Manual:IPv6/Address

From MikroTik Wiki

< Manual:IPv6

Contents

Summary

Sub-menu: `/ipv6 address`
Standards: `RFC 4291`

IPv6 uses 16 bytes addresses compared to 4 byte addresses in IPv4. IPv6 address syntax and types are described in RFC 4291.

There are multiple IPv6 address types, that can be recognized by their prefix. RouterOS distinguishes the following:

- multicast (with prefix ff00::/8)
- link-local (with prefix fe80::/10)
- loopback (the address ::1/128)
- unspecified (the address ::/128)
- other (all other addresses, including the obsoleted site-local addresses, and RFC 4193 unique local addresses; they all are treated as global unicast).

One difference between IPv6 and IPv4 addressing is that IPv6 automatically generates a link-local IPv6 address for each active interface that has IPv6 support.

Address Expression

IPv6 addresses are represented a little bit different than IPv4 addresses. For IPv6, the 128-bit address is divided in eight 16-bit blocks, and each 16-bit block is converted to a 4-digit hexadecimal number and separated by colons. The resulting representation is called **colon-hexadecimal**.

In example above IPv6 address in binary format is converted to **colon-hexadecimal** representation

```
0010000000000001 0000010001110000 0001111100001001 0000000100110001
0000000000000000 0000000000000000 0000000000000000 0000000000001001
```

```
2001:0470:1f09:0131:0000:0000:0000:0009
```

IPv6 address can be further simplified by removing leading zeros in each block:

```
2001:470:1f09:131:0:0:0:9
```

As you can see IPv6 addresses can have long sequences of zeros. These contiguous sequence can be compressed to **::**

```
2001:470:1f09:131::9
```

> **Note:** Zero compression can only be used once. Otherwise, you could not determine the number of 0 bits represented by each instance of a double-colon

Prefix

IPv6 prefix is written in `address/prefix-length` format. Compared to IPv4 decimal representation of network mask cannot be used. Prefix examples:

```
2001:470:1f09:131::/64
2001:db8:1234::/48
2607:f580::/32
2000::/3
```

Address Types

Several IPv6 address types exist:

- Unicast
- Anycast
- Multicast

As you can see there are no Broadcast addresses in ipv6 network, compared to IPv4 broadcast functionality was completely replaced with multicast.

Unicast Addresses

Packets addressed to a unicast address are delivered only to a single interface. To this group belong:

- globally unique addresses and can be used to connect to addresses with global scope anywhere.
- link-local addresses
- site-local addresses (FEC0::/48) - deprecated
- special purpose addresses
- compatibility addresses

Global unicast address can be automatically assigned to the node by ***Stateless Address auto-configuration***. `Read More >>`.

Link-local address

A link-local address is required on every IPv6-enabled interface, applications may rely on the existence of a link-local address even when there is no IPv6 routing, that is why link-local address is generated automatically for every active interface using it's interface identifier (calculated EUI-64 from MAC address if present). Address prefix is always FE80::/64 and IPv6 router never forwards link-local traffic beyond the link.

These addresses are comparable to the auto-configuration addresses 169.254.0.0/16 of IPv4.

A link-local address is also required for Neighbor Discovery processes.

> **Note:** If interface is set as bridge port, interface specific link-local address is removed leaving only bridge link-local address

Special purpose address

Address	Description
`Unspecified address (::/128)`	Never assigned to an interface or used as a destination address, used only to indicate the absence of an address. Equivalent to IPv4 0.0.0.0 address.
`loopback address (::1/128)`	Used to identify a loopback interface, enabling a node to send packets to itself. It is equivalent to the IPv4 loopback address of 127.0.0.1.

Compatibility address

Address	Description
`IPv4 compatible address`	used by dual-stack nodes that are communicating with IPv6 over an IPv4 infrastructure. When the IPv4-compatible address is used as an IPv6 destination, IPv6 traffic is automatically encapsulated with an IPv4 header and sent to the destination by using the IPv4 infrastructure. Address is written in following format `::w.x.y.z`, where w.x.y.z is the dotted decimal representation of a public IPv4 address. This address is now deprecated, RFC 4291 section-2.5.5.1
`IPv4 mapped address`	used to represent an IPv4-only node to an IPv6 node. It is used only for internal representation. The IPv4-mapped address is never used as a source or destination address for an IPv6 packet. The IPv6 protocol does not support the use of IPv4-mapped addresses. Address is written in following format: `::ffff:w.x.y.z`, where w.x.y.z is the dotted decimal representation of a public IPv4 address.
`2002::/16`	this prefix is used for 6to4 addressing. Here, an address from the IPv4 network 192.88.99.0/24 is also used.

Multicast address

Most important multicast aspects are:

- traffic is sent to a single address but is processed by multiple hosts;
- group membership is dynamic, allowing hosts to join and leave the group at any time;
- in IPv6, Multicast Listener Discovery (MLD) messages are used to determine group membership on a network segment, also known as a link or subnet;
- host can send traffic to the group's address without belonging to the corresponding group.

A single IPv6 multicast address identifies each multicast group. Each group's reserved IPv6 address is shared by all host members of the group who listen and receive any IPv6 messages sent to the group's address.

Multicast address consists of the following parts: [1] (http://www.ipv6style.jp/files/ipv6/en/tech/20030228/images/1.gif)

- The first 8 bits in multicast address is always 1111 1111 (which is FF in hexadecimal format).
- *Flag* uses the 9th to 12th bit and shows if this multicast address is predefined (well-known) or not. If it is well-known, all bits are 0s.
- *Scope ID* indicates to which scope multicast address belongs, for example, Scope ID=2 is link-local scope.
- *Group ID* is used to specify a multicast group. There are predefined group IDs, such as Group ID=1 - all nodes. Therefore, if multicast address is ff02::1, that means Scope ID=2 and Group ID=1, indicating all nodes in link-local scope. This is analogous to broadcast in IPv4.

Here is the table of reserved IPV6 addresses for multicasting:

Address	Description
FF02::1	The all-nodes address used to reach all nodes on the same link.
FF02::2	The all-routers address used to reach all routers on the same link.
FF02::5	The all-Open Shortest Path First (OSPF) routers address used to reach all OSPF routers on the same link.
FF02::6	The all-OSPF designated routers address used to reach all OSPF designated routers on the same link.
FF02::1:FFXX:XXXX	The solicited-node address used in the address resolution process to resolve the IPv6 address of a link-local node to its link-layer address. The last 24 bits (XX:XXXX) of the solicited-node address are the last 24 bits of an IPv6 unicast address.

The following table is a partial list of IPv6 multicast addresses that are reserved for IPv6 multicasting and registered with the Internet Assigned Numbers Authority (IANA). For complete list of assigned addresses read IANA document (http://www.iana.org/assignments/ipv6-multicast-addresses/).

Multicast addresses can be used to discover nodes in a network. For example, discover all nodes

```
mrz@bumba:/media/aaa/ver$ ping6 ff02::1%eth0
PING ff02::1%eth0(ff02::1) 56 data bytes
64 bytes from fe80::21a:4dff:fe5d:8e56: icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from fe80::20c:42ff:fe0d:2c38: icmp_seq=1 ttl=64 time=4.03 ms (DUP!)
64 bytes from fe80::20c:42ff:fe28:7945: icmp_seq=1 ttl=64 time=5.59 ms (DUP!)
64 bytes from fe80::20c:42ff:fe49:fce5: icmp_seq=1 ttl=64 time=5.60 ms (DUP!)
64 bytes from fe80::20c:42ff:fe21:f1ec: icmp_seq=1 ttl=64 time=5.88 ms (DUP!)
64 bytes from fe80::20c:42ff:fe72:a1b0: icmp_seq=1 ttl=64 time=6.70 ms (DUP!)
```

discover all routers

```
mrz@bumba:/media/aaa/ver$ ping6 ff02::2%eth0
PING ff02::2%eth0(ff02::2) 56 data bytes
64 bytes from fe80::20c:42ff:fe28:7945: icmp_seq=1 ttl=64 time=0.672 ms
64 bytes from fe80::20c:42ff:fe0d:2c38: icmp_seq=1 ttl=64 time=1.44 ms (DUP!)
```

Anycast address

Anycast address is a new type of address incorporated in IPv6.

Anycasting is a new networking paradigm supporting service–oriented Addresses where an identical address can be assigned to multiple nodes providing a specific service. An anycast packet (i.e., one with an anycast destination address) is delivered to one of these nodes with the same anycast address.

Anycast address is not assigned a specific address range. It is assigned from unicast address range.

Interface Identifier

The last 64 bits of an IPv6 address are the interface identifier that is unique to the 64-bit prefix of the IPv6 address. There are several ways how to determine interface identifier:

- EUI-64;
- randomly generated to provide a level of anonymity;
- manually configured.

EUI-64

Traditional interface identifiers for network adapters are 48-bit MAC address. This address consists of a 24-bit manufacturer ID and a 24-bit board ID.

IEEE EUI-64 is a new standard for network interface addressing. The company ID is still 24-bits in length, but the extension ID is 40 bits, creating a much larger address space for network adapters.

To create an EUI-64 address from the interface MAC address:

- 0xFFFE is inserted into the MAC address between the manufacturer ID and the board ID.
- seventh bit of the first byte is reversed.

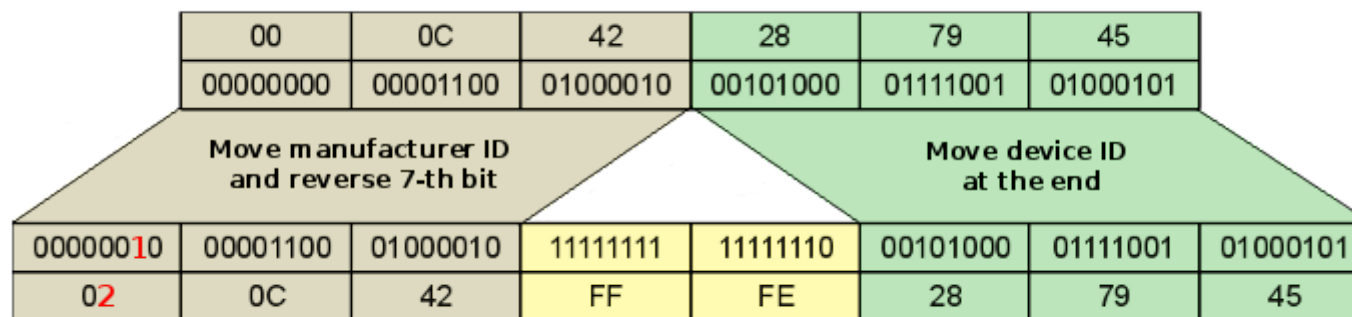Lets make an example with following MAC address 00:0C:42:28:79:45.

Image above illustrates conversion process.

When the result is converted to colon-hexadecimal notation, we get the interface identifier 20C:42FF:FE28:7945. As the result, corresponds link-local address is

```
FE80::20C:42FF:FE28:7945/64
```

In RouterOS, if the *eui-64* parameter of an address is configured, the last 64 bits of that address will be automatically generated and updated using interface identifier. The last bits must be configured to be zero for this case. Example:

```
[admin@MikroTik] > ipv6 address add address=fc00:3::/64 interface=ether3 eui-64=yes
[admin@MikroTik] > ipv6 address print
Flags: X - disabled, I - invalid, D - dynamic, G - global, L - link-local
 #   ADDRESS                              INTERFACE           ADVERTISE
 ...
 5  G fc00:3::20c:42ff:fe1d:3d4/64        ether3              yes
[admin@MikroTik] > interface ethernet set ether3 mac-address=10:00:00:00:00:01
[admin@MikroTik] > ipv6 address print
Flags: X - disabled, I - invalid, D - dynamic, G - global, L - link-local
 #   ADDRESS                              INTERFACE           ADVERTISE
```

```
...
5  G fc00:3::1200:ff:fe00:1/64                    ether3                    yes
```

Properties

Property	Description
address (*Address/Netmask*; Default:)	Ipv6 address. Allowed netmask range is 0..128. Address can also be constructed from the pool if **from-pool** property is specified. For example if address is set to ::1/64 then address will be constructed as follows <prefix_from_pool>::1/64
advertise (*yes* | *no*; Default: **no**)	Whether to enable stateless address configuration. The prefix of that address is automatically advertised to hosts using ICMPv6 protocol. The option is set by default for addresses with prefix length 64. Read more >>
comment (*string*; Default:)	Descriptive name of an item
disabled (*yes* | *no*; Default: **no**)	Whether address is disabled or not. By default it is disabled
eui-64 (*yes* | *no*; Default: **no**)	Whether to calculate EUI-64 address and use it as last 64 bits of the IPv6 address. Read more >>
from-pool (*string*; Default:)	Name of the pool from which prefix will be taken to construct IPv6 address taking last part of the address from **address** property. See example >>
no-dad (*yes* | *no*; Default: **no**)	If set indicates that address is anycast address and Duplicate Address Detection should not be performed.
interface (*string*; Default:)	Name of an interface on which Ipv6 address is set.

Read-only properties

Property	Description
actual-interface (*string*)	Actual interface on which address is set up. For example, if address was configured on ethernet interface and ethernet interface was added to bridge, then actual interface is bridge not ethernet.
dynamic (*yes* | *no*)	Whether address is dynamically created

global (*yes* | *no*)	Whether address is global
invalid (*yes* | *no*)	
link-local (*yes* | *no*)	Whether address is link local

Examples

Manual address configuration

This example shows how to set up simple addressing with global IPv6 addresses between two routers.

R1 configuration:

```
/ipv6 address
add address=2001:DB8::1/64 interface=ether1 advertise=no
```

R2 configuration:

```
/ipv6 address
add address=2001:DB8::2/64 interface=ether1 advertise=no
```

Check address list

```
[admin@R1] /ipv6 address> print
Flags: X - disabled, I - invalid, D - dynamic, G - global, L - link-local
 #    ADDRESS                               FROM-POOL INTERFACE    ADVERTISE
 0  G 2001:db8::1/64                                  ether1       no
 3 DL fe80::219:d1ff:fe39:3535/64                     ether1       no
```

Notice that our added address has G flag indicated that this address can be globally routed. We also have link local address on the interface which is created automatically for every IPv6 capable interface.

Test connectivity

```
[admin@R1] /ipv6 address> /ping 2001:DB8::2
HOST                                 SIZE TTL TIME  STATUS
2001:db8::2                      56  64 12ms  echo reply
2001:db8::2                      56  64 0ms   echo reply
    sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=6ms max-rtt=12ms
```

[Top | Back to Content]

Retrieved from "https://wiki.mikrotik.com/index.php?title=Manual:IPv6/Address&oldid=34073"

Categories: Manual │ IPv6

- This page was last edited on 18 June 2020, at 12:47.