupted at the end of the previous chunk. The decoder from the previous chunk must be used and its state it had at the end of the previous chunk

must be kept at the start of the decoding of this chunk.

*) brotli: the bytes are in brotli format [RFC7932]

*) shared brotli: the bytes are in the shared brotli format specified in section 7

8.3. Metadata Format

All the metadata chunk types use the following format for the uncompressed content:

Per field:

2 bytes: code to identify this metadata field. This must be two lowercase or two uppercase alpha ascii characters. If the decoder encounters a lowercase field that it does not recognise for the current chunk type, non-ascii characters or non-alpha characters, the decoder must reject the data stream as invalid. Uppercase codes may be used for custom user metadata and can be ignored by a compliant decoder.

varint: length of the content of this field in bytes, excluding the code bytes and this varint

N bytes: the contents of this field

The last field is reached when the chunk content end is reached. If the length of the last field does not end at the same byte as the end of the uncompressed content of the chunk, the decoder must reject the data stream as invalid.

8.4. Chunk Specifications

8.4.1. Padding Chunk (Type 0)

All bytes in this chunk must be zero, except for the initial varint that specifies the remaining chunk length.

Since the varint itself takes up bytes as well, when the goal is to introduce an amount of padding bytes, the dependence of the length of the varint on the value it encodes must be taken into account.

A single byte varint with value 0 is a padding chunk of length 1. For more padding, use higher varint values. Do not use multiple shorter padding chunks, since this is slower to decode.

8.4.2. Metadata Chunk (Type 1)

This chunk contains metadata that applies to the resource whose beginning is encoded in the subsequent data chunk or first partial data chunk.

The contents of this chunk follows the format described in Section 8.3.

The following field types are recognised:

id: name field. May appear 0 or 1 times. Has the following format:

N bytes: name in UTF-8 encoding, length determined by the field length. Treated generically but may be used as filename. If used as filename, forward slashes '/' should be used as directory separator, relative paths should be used and filenames ending in a slash with 0-length content in the matching data chunk should be treated as an empty directory.

mt: modification type. May appear 0 or 1 times. Has the following format:

8 bytes: microseconds since epoch, as a little endian signed twos complement 64-bit integer

custom user field: any two uppercase ASCII characters.

8.4.3. Data Chunk (Type 2)

A data chunk contains the actual data of a resource.

This chunk has the following extra header bytes:

1 byte: flags:

bit 0: if true, indicates this is not a resource that should be output implicitly as part of extracting resources from this container. Instead, it may be referred to only explicitly, e.g. as a dictionary reference by hash code or offset. This flag should be set for data used as dictionary to improve compression of actual resources.

```
        bit 1: if true, hash code is given

        bits 2-7: must be zero

      if hash code is given:

        1 byte: type of hash used. Only supported value: 3,
                indicating 256-bit Highwayhash.

        32 bytes: 256-bit Highwayhash checksum of the uncompressed
                  data
```

The uncompressed content bytes of this chunk are the actual data of the resource.

8.4.4. First Partial Data Chunk (Type 3)

This chunk contains partial data of a resource. This is the first chunk in a series containing the entire data of the resource.

The format of this chunk is the same as the format of a Data Chunk (Section 8.4.3) except for the differences noted below.

The second bit of flags must be set to 0 and no hash code given.

The uncompressed data size is only of this part of the resource, not of the full resource.

8.4.5. Middle Partial Data Chunk (Type 4)

This chunk contains partial data of a resource, and is neither the first nor the last part of the full resource.

The format of this chunk is the same as the format of a Data Chunk (Section 8.4.3) except for the differences noted below.

The first and second bits of flags must be set to 0.

The uncompressed data size is only of this part of the resource, not of the full resource.

8.4.6. Last Partial Data Chunk (Type 5)

This chunk contains the final piece of partial data of a resource.

The format of this chunk is the same as the format of a Data Chunk (Section 8.4.3) except for the differences noted below.

The first bit of the flags must be set to 0.

If a hash code is given, the hash code of the full resource (concatenated from all previous chunks and this chunk) is given in this chunk.

The uncompressed data size is only of this part of the resource, not of the full resource.

The type of this chunk indicates that there are no further chunk encoding this resource, so the full resource is now known.

8.4.7. Footer Metadata Chunk (Type 6)

This metadata applies to the resource whose encoding ended in the preceding data chunk or last partial data chunk.

The contents of this chunk follows the format described in Section 8.3.

There are no lowercase field types defined for footer metadata. Uppercase field types can be used as custom user data.

8.4.8. Global Metadata Chunk (Type 7)

This metadata applies to the whole container instead of a single resource.

The contents of this chunk follows the format described in Section 8.3.

There are no lowercase field types defined for footer metadata. Uppercase field types can be used as custom user data.

8.4.9. Repeat Metadata Chunk (Type 8)

These chunks optionally repeat metadata that is interleaved between data chunks. To use these chunks, it is necessary to also read additional information, such as pointers to the original chunks, from the central directory.

The contents of this chunk follows the format described in Section 8.3.

This chunk has an extra header byte:

1 byte: chunk type of repeated chunk (metadata chunk or footer metadata chunk)

This set of chunks must follow the following restrictions:

- It is optional whether or not repeat metadata chunks are present.
- If they are present, then they must be present for all metadata chunks and footer metadata chunks.
- There may be only 1 repeat metadata chunk per repeated metadata chunk.
- They must appear in the same order as the chunks appear in the container, which is also the same order as listed in the central directory.
- Compression of these chunks is allowed, however it is not allowed to use any internal dictionary except an earlier repeat metadata chunk of this series, and it is not allowed for a metadata chunk to keep the decoder state if the previous chunk is not a repeat metadata chunk. That is, the series of metadata chunks must be decompressible without using other chunks of the framing format file.

The fields contained in this metadata chunk must follow the following restrictions:

- If a field is present, it must exactly match the corresponding field of the copied chunk.
- It is allowed to leave out a field that is present in the copied chunk.
- If a field is present, then it must be present in *all* other repeat metadata chunks when the copied chunk contains this field. In other words, if you know you can get the name field from a repeat chunk, you know that you will be able to get all names of all resources from all repeat chunks.

8.4.10. Central Directory Chunk (Type 9)

The central directory chunk, along with the repeat metadata chunks, allow to quickly find and list compressed resources in the container file.

The central directory chunk is always uncompressed and does not have the codec byte. It instead has the following format:

varint: pointer into the file where the repeat metadata chunks are

located, or 0 if they are not present

per chunk listed:

varint: pointer into the file where this chunk begins

varint: amount of header bytes N used below

N bytes: copy of all the header bytes of the pointed at chunk, including total size, chunk type byte, codec, uncompressed size, dictionary references, X extra header bytes. The content is not repeated here.

The last listed chunk is reached when the end of the contents of the central directory are reached. If the end does not match the last byte of the central directory, the decoder must reject the data stream as invalid.

If present, the central directory must list all data and metadata chunks of all types.

8.4.11. Final Footer Chunk (Type 10)

Chunk that closes the file, only present if in the initial container header flags bit 2 was set.

This chunk has the following content, always uncompressed:

reversed varint: size of this entire framing format file, including these bytes themselves, or 0 if this size is not given

reversed varint: pointer to the start of the central directory, or 0 if there is none

A reversed varint has the same format as a varint, but has its bytes in reversed order and is designed to be parsed from end of file towards the beginning.

8.4.12. Chunk ordering

The chunk ordering must follow the rules described below, if the decoder sees otherwise, it must reject the data stream as invalid.

Padding chunks may be inserted anywhere, even between chunks for which the rules below say no other chunk types may come in between.

Metadata chunks must come immediately before the Data chunks of the resource they apply to.

Footer metadata chunks must come immediately after the Data chunks of the resource they apply to.

There may be only 0 or 1 metadata chunks per resource.

There may be only 0 or 1 footer metadata chunks per resource.

A resource must exist out of either 1 data chunk, or 1 first partial data chunk, 0 or more middle partial data chunks, and 1 last partial data chunk, in that order.

Repeat metadata chunks must follow the rules of section 8.4.9.

There may be only 0 or 1 central directory chunks.

If bit 2 of the container flags is set, there may be only a single resource, no metadata chunks of any type, no central directory, and no final footer.

If bit 2 of the container flags is not set, there must be exactly 1 final footer chunk and it must be the last chunk in the file.

9. Security Considerations

The security considerations for brotli [RFC7932] apply to shared brotli as well.

In addition, the same considerations apply to the decoding of new file format streams for shared brotli, including shared dictionaries, the framing format and the shared brotli format.

The dictionary must be treated with the same security precautions as the content, because a change to the dictionary can result in a change to the decompressed content.

The CRIME attack [CRIME] shows that it’s a bad idea to compress data from mixed (e.g. public and private) sources -- the data sources include not only the compressed data but also the dictionaries. For example, if you compress secret cookies using a public-data-only dictionary, you still leak information about the cookies.

Not only can the dictionary reveal information about the compressed data, but vice versa, data compressed with the dictionary can reveal the contents of the dictionary when an adversary can control parts of data to compress and see the compressed size. On the other hand, if

the adversary can control the dictionary, the adversary can learn information about the compressed data.

The most robust defense against CRIME is not to compress private data (e.g., sensitive headers like cookies or any content with PII). The challenge has been to identify secrets within a vast amount of to be compressed data. Cloudflare uses a regular expression [CLOUDFLARE]. Another idea is to extend existing web template systems (e.g., Soy [SOY]) to allow developers to mark secrets that must not be compressed.

A less robust idea, but easier to implement, is to randomize the compression algorithm, i.e., adding randomly generated padding, varying the compression ratio, etc. The tricky part is to find the right balance between cost and security, i.e., on one hand we don't want to add too much padding because it adds a cost to data, on the other hand we don't want to add too little because the adversary can detect a small amount of padding with traffic analysis.

Another defense in addition is to not use dictionaries for cross-domain requests, and only use shared brotli for the response when the origin is the same as where the content is hosted (using CORS). This prevents an adversary from using a private dictionary with user secrets to compress content hosted on the adversary's origin. It also helps prevent CRIME attacks that try to benefit from a public dictionary by preventing data compression with dictionaries for requests that do not originate from the host itself.

The content of the dictionary itself should not be affected by external users, allowing adversaries to control the dictionary allows a form of chosen plaintext attack. Instead, only base the dictionary on content you control or generic large scale content such as a spoken language, and update the dictionary with large time intervals (days, not seconds) to prevent fast probing.

The use of highwayhash for dictionary identifiers does not guarantee against collisions in an adversarial environment and is intended to be used for identifying the dictionary within a trusted, known set of dictionaries. In an adversarial environment, users of shared brotli should use another mechanism to validate a negotiated dictionary, such as using a cryptographically-proven secure hash.

10. IANA Considerations

This document has no IANA actions.

11. Normative References

[RFC7932] Alakuijala, J., Szabadka, Z., "Brotli Compressed Data Format", RFC 7932, Google, Inc., July 2016. http://www.ietf.org/rfc/rfc7932.txt

12. Informative References

[LZ77] Ziv, J., Lempel, A., "A Universal Algorithm for Sequential Data Compression". IEEE Transactions on Information Theory. 23 (3): 337–343., May 1977.

[CLOUDFLARE] https://blog.cloudflare.com/a-solution-to-compression-oracles-on-the-web/

[SOY] https://developers.google.com/closure/templates/

[CRIME] https://www.cve.org/CVERecord?id=CVE-2012-4929

Acknowledgments

The authors would like to thank Robert Obryk for suggesting improvements to the format and the text of the specification.

Authors' Addresses

Jyrki Alakuijala
Google, Inc.

Email: jyrki@google.com

Thai Duong
Google, Inc.

Email: thaidn@google.com

Evgenii Kliuchnikov
Google, Inc.

Email: eustas@google.com

Zoltan Szabadka
Google, Inc.

Email: szabadka@google.com

Lode Vandevenne (editor)
Google, Inc.

Email: lode@google.com