INTERNET-DRAFT
"Internet Protocol Five Fields - Addressing Architecture",
Alexey Eromenko, 2016-09-29,
<draft-eromenko-ipff-addressing-05.txt>
expiration date: 2017-03-29

Intended status: Standards Track

A. Eromenko
September 2016

IP Version 5 Addressing Architecture
aka Internet Protocol "Five Fields" (IP-FF; draft)

Abstract

This specification defines the addressing architecture of the IP Version 5 (IPFF) protocol. The document includes the IPFF addressing model, text representations of IPFF addresses, definition of IPFF unicast addresses, anycast addresses, and multicast addresses.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This specification defines the addressing architecture of the IP Version 5 protocol. It includes the basic formats for the various types of IPFF addresses (unicast, anycast, and multicast).

2. IPFF Addressing

The main benefit of IPFF, is that it looks familiar to all, whom ever used IPv4, just with an extra field. So "five fields".

Examples:

192.168.510.971.11

10.0.0.0.1

382.201.769.25.133

IPFF addresses are 50-bit identifiers for interfaces and sets of interfaces (where "interface" is as defined in Section 2 of [IPFF]). Each field is 10-bits wide, but due to human-memory constraits, values are limited to "999" per field.

Why go to the great length of dis-guard perfectly useable bits?
Why not up to 1023.1023.1023.1023.1023, as 10-bit fields imply?
Bits in IPFF are a bit under-utilized.

Well, IP-FF was designed with humans in mind.
Limit of three-digits allows for easy readability, comparison, memorization and visualizing network topology in human memory, just like in IPv4, unlike IPv6.
This is a small sacrifice of total address range with a huge human benefit.

And in modern computing devices are typically represented by 64-bit unsigned integer, pre-padded with 14-bits of zeroes.

There are several types of addresses:

Unicast: An identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address.

Anycast: An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to an anycast address is delivered to one of the interfaces identified by that address (the "nearest" one, according to the routing protocols' measure of distance).

Traditional Multicast:

An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to a multicast address is delivered to all interfaces identified by that address.
Traditional Multicast mandates IGMP advertisement, and allows for IGMP snooping.

Silent Multicast:

An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to a multicast address is delivered to all interfaces identified by that address.
Silent Multicast doesn't do any IGMP advertisement, and it's done for simplicity.
Typically has local-link significance.

Broadcast: An identifier for all node's interfaces
on a local-link.

Unlike IPv6, the use of IGMP is required only for Traditional Multicast; That is a system can join a Multicast group, but not send any message about it. This is called "silent multicast mode".

In IPFF, all zeros and all ones are legal values for any field, unless specifically excluded. Specifically, prefixes may contain, or end with, zero-valued fields.

2.1. Addressing Model

IPFF addresses of all types are assigned to interfaces, not nodes. An IPFF unicast address refers to a single interface. Since each interface belongs to a single node, any of that node's interfaces' unicast addresses may be used as an identifier for the node.

A single interface may also have multiple IPFF addresses of any type (unicast, anycast, and multicast), but typically has only one primary unicast address.

Currently, IPFF continues the IPv4 model in that a subnet prefix is associated with one link. Multiple subnet prefixes may be assigned to the same link, one per IP address.

2.2. Text Representation of Addresses

There are two conventional forms for representing IPFF addresses as text strings:

1. The preferred form is x.x.x.x.x, where the 'x's are one to three decimal digits.

2. Special case: shortening
   In order to make writing addresses containing zero fields easier, a special syntax is available to compress the zeros. The use of ".." indicates fields of 10 bits of zeros.

   For example, the following addresses

   ```
   0.0.0.0.1                      the loopback address
   0.0.0.0.0                      the unspecified address
   ```

   may be represented as

   ```
   ..1                            the loopback address
   ..0                            the unspecified address
   ```

Note, that this shortened form is allowed only for loopback and unspecified addresses, as a unicast & multicast are unlikely to have too many zeroes to warrant a special case.

2.3. Text Representation of Address Prefixes

The text representation of IPFF address prefixes is similar to the way IPv4 address prefixes are written in Classless Inter-Domain Routing (CIDR) notation [CIDR]. An IPFF address prefix is represented by the notation:

```
IPFF-address/prefix-length
```

where

IPFF-address   is an IPFF address in any of the notations listed in Section 2.2.

prefix-length   is a decimal value specifying how many of the leftmost contiguous bits of the address comprise the prefix. 40 is default, if not specified.
That means, 4 fields for the network portion, and one field of 10-bits for the hosts.
This is an equivalent of an IPv4 subnet mask.

Examples:

```
192.168.510.971.11/30

10.0.0.0.1   (implies /40)

382.201.769.25.133/34
```

2.4. Address Type Identification

The type of an IPFF address is identified by the high-order bits of the address, as follows:

Address type	Binary prefix	IPFF notation	Section
Unspecified	00...0 (50 bits)	..0/50	2.5.2
Loopback	00...1 (50 bits)	..1/50	2.5.3
Silent Multicast	01100011 000001001	99.9../20	2.7
Traditional Multicast	01100011 000001000	99.8../20	2.7
Broadcast		99.9.0.0.1	2.8
Unicast	(everything else)		

Anycast addresses are taken from the unicast address spaces (of any scope) and are not syntactically distinguishable from unicast addresses.

2.5. Unicast Addresses

IPFF unicast addresses are aggregatable with prefixes of arbitrary bit-length, similar to IPv6 and IPv4 addresses under Classless Inter-Domain Routing.

2.5.1. Private addresses

Similarly to IPv4 private addresses defined in RFC-1918, IPFF has several:

```
10.0.0.0.0    -   10.999.999.999.999  (10.x.x.x.x/10 prefix)

172.16.0.0.0  -   172.31.999.999.999  (172.16-31.x.x.x/16 prefix)

192.168.0.0.0 -   192.168.999.999.999 (192.168.x.x.x/20 prefix)
```

Those were chosen to be visually similar to their IPv4 counterparts, mainly to simplify migration from current IPv4 networks.

Those address ranges must not be routed on the Internet, but they can be routed inside an organization.

Private addresses are intended for Enterprises that are either not connected to the Internet, or for hosts that are hidden nehind a NAT (Network Address Translation) device, typically a router.

An enterprise that decides to use IP addresses out of the address space defined in this document can do so without any coordination with IANA or an Internet registry. The address space can thus be used by many enterprises. Addresses within this private address space will only be unique within the enterprise, or the set of enterprises which choose to cooperate over this space so they may communicate with each other in their own private internet.

2.5.2. AutoIP / link-local unicast addresses

In addition, for Auto IP (aka Zeroconf aka Stateless DHCP aka optional link-local addresses), another range was defined:

```
0.169.254.x.x/30
```

This address ranges should not be routed on the Internet, and not inside an organization.
Those are considered unique only in a layer-2 broadcast domain, link scope, therefore should not pass a router without a network address translation.

Because link-local addresses are not required in IPFF, and not recommended, the procedure of getting or generating them is not described in this document. The packet originating from those addresses should have a TTL value of 1.

2.5.3. The Unspecified Address

The address 0.0.0.0.0 is called the unspecified address. It must never be assigned to any node. It indicates the absence of an address. One example of its use is in the Source Address field of any IPFF packets sent by an initializing host before it has learned its own address.

The unspecified address must not be used as the destination address of IPFF packets or in IPFF Routing headers. An IPFF packet with a source address of unspecified must never be forwarded by an IPFF router.

2.5.4. The Loopback Address

The unicast address 0.0.0.0.1/50 is called the loopback address. Can be shortened to ..1 on input.

It may be used by a node to send an IPFF packet to itself. It must not be assigned to any physical interface.

The loopback address must not be used as the source address in IPFF packets that are sent outside of a single node. An IPFF packet with a destination address of loopback must never be sent outside of a single node and must never be forwarded by an IPFF router. A packet received on an interface with a destination address of loopback must be dropped.

2.5.5. Global Unicast Addresses

The can be anything that is not reserved for other purposes.

That is from 0.0.0.0.0 up to 999.999.999.999.999 but excluding unspecified, loopback, multicast range, private IPs and auto IPs.

Packets received with addresses above "999" in any one field should be dropped.

2.6. Anycast Addresses

An IPFF anycast address is an address that is assigned to more than one different node, with the property that a packet sent to an anycast address is routed to the "nearest" interface having that address, according to the routing protocols' measure of distance.

Anycast addresses are allocated from the unicast address space, using any of the defined unicast address formats. Thus, anycast addresses are syntactically indistinguishable from unicast addresses.

The Router's job is to make sense of them, and route traffic to the nearest node.

2.7. Multicast Addresses

An IPFF multicast address is an identifier for a group of interfaces (typically on different nodes). An interface may belong to any number of multicast groups.

They start with 99.x.x.x.x/10

In IPFF, we have four types of Multicast Addresses:

-Silent Multicast addresses,
-Private Multicast addresses, and
-Public Multicast addresses.

2.7.1. Pre-Defined Multicast Addresses

The following well-known multicast addresses are pre-defined.

Use of these group IDs for any other scope values, with the T flag equal to 0, is not allowed.

Traditional Multicast address range defined:

    99.8.0.0.x/30

Silent Multicast address range defined:

    99.9.0.0.x/30

Private (Traditional) Multicast address range defined:

    99.8.0.1.x/40

Reserved Multicast Addresses:

    99.9.0.0.1 = All Nodes (= Broadcast)
    99.8.0.0.2 = All Routers
    99.9.0.0.3 = DHCP Servers
    99.9.0.0.4 = DHCP Clients

Note, that "All Routers" use a traditional Multicast address with IGMP advertisement, starting with 99.8.x

DHCP clients and servers, since are not generating lots of traffic, do not need to advertise themselves for IGMP groups, and therefore are "Silent listeners". They can be minimal embedded devices, and don't have to implement the full IGMP protocol.

2.7.2. Silent Multicast Addresses

The big difference between silent (link-local) and traditional Multicast addresses is "IGMP".
Silent multicast addresses are logically in-between traditional multicast, that advertises itself by IGMP, and broadcast that does not.

It means, that services, that nodes listening on Link-local Multicast addresses *DO NOT* advertise that they have joined Multicast group X, but nodes listening on Private Multicast addresses *DO* advertise via IGMP.

Link-local Multicast address listeners also called "silent multicast listeners".

"Silent Multicast" mode compared to traditional Multicast:
Benefit is two fold: simple IPFF implementations are possible, as IGMP implementation is not needed, and if you want to receive only a few packets, you really don't need to flood whole network by IGMP advertisements.
So on the operating system level, a listener is still required join a "Multicast group", but no advertisement is sent.

"Silent Multicast" mode compared to traditional broadcast:
...nodes still get Layer 2 flooding, like broadcasts, but using different IP addresses and different layer 2 addresses, nodes can filter unnecessary traffic at layer 2 (by a node's Ethernet controller, for example), instead of by Transport layer, saving CPU time.

Ethernet examples are part of the [IPFF-ARPv5] spec.

2.8. Broadcast Address

99.9.0.0.1 the "All Nodes" silent multicast address doubles down as a broadcast address, on a link-local scope.

This address is used by DAD (Duplicate Address Detection) mechanism, and intended to be used only for network discovery purposes, but most other traditionally-Broadcast applications should avoid it's use, and opt for a Multicast or a Silent Multicast address instead.

2.9. A Node's Required Addresses

Only few:
Loopback address. 0.0.0.0.1
Additionally, it must listen on the broadcast address of 99.9.0.0.1,
plus Node Multicast address.
At least one Unicast address, plus prefix (similar to subnet mask).
That's it !

Beyond required addresses, here are some RECOMMENDED addresses:
-DNS address + backup DNS address(es)
-Default Gateway + backup Default Gateway(s)

3. Routing and Invalid Addresses

Routing should be done in full 50-bit addresses, but if any field is bigger than 3 decimal digits "999", this traffic is invalid.

Actual enforcement of "999" limit MUST be done at Transport-layer, for connection-oriented protocols only once by end-node, when socket is created. (i.e incoming TCP session or outgoing TCP session, only during <SYN> and <ACK> stage, and only for connection-oriented Transport protocols). Additionally, it MUST be enforced by both DNS clients and DNS servers.

Connectionless Transport protocols, such as UDP, are not required to check this 999 limit.
This traffic is still invalid, but check is not actively enforced.

End-nodes and middleboxes MAY check this 999 field limit per-packet, at the IP layer, and silently drop such invalid traffic.
No ICMP message needs to be sent.
Core Routers are not required to check this, for efficiency reasons.

3.1. Default Gateway

All-zero-bits subnet mask, or /0 prefix, together with 0.0.0.0.0 IP address (or ..0 for short), means a default gateway.

3.2. Special meaning of /50 subnet prefix

All-one-bits subnet mask, or /50 prefix, which is the same written differently, has a special meaning. It means that host has access only to itself, or to a default gateway.

3.3. Non-contiguous wildcard masks

Some firewall vendors allow for Non-contiguous masks in IPv4, and the same tradition can be kept with IPFF, but only for traffic filtering purposes.

Wildcard mask is somewhat similar to the IPv4 subnet mask, but in reverse. 1-bits are for *host portion*, while 0-bits are for *network portion*. A Non-contiguous IPFF wildcard mask may look like:
0.0.1018.517.1023

All connectivity and routing-related decisions must use a prefix notation, and thus be contiguous.

NOTE: While quad-digits per field (like .1023) are prohibited from IPFF addresses, they are okay for the purpose of Wildcard masks.

3.4. Not all subnets are born equal

Let's take a /45 prefix, for example.
It means that I want to divide my 5th field into 32 subnets
with 32 hosts each.

10.0.0.0.0/45 subnet will have 32 hosts. (up to .31)
10.0.0.0.32/45 subnet will have 32 hosts. (up to .63)
...
But what about the last subnet ?

10.0.0.0.992/45 subnet should have 32 hosts...
But it can't ! Why ?

Theoretically, it should have up to .1023 but the limit is .999,
This implies that some subnets may be not full, or not usable.
So this last subnet will have only 8 hosts ! (up to .999)

The limit was introduced to conserve *human* memory, which is more precious than computer memory nowadays... So we, humans, need to remember only three digits per field, instead of four.

4. Acknowledgements

The author would like to thank all previous IPv4 and IPv6 developers for their hard work, and benefit for humanity.
This document is based on RFC-4291 (IPv6 addressing architecture) and RFC-1918 (private IPv4 addressing).

Authors' Contacts

Alexey Eromenko

Israel

Skype: Fenix_NBK_
EMail: al4321@gmail.com
Facebook: https://www.facebook.com/technologov

Based on the hard work of "Stephen E. Deering" and "Robert M. Hinden",
from IPv6 project, as well as all previous TCP/IP developers from DARPA.