IPv6 Address Space

Best Practice Document

Produced by the CESNET-led working group on IPv6 (CBPD116)

Author:
Tomáš Podermański

August 2011

Document No: GN3-NA3-T4-CBPD116
Version / date: 1 August 2011
Original language: Czech
Original title: “IPv6 Mýty a skutečnost, díl II. - Adresový prostor”
Original version / date: 1 of 1 August 2011
Contact: tpoder@cis.vutbr.cz

This translated version is based on the Czech version published in the electronic journal, Lupa.cz, on 17 February 2011.

The translation of this report has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 238875, relating to the project 'Multi-Gigabit European Research and Education Network and Associated Services (GN3)'.

Table of Contents

Executive Summary

This document describes network structure, the ways of creating IPv6 addresses in end-user networks, and the methods used to connect home, corporate and campus networks.

1 IPv6 address scheme

The primary motivation for creating a new generation of the Internet Protocol was expansion of the address space. IPv6 offers a significantly longer length of addresses. Compared to IPv4, which, in theory, can provide addresses for around four billion devices, the IPv6 protocol features an address length of 128 bits. This means that up to 3.4 x 10^{38} devices can be connected to the Internet. Such a number is hard to imagine. Several trillion addresses could be assigned to each person on the earth.

Another attempt to imagine the total address space would be as follows. If one were to assign one network with a length of 40 bits every second, this stockpile would last for 35 thousand years (it would last for 136 years with a prefix length of 32 bits). Hence, the total number of addresses can be considered to be truly unlimited.

An IPv6 address is written as 32 hexadecimal numbers, divided into quadruplets, separated with a colon. The following simplifications can be used when writing the address. Variants are shown, using as an example the address 2001:067c:1220:0004:0000:0000:93e5:0394.

Omit the introductory zeroes in each quadruplet	2001:67c:1220:4:0:0:93e5:394
Omit the longest sequence of 0 and replace it with a the symbol ::	2001:067c:1220:4::93e5:394
Record the last 32 bits using the IPv4 notation	2001:067c:1220:4::147.229.3.148

All of the above formatting can be used to express the same address. There is also a format that uses number-pairs, separated with colons, i.e., 20:01:06:7c:12:20:00:04:00:00:00:00:93:e5:03:94, but this format is fairly rare, and is only used, for example, in Management Information Base (MIB) tree definitions.

The prefix length is another typical parameter that is specified in an IPv6 address. The equivalent of the prefix in an IPv4 network is a network mask. In an IPv6 network, the bit field format, which defines the part of the address that belongs to the network and host part, is not used; only the notation that defines the network mask length is used. This is recorded behind the "/" symbol by specifying the number of bits in the network part of the address. The resulting address looks like this:

2001:718:802:4:250:56ff:feba:4a85/64 2001:718:802:4:250:56ff:feba:4a85/54
2001:718:802:4:250:56ff:feba:4a85/128
fe80::c62c:3ff:fe36:4f4d/64

2 Address types

Like IPv4, IPv6 defines several address types[1]. The following table summarises the basic types of IPv6 addresses currently in use, together with their equivalents from IPv4.

Prefix	Name	Meaning	IPv4 equivalent
::1/128	Loopback	loop	127.0.0.1/8
fe80::/10	Link-Local	local addresses	does not exist
fec0::/10	Site-Local	cancelled	10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16
fc00::/7	Unique-Local	unique private addresses	
ff00::/8	Multicast	group calls	224.0.0.0/4
does not exist	Broadcast	broadcast	255.255.255.255
2000::/3	Global	global addresses	global IPv4 addresses

Some IPv4 address types were cancelled without replacement, while other new types have appeared.

2.1 Link-Local

Link-Local addresses are a revolutionary innovation. This is a special address type, which is in the configuration of each IPv6 device as soon as the interface comes up. Link-Local address cannot be cancelled or changed on some devices. Packets that use the Link-Local address can only spread within one network segment, i.e., up to the first router.

Link-Local addresses can easily be recognised by their unique combination *fe80::*. A full Link-Local address appears as:

[1] http://www.iana.org/assignments/ipv6-address-space/ipv6-address-space.xml

```
fe80::20c:29ff:fec9:92df/64
```

A special feature of these addresses is that an address on one device often has the same value on various interfaces. Usually one must add a name or some interface identifier right after the % symbol to identify the interface that one wants to communicate with.

The existence of Link-Local addresses enables moving a large part of the communication to this level, especially in the signalling field. For example:

- **All signalisation at the local level: neighbour discovery (ND), router advertisement (RA), detection of duplicate addresses etc.** For example, IPv6 does not use the Address Resolution Protocol (ARP). All functions of this protocol were moved to ICMPv6 (Internet Control Message Protocol version 6) signalling messages, which use only the Link-Local address.
- **Routing protocols such as IPv6 Dynamic Routing and Redistribution (OSPFv3/RIP-NG).** A positive result is the disappearance of the need to configure the connecting network between individual router interfaces. It is only necessary to assign the relevant router interface to the OSPFv3 or RIP-NG process and that completes the configuration of the connecting network between routers.

However, Link-Local addresses can sometimes be used for communication between nodes. Because Link-Local addresses can only reach the local network, no special security measures would seem to be required. However, it is possible to communicate in any network through Link-Local addresses; hence it would be a mistake to enter the following record into */etc/host.allow*:

```
ALL : [FE80::]/64 : allow
```

If this occurs, the user's services would inadvertently be offered to all neighbours, in all of the networks where the user appears, e.g., to all the 'bad guys' connected to the same WiFi network. Communication on Link-Local addresses should also be blocked at the corporate-firewall level. No one could communicate through these addresses, but an attacker could use a good, counterfeit Link-Local address, entered as a source address in a packet. In addition, the end system should automatically discard all packets that have the Link-Local address in its source and that have the destination address set as *global*.

2.2 Broadcast

There are no broadcast addresses in IPv6. All communication that used broadcast in IPv4 has been moved to multicast, without exception.

2.3 Unique-Local, Site-Local

Another group of addresses are the IPv6 addresses that can be compared to the private addresses in IPv4. In IPv6, the equivalent addresses are Unique Local IPv6 Unicast Addresses (ULA)[2], defined in RFC 4193. Like private IPv4 addresses, they can only be used within a limited location and must not be routed to the global Internet. They have one key advantage, compared to private IPv4 addresses. The algorithm to create a

[2] http://www.ietf.org/rfc/rfc4193.txt

network prefix is designed to be unique. A prefix is created through an algorithm, into which a date and the MAC address of a device are entered. This should ensure that the prefix created is unique. Those who still doubt the uniqueness of their ULA addresses can use a registration obtained from a database managed within the SixXS[3] website.

On some systems, and even in recent literature, one can still find Site-Local addresses. In contrast to ULA, they are not unique, and resemble the private addresses of the IPv4 protocol. Using these addresses proved to be a dead-end, and Site-Local addresses were abolished by RFC 3879[4] in 2004. The address block *fec0::/10* assigned to these addresses should no longer be used.

2.4 Multicast, Anycast

Group (multicast) and selection (anycast) addresses will be described, in detail, in a separate document, and only their existence is mentioned in this report.

2.5 Global

The one remaining address type are Global addresses. These are the most important addresses and global Internet nodes communicate through them.

[3] http://www.sixxs.net/tools/grh/ula/
[4] http://www.ietf.org/rfc/rfc3879.txt

3 The network part of global addresses and network structure

There was an effort to divide addresses in the global Internet, so that, if read from left to right, they would reflect the physical structure of the network. The highest level has address blocks managed by IANA[5]. It assigns blocks to individual regional registries (RIR), i.e., RIPE NCC, ARIN, APNIC, AFRINIC, and LACNIC. Regional registries then assign these blocks to local registries (LIR), which connect the end-user networks. Each of these networks then receives its own prefix, which can be divided further. It was assumed that each end customer (user) would get an assigned network with a prefix length of 48 bits, leaving sixteen bits for internal addressing. In practice, with a prefix length of 64 bits the customer can create 65,535 end networks. There is still an on-going discussion about whether the assignment of a prefix of 48 bits to normal users is a waste, and hence, the current draft[6] sets this length to 56 bits. As a result, 256 subnets can be created in this way and 264 devices could be addressed in each subnet, which should be a sufficient number for end customers.

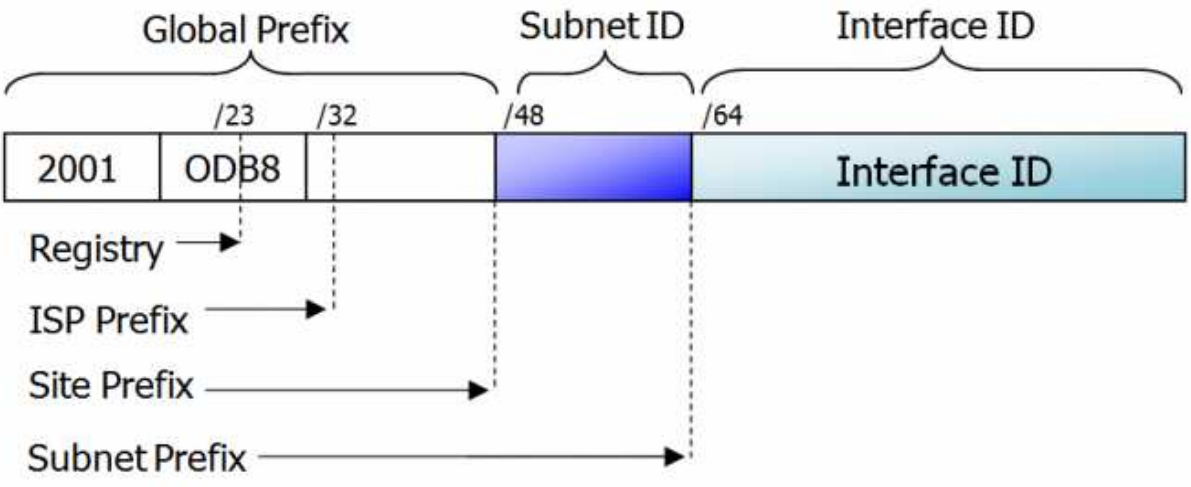

Figure 1: The structure of the IPv6 address

[5] http://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xml
[6] http://tools.ietf.org/html/draft-ietf-v6ops-3177bis-end-sites-01

3.1 Home networks

The assignment of addresses to smaller or home networks is still open to discussion. Currently, a single IPv4 address is assigned to a network and an end device is connected directly to this address, or it can be connected to private addresses through Network Address Translation (NAT). The existence of NAT is unacceptable for IPv6. Therefore, the question remains how to assign addresses to home networks. A box at home might be seen as an L2 switch, L3 switch or even some form of IPv6 NAT.

The fairly simple solution that is used in IPv4 networks, becomes significantly more complicated in the IPv6 world. Providers who have decided to implement IPv6 for customers connected in this way face a significant problem. They could choose one of the following solutions, but none of these is trouble-free. The issue is further complicated by the fact that during the transition period, which will be quite long, the current IPv4 infrastructure with its NAT mechanism will operate in parallel. This problem can be resolved in three ways.

IPv6-NAT: NAT should be completely eliminated in IPv6 networks. There are efforts to introduce NAT to the IPv6 paradigm, but so far, these are only proposals. A working and usable implementation is probably impossible to find. One compromise solution could be the stateless mapping of ULA addresses to public addresses as proposed in an Internet Draft[7]. There is an implementation of this solution is available, in an experimental form, as Linux netfilter[8]. Some modifications would have to be made in order to make it usable in this mode. This solution is the least popular.

A device in the L3 router mode: In this solution, a home router would provide address translation through NAT for IPv4, and it would provide a complete L3 router for IPv6 operation. This raises the questions of how a home router could be informed of the internal network prefix and vice versa, and how the routing record on the Internet Service Provider (ISP) side that would point to networks behind the home router can be created. A special DHCPv6 option with prefix delegation name (RFC 3633) is used for this purpose, and it tells the home device about networks which it should use for internal addressing. The requirements are detailed in an Internet Draft[9], but so far, it is unclear whether manufacturers of home routers and ISPs will accept it. Relying on the ISP to create the routing records also remains an issue. With thousands of users (customers), this may not be a negligible issue, and the mechanism must somehow be automated. The solution suggested in another Internet Draft[10] assumes that the provider's router (PE) will create these records, based on the contents of the communication between a DHCPv6 server and DHCPv6 agent. However, there do not appear to be any devices that would support this option. This is the most complicated form of connecting home networks, but so far, the prevailing opinion is that this is the "clean" and "correct" solution.

A device in the L2 router mode: This solution is based on connecting the provider’s port and the local network at the link level, which involves a bridge or switch. At first glance, this is a simple solution and would not require more standards, but it is not without problems. Since IPv6 will need to coexist with IPv4 for quite some time, this solution is almost out of the question. A variant worth considering is L2 bridging/switching only of those packets that belong to the IPv6 protocol. This variant is easy to use in theory, but so far, there is no device on the market that would provide this functionality. Another significant disadvantage of this solution is that all of the user’s devices are in the same subnet as a neighbouring laptop, coffee machine, or washing machine. This creates new security problems in the network, which are not insolvable. However, at present, there is not much attention given to home devices that would work in this way.

[7] https://tools.ietf.org/html/draft-mrw-nat66-07
[8] http://sourceforge.net/projects/map66/
[9] http://tools.ietf.org/html/draft-ietf-v6ops-ipv6-cpe-router-09
[10] http://tools.ietf.org/html/draft-yeh-dhc-dhcpv6-prefix-pool-opt-02

There do not appear to be any significant variants to the options presented. Unfortunately, there is no standardised and generally usable method to connect home networks at the present time. Even if the manufacturers of home routers want to implement support for IPv6 into their devices today, they are faced with the problem that they do not know how to accomplish this. At this time, it would be quite hard to tell what the IPv6 support declared on the package of this type of equipment actually means, and whether such a device would be obsolete in a year or two. More documentation on some devices is available on the SixXS[11] pages. It is evident that IPv6 support has a different meaning for each producer. Certification with a silver or golden 'IPv6 Ready' logo will not be of much help, because the criteria for IPv6 support for this type of device are not part of the certification. A DHCPv6 certification is required. In current practice, the support for home devices focuses on static IPv6 routing and it may also feature support for migration mechanisms. Only a few devices offer support for the delegation of the DHCPv6 prefix, and this is usually at a very experimental level.

3.2 Multihoming

Another complication with the assignment of network addresses is related to the opposite end of the range of clients of IPv6 networks - large corporations. The original idea to divide addresses hierarchically, according to network topology, is certainly very good. It significantly reduces the number of records in the routing tables due to aggregation of individual prefixes. Multihoming, i.e., connecting the network to multiple providers, breaks this concept. In the IPv4 protocol, full-featured multihoming is almost entirely solved by the assignment of a dedicated autonomous system (AS) and the allocation of a special address space (PI – provider independent address[12]). Routers that connect the network to multiple providers through the Border Gateway Protocol (BGP) then provide connection to the routing of the global Internet.

Creators of the IPv6 protocol tried to solve the multihoming problem with a totally different concept, in order to maintain the hierarchical network structure. The Shim6 protocol (Site Multihoming by IPv6 Intermediation) provides some hope, but it is only a proposal at this stage and has only one, mostly experimental, implementation[13]. It will certainly be a long time before this protocol can be used in practice.

Since no acceptable and working solutions have been found so far, it was inevitable to try out an already tested mechanism in the form of allocation of provider-independent (PI) addresses. However, each PI network generates a separate routing record on the level of the global routing table in BGP, and this breaks the original concept of optimised routing. The multihoming question will be addressed in a separate document.

[11] http://www.sixxs.net/wiki/Routers
[12] http://www.ripe.net/ripe/docs/ripe-509#9
[13] http://www.openhip.org/

4 End-user networks

The remaining part of the address is designed to identify individual devices in end-user networks. Usually, and almost without exception, networks with a prefix of 64 bits are assigned. This divides the IPv6 address into two parts: the network part in the most significant 64 bits and the host part (Host ID or Interface ID) in the least significant 64 bits. Some may consider such division as quite wasteful. To have 64 bits for addressing within a single network segment is certainly an indulgence, but this division has already become ingrained and any attempts to use terminal network prefixes, other than 64 bit, would face a whole host of practical complications. Unless there is an extremely good reason, prefixes longer than 64 bits should not be used, even if there are only two or three devices in the network.

4.1 The host part of the address – Host ID, Interface ID

In the past, at the time of Internetwork IPX/SPX networks, it was not necessary to burden oneself with the addressing mechanism of terminal stations. A card was inserted into a PC and the PC was connected to the network. This was the entire configuration. The network layer address was derived from the network-card link-address and that was all that had to be done. Of course, this simple solution was impossible for IPv4 networks, because of the relatively small space for addressing the devices in the end-user networks. The possibilities are completely different with the IPv6 protocol. The space reserved for the host part is 64 bits. Such abundant address space enables things not seen before.

4.2 IPv6 addresses EUI-64

The first proposals assumed that the host part of the address would be derived from "something" that already had an address. Using the existing link-layer address, i.e., the MAC address was the obvious thing to do. Mapping 48 bits (the MAC address length) to 64 bits was no problem. The IPv6 EUI-64 algorithm to create the host part of the address was made through a small modification of the IEEE EUI-64 algorithm[14].

[14] http://standards.ieee.org/develop/regauth/tut/eui64.pdf

4.3 Mapping the IPv6 EUI-64 addresses

As illustrated in Figure 2, moving forward and backward can be done simply and effortlessly, and the relation between both addresses is obvious at first glance. This very simple and practical way of creating IPv6 addresses began to meet serious problems because of privacy protection. Since the host part of the address does not change if a laptop or cell phone is moved anywhere, it is very easy to discover which specific device has accessed the relevant service and also from which network. This kind of address creation is implemented and activated by default in most operating systems (MAC OS, Linux, and FreeBSD) and devices.

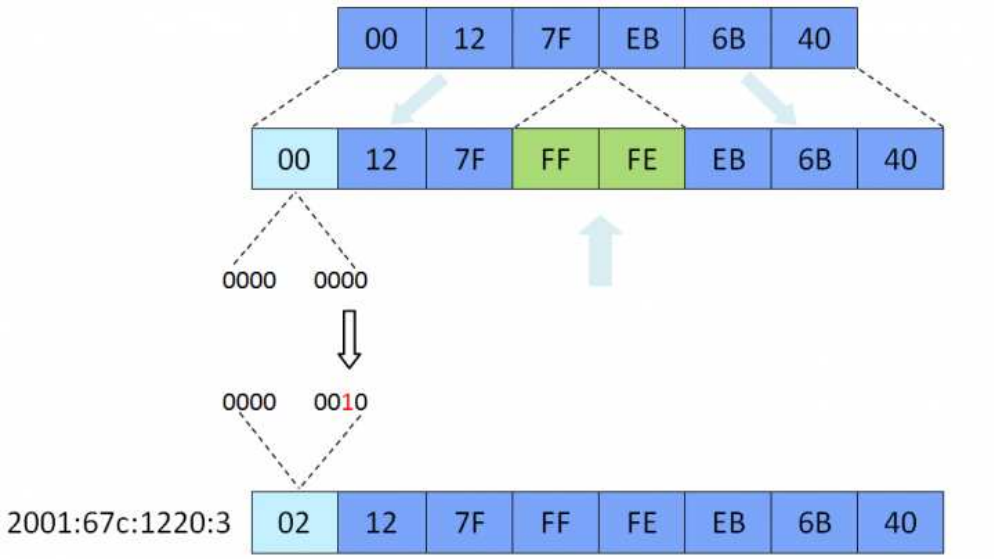

Figure 2: Mapping the IPv6 EUI-64 addresses

4.4 Privacy Extensions

The issue of privacy led to consideration of how it can be avoided that end-user devices are simply identified. A solution called „Privacy Extensions“ (Privacy Extensions for Stateless Address Auto configuration in IPv6) was created, initially as RFC 3041[15] and later replaced by RFC 4941[16]. The Privacy Extension concept is based on random generation of the host part of the IPv6 address. Addresses created in this way change at regular intervals. Typically, a new address is created each day or each week and is maintained in the system for ten days. The IPv6 address of the end-equipment is created randomly, cannot be predicted, and changes continuously – thus, each of these features is a nightmare for network administrators. It is obviously unacceptable for them to have devices in the network that they cannot identify. Privacy Extensions are, by default, activated in all Microsoft end-user systems (Windows 7, Vista, XP). Privacy Extensions can be activated for most other systems (Mac OS, Linux, FreeBSD) in their configuration. Figure 3 shows some IPv6 addresses in Windows XP after a week of non-stop operation.

[15] http://tools.ietf.org/html/rfc3041
[16] http://tools.ietf.org/html/rfc4941

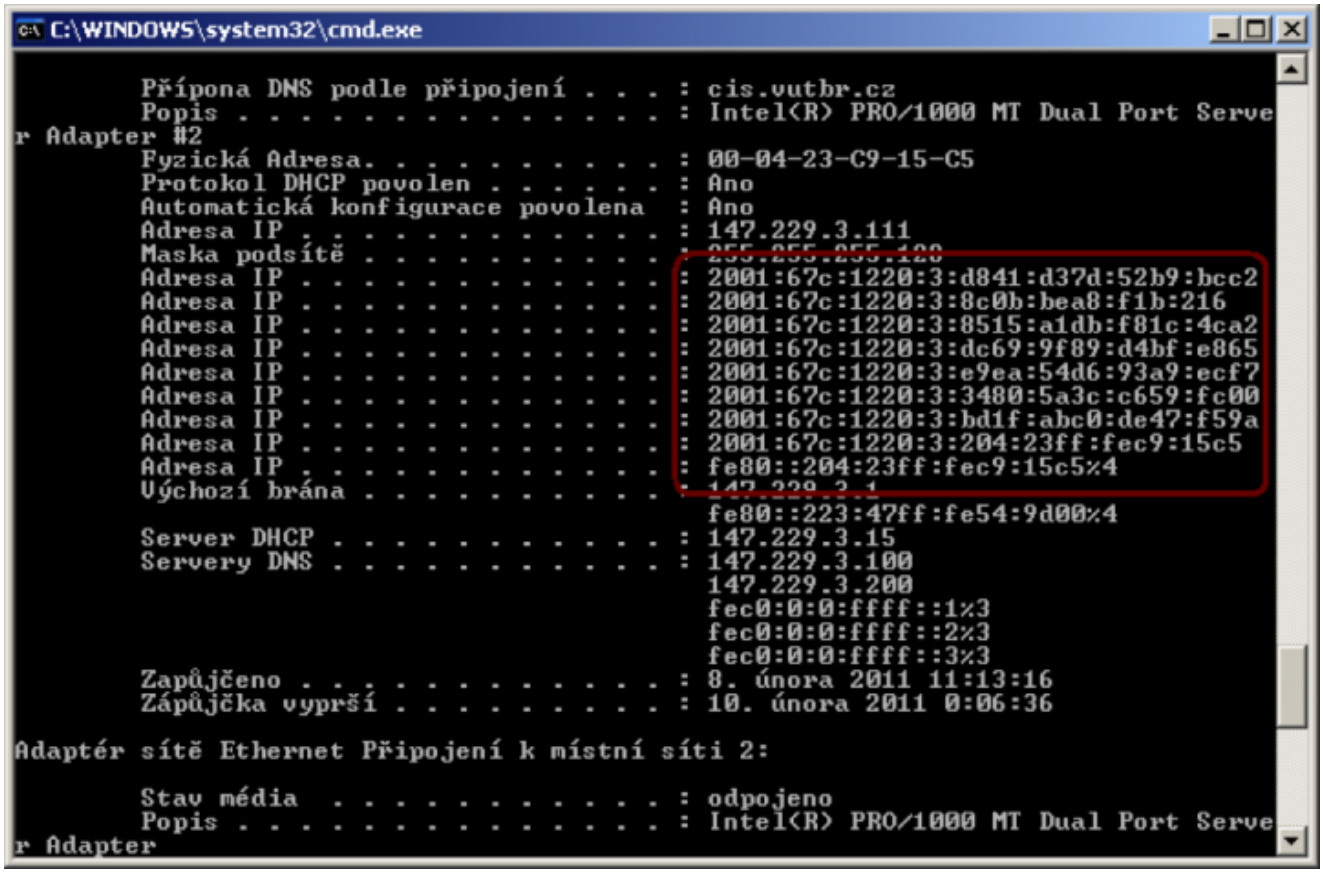

Figure 3: Random IPv6 addresses in Windows XP

Since the identification of terminal devices in the network is fairly critical for administrators to maintain order, it is a problem that must be solved. One possibility is to deactivate Privacy Extensions on client systems directly, but this abridges the users of their privacy, and reconfiguration of all devices that appear in the network is simply not feasible. With time, this extension will probably be activated by default in most systems, following the Microsoft example.

One solution would be to collect the content of Neighbour Cache tables on the router. This is similar to the ARP table from IPv4. Just like the ARP, the Neighbour Cache contains the link address (MAC address) and its IPv6 address or addresses. If the IPv6 – MAC address is known, some conclusions can be derived from their relation. For instance, if the network supports authentication through IEEE 802.1X, this data can be connected with the authentication data of the RADIUS server. The following chart shows the content of Neighbour Cache received in this way, and illustrates how the PC communicates with different IPv6 addresses in specific time periods during one week.

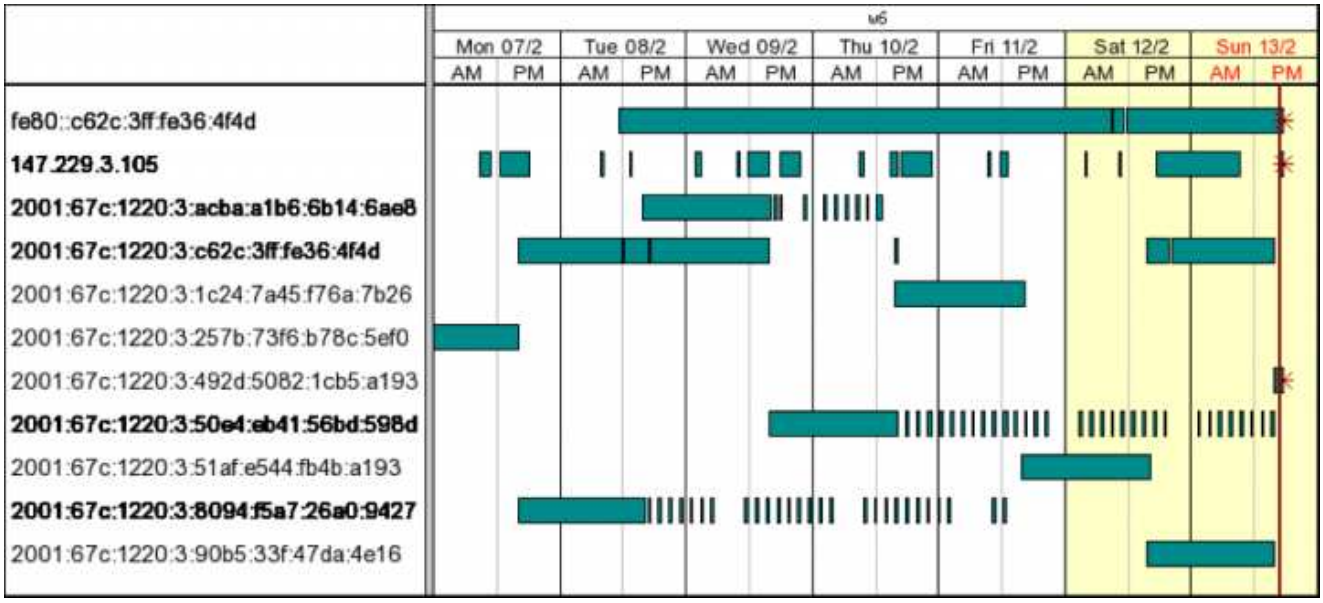

Figure 4: Temporary IPv6 addresses in the system within one week

So far, there are not many tools that can deal with this problem. One tool that can be used is MetaNAV[17]. It reads the contents of router Neighbour Caches regularly with the Simple Network Management Protocol (SNMP) and saves the data in a database. This database can subsequently be connected with another internal system. The web interface offered by MetaNAV can also be used.

[17] http://metanav.uninett.no/

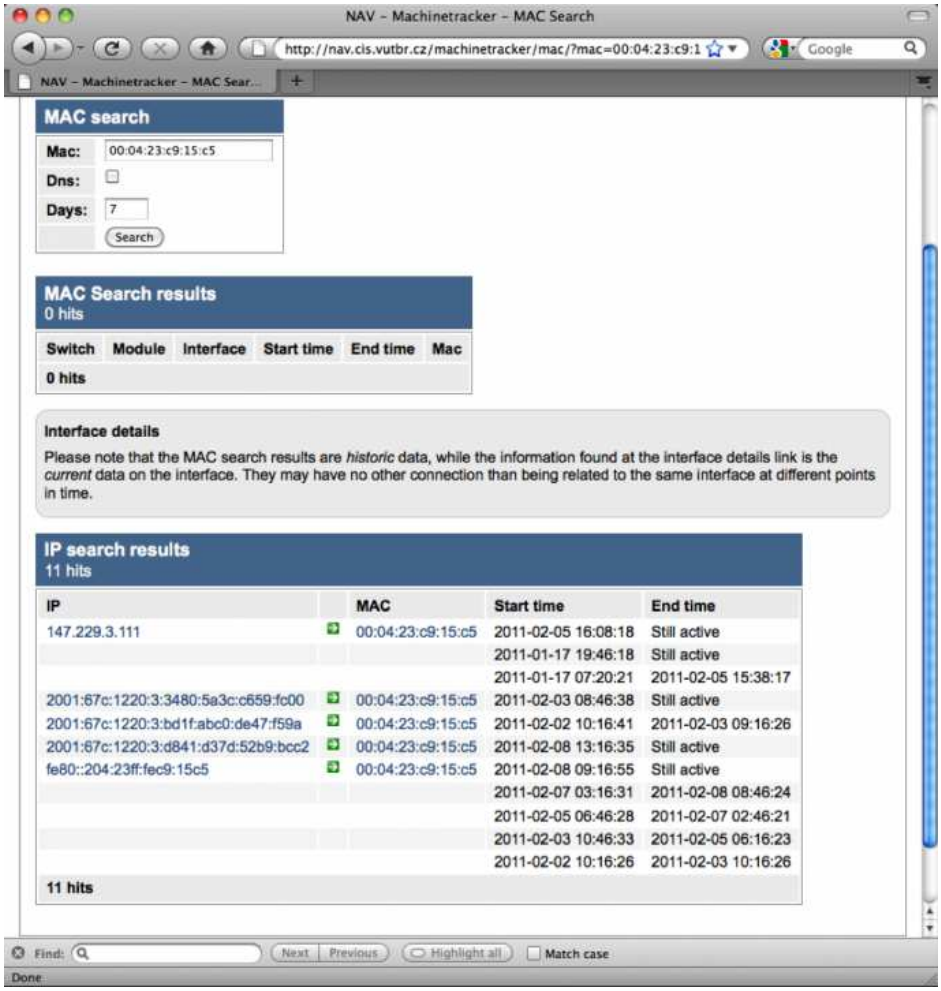

Figure 5: A look at IPv6 addresses in MetaNAV

Another option is to use a tool that can analyse the contents of signalling protocols. Since all signalling takes place at the multicast level, it is easy to acquire this information through any device connected to the local network. The tool favoured by many in the IPv4 world is arpwatch[18]. One alternative for use in IPv6 is NDPMon[19]. However, the development of this project has been stagnating since 2009, and the system is burdened with a large number of errors. Nonetheless, it is usable with some effort.

Network monitoring is another issue in IPv6. Some practical ideas to perform IPv6 network monitoring are described in a separate best practice document, CBPD 132 “Practical IPv6 Monitoring on Campus”[20].

4.5 Manual IPv6 configuration and other options

The options for IPv6 address configuration described above are mainly suitable for client stations. The option to use manual configuration remains the same as with IPv4. This option is used, in particular, for servers. For

[18] http://ee.lbl.gov/
[19] http://ndpmon.sourceforge.net/
[20] http://www.terena.org/activities/campus-bp/pdf/gn3-na3-t4-cbpd132.pdf

these, a simplified definition of the IPv6 address can be used by writing the last 32 bits in the form of an IPv4 address. This also empowers creative administrators. They can embed a useful note, message, or alert in the host part of the IPv6 address of their favourite colleague or boss for others to see. Below are a few inspiring examples showing that the IPv6 system is entertaining and *fe80::c00l*:

```
2001:0fa:fff0:fe00:dead:face:c156:7b93
2001:15c0:6603:add:bad:dad:c0b0:1
2001:15c0:6603:0:fade:b00b:babe:1
2001:968::c0f:fee
```

So far, the options of address configuration through DHCPv6 or using a cryptographic method via SeND have not been mentioned. These options will be addressed in a separate report, focused on the autoconfiguration of terminal equipment and safety.

5 Conclusion

The initial motivation to create the IPv6 protocol was the need to extend the address space. At this moment, it can be said that this requirement has been fulfilled. The hierarchisation of addresses has the potential to enable much more effective management of routing information at a global level. This concept has been somewhat broken by the concept of PI address allocation. Sadly, the mechanism of address allocation to home networks remains unstable. This is not so much an issue of the address space itself; rather, it is an issue of the gradual stabilisation of standards and technologies.

At the level of the end-user network, IPv6 offers some completely new possibilities for creating addresses in end-user systems. The creation of IPv6 addresses in client systems with Privacy Extensions is the most significant change. It can be expected that, following the Microsoft example, this extension will be used by most client systems. This innovation has complicated the management of local networks, and it will probably change the way we look at network management and network-management habits significantly**.**

6 List of Figures