The NAROS Approach for IPv6 Multihoming with Traffic Engineering

Cédric de Launois *, Olivier Bonaventure, and Marc Lobelle

Université catholique de Louvain
Department of Computing Science and Engineering
http://www.info.ucl.ac.be
{delaunois,bonaventure,ml}@info.ucl.ac.be

Abstract. Once multihomed, an IPv6 site usually wants to engineer its interdomain traffic. We propose that IPv6 multihomed hosts inquire a so called "Name, Address and ROute System" (NAROS) to determine the source and destination addresses to use to contact a destination node. By selecting these addresses, the NAROS server roughly determines the routing. It thereby provides features like traffic engineering and fault tolerance, without transmitting any BGP advertisement and without impacting on the worldwide routing table size. The performance of the NAROS server is evaluated by using trace-driven simulations. We show that the the load on the NAROS server is reasonable and that we can obtain good load-balancing performances.

Key words: multihoming, traffic engineering, IPv6, BGP.

1 Introduction

The size of BGP routing tables in the Internet has been growing dramatically during the last years. The current size of those tables creates operational issues for some Internet Service Providers and several experts [1] are concerned about the increasing risk of instability of BGP.

Part of the growth of the BGP routing tables [2] is due to the fact that, for economical and technical reasons, many ISPs and corporate networks wish to be connected via at least two providers to the Internet. Nowadays, at least 60% of those domains are connected to two or more providers [3].

Once multihomed, a domain will usually want to engineer its interdomain traffic to reduce its costs. Unfortunately, the available interdomain traffic engineering techniques [4] are currently based on the manipulation of BGP attributes which contributes to the growth and the instability of the BGP routing tables.

It can be expected that IPv6 sites will continue to be multihomed and will also need to engineer their interdomain traffic. Although several solutions to the IPv6 multihoming problem have been discussed within the IETF [5–11], few

* Supported by a grant from FRIA (Fonds pour la Formation à la recherche dans l'Industrie et dans l'Agriculture, rue d'Egmont 5 - 1000 Bruxelles, Belgium).

have addressed the need for interdomain traffic engineering. We propose and evaluate in this paper an innovative host-centric solution to the IPv6 multihoming problem. This solution allows sites to engineer their incoming and outgoing interdomain traffic without any manipulation of BGP messages.

In the following section, we briefly present the technical and economical reasons for multihoming in the Internet, and situate our solution among other proposed multihoming solutions. Next, we describe the NAROS architecture and explain how it supports multihoming and traffic engineering. Finally, we use trace-driven simulations to evaluate the performance of our solution.

2 Multihoming Issues

IPv6 multihoming solutions are significantly different from IPv4 ones because they must allow the routing system to scale better. Morever, the IPv6 address space is much larger, which gives more freedom when designing multihoming. An IPv6 host may have several global addresses. Paradoxically this can help in reducing the BGP table sizes but it requires that hosts correctly handle multiple addresses. Requirements for IPv6 multihoming are stronger and multiple [12]. In this paper, we essentially focus on the following requirements.

Fault Tolerance. Sites connect to several providers mainly to get fault tolerance. A multihoming solution should be able to insulate the site from both link and ISP failure.

Route Aggregation. Every IPv6 multihoming solution is required to allow route aggregation at the level of their providers [1], [12]. This is essential for the scalability of the interdomain routing system.

Source Address Selection. A multihomed IPv6 host may have several addresses, assigned by different providers. When selecting the source address of a packet to be sent, a host could in theory pick any of these addresses. However, for security reasons, most providers refuse to convey packets with source addresses outside their address range. So, the source address selected by a host also determines the upstream provider used to convey the packet. This has a direct impact on the flow of traffic. Moreover, if a host selects a source address belonging to a failed provider, the packet will never reach its destination. Thus, a mechanism must be used to select the most appropriate source address.

Destination Address Selection. When a remote host contacts a multihomed host, it must determine which destination address to use. The destination address also determines the provider used. If a provider of the multihomed site is not available, the corresponding destination address cannot be used to reach the host. So we must make sure that an appropriate destination address is always selected.

Traffic Engineering. A multihomed site should be able to control the amount of inbound and outbound traffic exchanged with its providers.

ISP Independence. It is desirable that a multihoming solution can be set up independently without requiring cooperation of the providers.

2.1 Related Work

All current IPv6 multihoming approaches allow route aggregation and provide at least link fault tolerance. A summary of desired features provided by various multihoming solutions is provided in table 1. The solutions and their features are detailed in a survey on multihoming mechanisms [5].

Table 1. Features provided by current IPv6 multihoming solutions

Feature	*[8]*	*[9]*	*[7]*	*[10]*	*[6]*	*[13]*	*[11]*	*NAROS*
Link fault tolerance	x	x	x	x	x	x	x	x
ISP fault tolerance			x	x	x	x	x	x
Stable configuration in case of long term failure			x	x		x	x	x
Explicit ISP selection					x	x		x
Allows load sharing	x	x			x	x	x	x
Explicit traffic engineering								x
Solve source address selection problem							x	x
Transport-layer survivability	x	x		x	x			
Site-ISP independency			x	x	x	x	x	x
Inter ISP independency		x	x	x	x	x	x	x
No changes for Internet routers	x	x	x	x	x	x	x	x
No changes for site exit routers	x	x	x	x	x			x
No changes for hosts	x	x	x					
No changes for correspondent nodes	x	x	x		x	x	x	x
No new security issues	x	x	x			x	x	x
No need of tunnels			x	x	x	x		x
No modification to current protocols	x	x	x		x			x
No new protocol	x	x	x	x	x	x	x	
Valid for both TCP and UDP	x	x	x		x	x	x	x

The first two approaches [8], [9] use tunnels and/or backup links with or between the providers. The third solution [7] uses the Router Renumbering [14] and Neighbor Discovery [15] protocols to deprecate addresses in case of ISP failure. The fourth approach [10], proposes to modify the TCP protocol to preserve active TCP connections. The fifth solution uses the IP mobility mechanisms to switch between delegated addresses in case of failure [7], [6]. The sixth approach [13] consists in enhancing the Neighbor Discovery protocol to help the hosts in selecting the appropriate site exit routers. The solution proposed in [11] defines new ICMP redirection messages to inform a host of the site exit router to use.

The last approach is the NAROS approach presented in this paper. It relies on the utilization of several IPv6 addresses per host, one from each provider. The basic principle of NAROS is that before transmitting packets, hosts contact the NAROS service to determine which IPv6 source address they should use to reach a given destination.

This approach has never been developped, although briefly suggested in [11]. To the best of our knowledge, this is the first approach that explicitly allows load-balancing and traffic engineering in IPv6 multihoming sites.

3 The NAROS Service

Figure 1 illustrates a standard multihomed site. Suppose three Internet Service Providers (ISPA, ISPB and ISPC) provide connectivity to the multihomed site. The site exit router connecting with ISPA (resp. ISPB and ISPC) is RA (resp. RB and RC). Each ISP assigns a site prefix. The prefixes (PA, PB and PC), together with a subnet ID (SA, SB or SC) are advertised by the site exit routers and used to derive one IPv6 address per provider for each host interface.

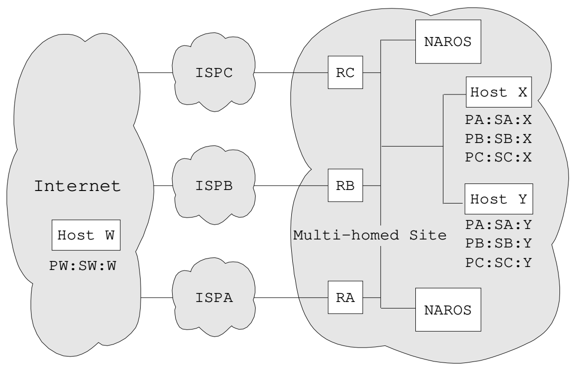

Fig. 1. A multihomed site connected with three providers

In the NAROS architecture, the site advertises ISPA addresses only to ISPA, and ISPA only announces its own IPv6 aggregate to the global Internet.

Since each host has several IPv6 addresses, it must decide which address to use when transmitting packets. The basic principle of our solution is to let the NAROS service manage the selection of the source addresses. This address selection will influence how the traffic flows through the upstream providers and a good selection method will allow the site to engineer its interdomain traffic.

We now consider in details how NAROS addresses four main issues : source and destination address selection, fault-tolerance and traffic engineering.

Source Address Selection. When a host initiates a connection with a correspondent node, it must determine the best source address to use among its available addresses. The source address selection algorithm described in [16] already provides a way to select an appropriate address. However, this selection is arbitrary when a host has several global-scope IPv6 addresses as in our case.

The principle we propose is that the host asks the NAROS service which source address to use. It complements in this way the default IPv6 source address selection algorithm [16].

Many factors could possibly influence the selection process, such as the current loads and states of the links or administrative policies. A NAROS server could also rely on informations contained in BGP tables, e.g. the path length towards the destination.

In its simplest form, the basic NAROS service is independent from any other service. A NAROS server does not maintain state about the internal hosts. It is thus possible to deploy several NAROS servers in anycast mode inside a site for redundancy or load-balancing reasons. A NAROS server can also be installed on routers such as the site exit routers. The NAROS protocol runs over UDP and contains only two messages : NAROS request and NAROS response [17].

The first message is used by a client to request its connection parameters. The parameters included in a NAROS request are at least the destination address of the correspondent node and the source addresses currently allocated to the client. The NAROS server should only be contacted when the default source address selection procedure [16] cannot select the source address.

The NAROS response message is sent by a NAROS server and contains the connection parameters to be used by the client. The parameters include at least the selected best source address, a prefix and a lifetime. It tells that the client can use the selected source address to contact any destination address matching the prefix. These parameters remain valid and can be cached by the client during the announced lifetime.

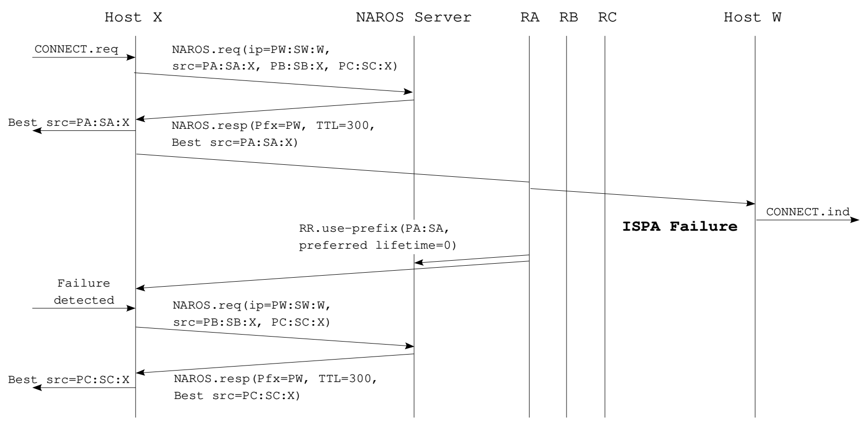

Fig. 2. Basic NAROS scenario example

The upper part of figure 2 shows an example of how the NAROS messages and parameters are used. The exact format of the NAROS message is outside the scope of this paper. When Host X sends its first packet to remote Host W (PW:SW:W), it issues a NAROS request in order to obtain the source address to use to reach Host W. Upon receipt of the request, the NAROS server identifies the prefix PW associated with Host W and selects for example PA:SA:X as the best source address. The prefix can be determined arbitrarily, e.g. using the /8 prefix corresponding to the destination address. Another solution is to extract from a BGP table the prefix associated with the destination. The server then indicates the lifetime (e.g. 300 seconds) of these parameters in the NAROS response message.

After having processed the reply, Host X knows that it can use PA:SA:X to reach any destination inside prefix PW, including Host W. The selected source address should be used for the whole duration of the flow, in order to preserve the connection. If new TCP or UDP connections for the same destination are initiated before the announced lifetime expires, the client can use the cached parameter. Otherwise the host must issue a new NAROS request and it may get a different source address for the same destination. By using appropriate values for the lifetime and the prefix in the NAROS response, it is possible to reduce the number of NAROS requests sent by hosts as will be shown in section 4.

Destination Address Selection. A second case is when Host W on the Internet needs to contact Host X in the multihomed site. It first issues a DNS request. The DNS server of the multihomed site could reply with all the addresses associated to Host X. At worst, Host W will try the proposed addresses one by one. Eventually, a connection will work.

Fault Tolerance. A third problem to consider is when one of the upstream providers fails. As in the solution described in [7], [11], the site exit routers use router advertisement messages to communicate to hosts the available prefixes [15]. When a provider crashes, the site exit router connected to this provider detects the event and advertises a null preferred lifetime for that prefix. A client can take this event into account by immediately asking new parameters to the NAROS server. More generally, a host can ask updated parameters each time it detects a failure which affects one of its communications. Once the new source address is known, IP mobility or other mechanisms can be used in order to preserve the established TCP connections [6], [10].

In the lower part of fig. 2, consider for example that ISPA becomes unavailable. The site exit router connected to ISPA detects the failure and advertises a null preferred lifetime for prefix PA. The NAROS server immediately takes this advertisement into account and future NAROS replies will not contain this prefix. Host X will also receive this advertisement. The standard effect is that it should no longer use this source address for new TCP or UDP flows. If Host X is currently using a deprecated address, it can issue a new NAROS request to choose among its other available source addresses. The host can then use IP mobility mechanisms to switch to the new source address in order to maintain its connection alive.

Traffic Engineering. When a host selects a source address, it also selects the provider through which the packets will be sent. Since the source address to use is selected by NAROS, this can naturally be used to perform traffic engineering. For example, in order to balance the traffic among the three providers in figure 1, a NAROS server can use a round-robin approach. For each new NAROS request, the server selects another provider and replies with the corresponding source address. Except when a provider fails, this source address, and thus the upstream provider, remains the same for the whole duration of the flow.

NAROS Advantages Beside the above functionalities, the NAROS approach has several advantages. First, the NAROS service can be set up independently from the providers. A provider only delegates a prefix to the site. This makes the solution applicable for small sites such as enterprise networks. Next, since routes to addresses delegated by one provider are not announced to other providers, full route aggregation is possible. Another advantage is that the solution allows traffic engineering without injecting any information in the internet routing system. Moreover, the NAROS service can easily support unequal load distribution, without any additionnal complexity. Next, NAROS allows the providers to perform ingress filtering, which benefits to security. Finally, changes are limited to hosts inside the multihomed site. Legacy hosts are still able to work, but they cannot benefit from all the NAROS advantages.

4 Performance Evaluations

The NAROS protocol depends on choosing two base parameters : the size of the prefix associated with the destination and its lifetime. We now evaluate the impact of these parameters on the cache size of the hosts, the number of NAROS requests and consequently the server load, and finally the load-balancing quality.

The evaluation of the NAROS service presented in this section is based on a real traffic trace [18]. This trace is a flow-trace collected during 24 hours on November 18, 2002 and contains all the interdomain traffic of a university site. 7687 hosts were active in the network and the volume of traffic exchanged is about 200 GB (18.8 Mb/s in average). The trace contains information about 322 million packets forming more than 17.5 million TCP and UDP flows. The average flow lifetime is 12 seconds. We evaluated the NAROS protocol with IPv4 because no significant IPv6 network is available today.

The first performance parameter to consider is the size of the NAROS cache maintained by the hosts. We evaluate the impact of the prefix length used in the NAROS replies on the cache size of the hosts. For example, if a host requests for destination 1.2.3.4, the NAROS may reply with a /24 prefix, meaning that the parameters are valid for all addresses in 1.2.3.0/24. It may also extract the corresponding prefix from a BGP table. In this case, the prefix length is variable because it depends on the prefix matched in the BGP table for this destination.

Figure 3 shows on a log-log scale $p_1(x)$: the percentage of hosts having a maximum cache size greater than x. It shows for example that if we use /24 prefixes as in the example, the cache size remained below 100 entries during the whole day for 95% of the hosts. We used a lifetime of 300s. The hosts which present the largest cache size were found to be either compromized machines sending lots of probes or very active peer-to-peer clients.

The use of lower lifetimes (not shown) yields to smaller cache sizes. A consequence of this figure is that small prefix lengths and low lifetime contribute to small cache sizes. A value of 300s seems appropriate for the studied site.

We also evaluate the impact of the lifetime on the cache performance. A good cache performance is necessary to limit the number of NAROS requests that a

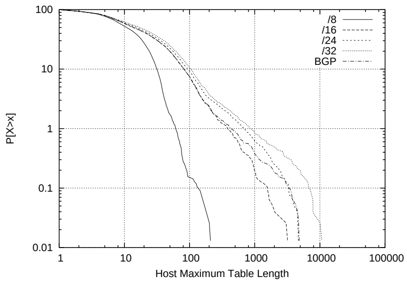

Fig. 3. NAROS Cache size for a lifetime of 300s and various prefix lengths

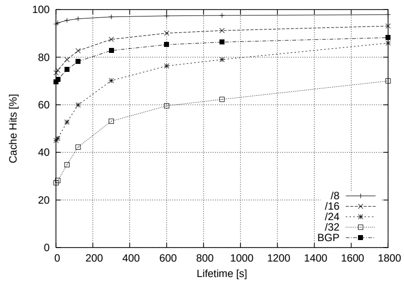

Fig. 4. Impact of the response lifetime on the cache performance

host issues. Figure 4 evaluates the percentage of cache hits versus the lifetime in seconds. It shows that the cache hit ratio is higher when longer lifetime or smaller prefix lengths are used. However, we get no significant improvement by using lifetimes longer than about 300 seconds. We also see that the lifetime has little impact on the cache hit ratio when /8, /16 or BGP prefixes are used.

A second element to consider is the server load. Figure 5 shows on a log-log scale $p_2(x)$: the percentage of hosts issuing more NAROS requests than x, during the whole day. We use here a lifetime of 300s and simulate various prefix lengths. Figure 5 shows that when BGP prefixes are used, 90% of the hosts issue less than about 300 requests during the whole day. The resulting server load is illustrated in figure 6. This load is proportional to the number of host and essentially follows the traffic load. The load average is about 35 requests per second, which is still reasonable. In comparison, this is no more than the number of DNS requests coming from the Internet and addressed to the site studied. The bandwidth overhead of the NAROS approach is evaluated to about 0.35%.

We now compare the performance of the NAROS load-balancing technique with the best widely used load-balancing technique which preserves packet ordering, i.e. CRC16 [19]. We focus on the common case of load-balancing between two outgoing links of the same capacity. For the NAROS load-balancing, we use a round-robin approach, i.e. a new flow is alternatively assigned to the first and the second links. CRC16 is a direct hashing-based scheme for load-balancing where the traffic splitter uses the 16-bit Cyclic Redundant Checksum algorithm as a hash function to determine the outgoing link for every packet. The index of the outgoing link is given by the 16-bit CRC checksum of the tuple (source IP, destination IP, source port, destination port, protocol number), modulo the number of links. CRC16 is often used on parallel links from the same router.

We measure the performance of the load-balancing by looking at the deviation from an even traffic load between the two links. Let $load_1$ and $load_2$ be respectively the traffic load of the first and the second link. We define the deviation as a number in $[-1, 1]$ computed by $(load_1 - load_2)/(load_1 + load_2)$. A null deviation means that the traffic is balanced, while a deviation of 1 or -1 means that all the traffic flows through the first or the second link. Fig. 7 compares the

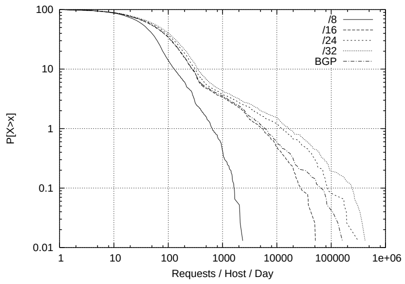

Fig. 5. Number of requests per host during the day, with a lifetime of 300s

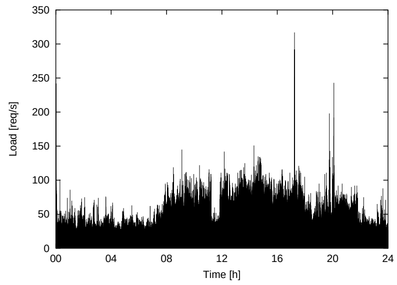

Fig. 6. Server load using BGP and a lifetime of 300s

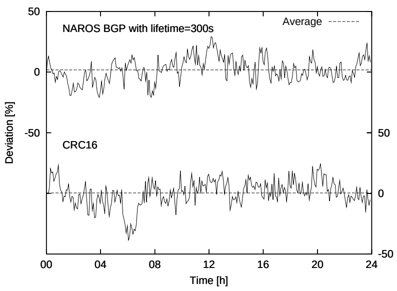

Fig. 7. NAROS and CRC16 load balancing comparison

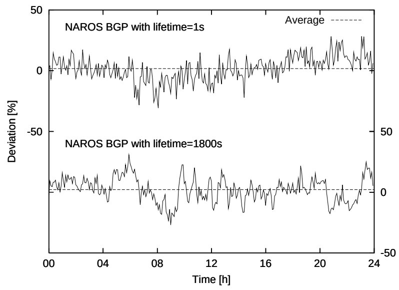

Fig. 8. NAROS load balancing with lifetime of 1s and 1800s

deviation in percent of the NAROS and CRC16 load-balancing. For NAROS, we used BGP prefixes and a lifetime of 300s. We see that the NAROS solution is able to provide load-balancing as good as the best current static load-balancing mechanism. Fig. 8 compares the NAROS load-balancing quality for a lifetime of 1s and a lifetime of 1800s. It shows that the quality of the load-balancing is better when short lifetimes are used, at the expense of a larger server load.

5 Conclusion

In this paper, we have proposed a solution which provides fault-tolerance and traffic engineering capabilities without impacting on the Internet routing tables. When a host needs to communicate with a remote host, it contacts its NAROS server to determine the best source IPv6 address to use. The NAROS server does not maintain any per-host state, can easily be deployed as an anycast service inside each site, and can be set up independently from the providers. It allows to indirectly, but efficiently, engineer the interdomain traffic, without manipulating any BGP attribute. Changes are limited to hosts inside the multihomed site. Legacy hosts are still able to work, even if they cannot benefit from site-multihoming. We have also shown that the load on the NAROS server was reasonable and that, when used to load-balance the outbound traffic between two providers, the NAROS server obtained a similar performance as classical CRC-16 based load-balacing mechanisms. Further investigations include the address selection procedure used by the server and how NAROS can help in engineering the inbound traffic.

Acknowledgements. Steve Uhlig provided the traffic traces and contributed useful comments. We also thank Luca Deri for the `ntop` tool.

References

1. Atkinson, R., Floyd, S.: IAB concerns & recommendations regarding internet research & evolution. Internet Draft, IAB (2003) <draft-iab-research-funding-00.txt>, work in progress.
2. Bu, T., Gao, L., Towsley, D.: On routing table growth. In: Proc. IEEE Global Internet Symposium. (2002)
3. Agarwal, S., Chuah, C.N., Katz, R.H.: OPCA: Robust interdomain policy routing and traffic control. In: Proc. OPENARCH. (2003)
4. Quoitin, B., Uhlig, S., Pelsser, C., Swinnen, L., Bonaventure, O.: Interdomain traffic engineering with BGP. IEEE Communications Magazine (2003)
5. Bagnulo, M., et al.: Survey on proposed IPv6 multi-homing network level mechanisms. Internet Draft (2001) <draft-bagnulo-multi6-survey6-00.txt>.
6. Bagnulo, M., et al.: Application of the MIPv6 protocol to the multi-homing problem. Internet Draft (2003) <draft-bagnulo-multi6-mnm-00.txt>, work in progress.
7. Dupont, F.: Multihomed routing domain issues for IPv6 aggregatable scheme. Internet Draft, IETF (1999) <draft-ietf-ipngwg-multi-isp-00.txt>, work in progress.
8. Jieyun, J.: IPv6 multi-homing with route aggregation. Internet Draft, IETF (1999) <draft-ietf-ipng-ipv6multihome-with-aggr-00.txt>, work in progress.
9. Hagino, J., Snyder, H.: IPv6 multihoming support at site exit routers. RFC 3178, IETF (2001)
10. Tattam, P.: Preserving active TCP sessions on multi-homed networks (2001) http://jazz-1.trumpet.com.au/ipv6-draft/preserve-tcp.txt.
11. Huitema, C., Draves, R.: Host-centric IPv6 multihoming. Internet Draft (2003) <draft-huitema-multi6-hosts-02.txt>, work in progress.
12. Abley, J., Black, B., Gill, V.: Goals for IPv6 site-multihoming architectures. Internet Draft, IETF (2003) <draft-ietf-multi6-multihoming-requirements-04.txt>, work in progress.
13. Draves, R., Hinden, R.: Default router preferences, more-specific routes, and load sharing. Internet Draft, IETF (2002) <draft-ietf-ipv6-router-selection-02.txt>, work in progress.
14. Crawford, M.: Router renumbering for IPv6. RFC 2894, IETF (2000)
15. Narten, T., Nordmark, E., Simpson, W.: Neighbor discovery for IP version 6 (IPv6). RFC 2461, IETF (1998)
16. Draves, R.: Default address selection for internet protocol version 6 (IPv6). RFC 3484, IETF (2003)
17. de Launois, C., Bonaventure, O.: Naros : Host-centric ipv6 multihoming with traffic engineering. Internet Draft (2003) <draft-de-launois-multi6-naros-00.txt>, work in progress.
18. http://www.info.ucl.ac.be/people/delaunoi/naros/ (June 2003)
19. Cao, Z., Wang, Z., Zegura, E.W.: Performance of hashing-based schemes for internet load balancing. In: INFOCOM (1). (2000) 332–341