Available online at www.sciencedirect.com
SCIENCE DIRECT
Computer Communications 29 (2006) 582–592
computer communications
www.elsevier.com/locate/comcom

An incremental approach to IPv6 multihoming

Marcelo Bagnulo[a,*], Alberto García Martínez[a], Arturo Azcorra[a], Cedric de Launois[b]

[a] Universidad Carlos III de Madrid, Av. Universidad, 30, 28911, Leganés, Madrid, Spain
[b] Université Catholique de Louvain (UCL), Place Ste-Barbe 2, B-1348 Louvain-la-Neuve, Belgium

Received 3 June 2005; accepted 3 June 2005
Available online 25 July 2005

Abstract

The availability of two or more connectivity providers (configuration known as multihoming) allows improvements in failure tolerance and enables traffic engineering capabilities. Current IPv4 multihoming solutions suffer from scalability limitations. In this article we present a solution that allow IPv6 networks to benefit from multihoming, taking advantage from the fact that each provider delegates its own set of addresses. The proposed solution consists in multiple mechanisms that provide different benefits to the multihomed site. More precisely, the solution includes a mechanism for the provision of ingress filtering compatibility, a mechanism for establishing new communications after an outage, a set of tools for traffic engineering and a protocol for preserving established communications through outages. In addition we propose a roadmap for the incremental deployment of the set of mechanisms included in the solution based on the trade-off between deployment effort required by each mechanism and the benefits obtained by the involved party.

Keywords: IPv6; Multihoming; Fault tolerance; Communications security

1. Introduction

Over the last decade, sites have massively adopted multihoming (i.e. the connection to the Internet through multiple providers) as a mean to improve the fault tolerance capabilities of their Internet access.

In IPv4, the most popular multihoming configuration for multihoming consists on the injection into the BGP routing system of the prefix or prefixes of the multihomed network [1], as it is shown in Fig. 1. If a failure affects one of the paths in the meshed topology, the routing system will find a valid alternative path, if one is available.

The massive adoption of this solution implies the addition of new entries in the global routing table that account for the prefixes of the multihomed networks. These (more-specific) entries are not needed for single home sites, because reachability is propagated through an aggregated prefix of the provider. The addition of new entries in the global routing table is undesirable, since it results in scalability problems for the BGP route processing that yields to increases in the convergence time [2], among other inconveniences [3].

The IPv6 Internet is designed to host a number of users significantly higher than the IPv4 Internet. For this reason, a cornerstone of the IPv6 design is to preserve the scalability of the routing system through the usage of provider aggregation of addresses. This implies that the number of entries in the routing table is to be limited to those corresponding to the service providers, precluding the adoption of an IPv4-like solution for multihoming. The result is that medium and small IPv6 networks will receive addresses from address blocks assigned to their providers. These large providers exchange routing information through BGP. A medium or small network that seeks for multihoming benefits, i.e. a residential user with two connectivity providers, receives different address blocks from each provider to which it is connected. Therefore, the end systems of this network will be configured with addresses built upon the prefixes of each one of the providers. This configuration is known as *multiaddressing*.

Even if this set up guarantees the scalability of the multihoming solution, such multi-addressed configuration is not without difficulties of its own when attempting to

* Corresponding author.
E-mail addresses: marcelo@it.uc3m.es (M. Bagnulo), alberto@it.uc3m.es (A. García Martínez), azcorra@it.uc3m.es (A. Azcorra), delaunois@info.ucl.ac.be (C. de Launois).

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.06.009

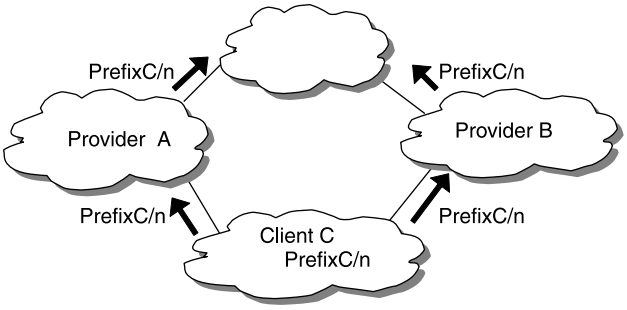

Fig. 1. BGP route injection for the provision of multihoming in IPv4 networks.

provide the additional features mentioned above. In particular, this configuration is not directly compatible with ingress filtering techniques usually deployed in service providers [4]. The incompatibility is caused by the lack of coordination between the IPv6 source address selection mechanism, performed by the host, and the path selection mechanism, performed by the intra-site routing system. As long as outgoing packets are routed through the provider that has delegated the prefix contained in the source address, packets will flow freely, but when those packets are routed through a different ISP, they will be discarded by the ingress filtering mechanism due to source address incompatibility. It must be noted that because of this issue packets may be discarded even in a scenario without failures.

Additional difficulties arise when providing reliability features as well. Required fault tolerance capabilities include both the establishment of new communications after an outage and the preservation of established communications through failures. Multi-addressing complicates the establishment of new communications after an outage, since in order to establish new communications after an outage, the endpoints of the new communication have to wisely select the address to use, avoiding those corresponding to unavailable paths. Currently available address selection procedures do not consider this issue, so additional mechanisms are required to provide the proposed feature.

Further obstacles have to be sorted out in order to preserve established communications through outages. In addition to the need to adapt the address used during the lifetime of the communication according to the available providers, the address replacement has to be performed in a transparent fashion with respect to transport and application layers, in order to actually preserve the established communication. Current applications and transport layers, such as TCP, identify the endpoints of a communication through the IP addresses of the nodes involved, implying that the IP addresses selected at the communication establishment time must remain invariant through the lifetime of the communication. But as it has been presented earlier, once that an outage has occurred in one of the available ISPs, the associated address becomes unreachable, so an alternative address has to be used in order to convey packets to the multi-homed host. These two constraints impose that after an outage, packets will carry a different address, corresponding to an available ISP, but they will be presented to transport and application layers as if they contain the original address, in order to be recognized as belonging to the established communication. Such approach requires additional mechanisms in both ends of the communication in order to preserve a coherent mapping between the IP addresses presented to the transport and application layers and those addresses actually contained in the packets.

Finally, multi-addressing also implies that traffic engineering capabilities will depend on address selection, since the path will be determined by the address used to reach the multihomed host, conversely to the current single-address approach in which traffic engineering is provided through proper configuration of the routing protocols, essentially through the manipulation of BGP attributes. Summarizing, multi-addressing guarantees the required scalability, but it introduces difficulties when providing ingress filtering compatibility, establishing new communications after an outage, preserving established communications through outages and when performing traffic engineering.

In order to overcome the identified difficulties, additional mechanisms are needed. Moreover, because the required mechanisms present dissimilar level of complexity and involve different amount of network elements, the deployment effort for solving each of the different parts of the problem will vary considerably. It seems wise then, to propose a roadmap to multihoming, in which different intermediate goals are defined based on the deployment effort required to deploy the mechanism and on the benefit obtained from it. In the first stage, mechanisms to restore minimal functionality equivalent to single homed environment are deployed. Some of these mechanisms are limited to the multi-homed site, so their deployment only depends on the involved party. The next stage would be to provide some of the benefits of multihoming, such as limited fault tolerance capabilities to allow the establishment of new communications after an outage. In this case, the additional mechanisms required mainly reside within the multihomed site, so again, the deployment effort is limited to the interested parties. However, these fault tolerance mechanisms imply the modification of multi-homed hosts and/or routers, imposing additional costs. Similarly, the deployment of some traffic engineering mechanisms affects only the multihomed site, so its rapid adoption can be expected. Finally, mechanisms proposed to preserve established communications require modifications in both ends of the communications, imposing the upgrading of all the hosts of the Internet. It is then expected that its adoption will take place in a longer timeframe than the previous ones.

The contributions of this paper include the presentation and analyse mechanisms suitable for the proposed roadmap, i.e. the provision of the minimal functionality equivalent to the single homed environment, the provision of fault tolerance capabilities to allow the establishment of new communications after an outage, the provision of traffic engineering

capabilities at the multihomed site and finally for the preservation of established communications through outages.

The remaining of this paper is structured as follow: In Section 2 we present the reference topology and some additional background information required. In Section 3 we describe the mechanisms to be deployed in the first stage of the roadmap i.e. the mechanisms for the provision of ingress filtering compatibility, the mechanisms for establishing new communications after an outage and the tools for traffic engineering. In Section 4, we present the mechanisms to be deployed in the second stage of the proposed roadmap i.e. the protocol for preserving established communications through outages. Section 4 describes the related work and Section 5 present the conclusions of this work.

2. Background for the incremental deployment of IPv6 multihoming

In this section we will present some background information required for the presentation of the proposed roadmap for the incremental deployment of the different mechanisms involved in an IPv6 multihoming solution. We will first present the reference topology where the multihoming solution is deployed and then we will present the rationale for the proposed roadmap.

2.1. Reference topology

As described in the introductory section, we are considering a multihomed site that obtains internet service from several ISPs. Each of the ISPs delegates an address block from the own aggregate to the multihomed site, resulting in the aforementioned multiaddressing configuration. In such configuration, each host of the multihomed site will configure multiple addresses, one from each address block available in the site.

The resulting configuration is illustrated in Fig. 2. In the depicted configuration, the multihomed site is connected to ProviderA and ProviderB, which delegate prefixes P_A and P_B respectively. A given host within the multihomed site obtains two addresses (one for each prefix available) $P_A{:}L_1{:}i_1$ and $P_B{:}L_1{:}i_2$

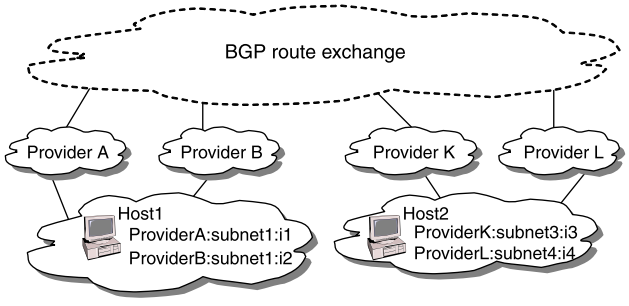

Fig. 2. Multiaddressing-based multihomed in IPv6.

2.2. Rationale of the roadmap for an incremental deployment of IPv6 multihoming

As it has been described in the introduction, a multihoming solution for IPv6 will comprise multiple mechanisms to address the different issues involved. As stated earlier, the following mechanisms are needed: an ingress filtering compatibility mechanism, a mechanism to establish new communications after an outage, a solution for preserving established communications through outages and traffic engineering tools. Each ones of the aforementioned mechanisms involve different deployment effort and result in different benefits. The roadmap for the deployment of the different mechanisms required proposed in this paper is based on the deployment effort required for the adoption of the mechanisms.

We can roughly identify two types of mechanisms:

First, those mechanisms that only require modifications on the multihomed side of the communication. Such mechanisms result in immediate benefit once that they are adopted by the multihomed site. Within this category we can find the mechanisms to provide ingress filtering compatibility, the mechanisms to establish new communications after an outage and some of the tools to provide traffic engineering. Even though all these mechanisms only require modifications within the multihomed site, some of them require modifications on the end systems, while others support legacy hosts within the multihomed site. So, we can formulate an additional subdivision inside this category, based on the elements affected by the deployment of the mechanisms. The mechanism for providing ingress filtering compatibility and some of the traffic engineering tools support legacy hosts, while the mechanism for establishing new communications after an outage require upgrading the host within the multihomed site.

Second, those mechanisms that require support from both sides of the communications i.e. the multihomed host and its peer. The deployment of such mechanisms does not involve the multihomed site, which obtains direct benefits from its deployment, but it also involves the potential parties, which do not have any direct motivation to deploy the mechanisms. The mechanisms for preserving established communication through outages fall within this category, significantly increasing the deployment effort required.

So, the proposed roadmap is the following:

mechanisms that only require support form the multihomed site:

- Mechanisms that support legacy hosts: ingress filtering compatibility
- Mechanisms that require upgrading the hosts of the multihomed site:
  - Establishing new communications after an outage
  - Traffic engineering tools

Second stage: mechanisms that require support from both ends of the communication

- Preserving established communications through outages

In the following sections we will present all the mechanisms involved in the provision of IPv6 multihoming.

3. First stage: mechanisms that only involve the multihomed site

In this section we will present the mechanisms proposed for the first stage of the deployment roadmap. The mechanisms included in this stage only affect the multihomed site i.e. they do not require support from devices external to the multihomed site. This means that the deployment effort is concentrated only in those parties that obtain a direct benefit from the adoption of the mechanisms. For this reason, we consider that such mechanisms can be deployed in the short term. Within those mechanisms, we can identify two types of mechanisms, those that support legacy hosts and those that require modifications in the multihomed hosts. We consider that those mechanisms that support legacy hosts can be deployed faster than the other ones, since they involve a reduced number of devices. So, next we will present the mechanisms that support legacy hosts and in the following section we will present the mechanisms that require support from the hosts of the multihomed site.

3.1. Mechanisms that support legacy hosts: ingress filtering compatibility

The mechanism for the provision ingress filtering compatibility is the first one to be deployed in the proposed roadmap since it only requires the modification of a reduced number of devices within the multihomed site. In addition to this, this mechanism restores functionality that is lost when becoming multihomed, as it is described next.

Provided that there is no failure in the network infrastructure, it may happen that ingress filtering [4] may preclude the possibility of communication. This filtering is applied to avoid malicious usage of addresses that are not owned by the users, i.e. address spoofing. The motivation is as follows: If a user spoofs an address, even an address that does not belong to the topological IP region in which the user is placed, it could perform attacks anonymously, since the real location of the attacker could not be easily obtained. Note that if the attacker is not placed in the path between the attacked node and the network from which it has obtained the address—improperly—the attacker may send packets, but could not receive the responses, that will be sent to the network for which the address originally belongs. However, in this situation Denial of Services attacks can be issued, by flooding the network to which the attacker's packets are directed, or by flowing the network from which the source addresses have been spoofed (if the response traffic generated upon the attacker's resquet is high). A protection measure against address spoofing is to deploy ingress filtering in the providers of connectivity. Ingress filtering consists of filtering those packets generated by a client with source addresses that do not belong to the address range assigned to that client. With respect to the considered multihoming scenario, a packet generated by Host1, with a legitimate address obtained from ProviderA, will freely flow if the intra-site of the multihomed network forwards the packet through ProviderA but it will be discarded by ingress filters if routed through ProviderB. Such behavior results in packet loss even if no outage has occurred. This means, that the resulting performance is worse than the single homed case, when packets are discarded only if a failure has occurred. For this reason, the adoption of this mechanism is deemed urgent and it is placed in the first stage of the roadmap. The proposed mechanisms for the provision of ingress filtering compatibility are described next.

Because of the reasons presented earlier, in order to preserve ingress filtering compatibility, the packet has to be forwarded through the ISP that is compatible with the selected source address. Current destination address based routing does not take into account the source address of the packet, making it unsuitable to provide ingress filtering compatibility. Source Address Dependent (SAD) routing is a natural option to overcome the difficulties caused by ingress filtering. SAD routing essentially means that routers maintain as many routing tables as source address prefixes involved, and packets are routed according to the routing table corresponding to the source address prefix that best matches the source address contained in the packet header. SAD routing would then provide ingress filtering compatibility for routing packets flowing from the multihomed site to the Internet. In this case, the source address of the exiting packets has been determined by the host that initiated the communication (the host in the multihomed site, or the external host through the selection of the destination address of the initial packet) and then the routing system will forward the packet to the appropriate exit router in order to guarantee ingress filtering compatibility. The source address selection determines the ISP to be used for routing packets, since, because of address filtering, the source address determines the forward path from the multihomed site to the rest of the Internet; it also determines the ISP to be used in the reverse path, since the source address used in the initial packets will become the destination address of the reply packets. In order to enable SAD routing within a site, SAD routing support is not required in all the routers within the site, but it has to be adopted in a connected SAD routing domain that contains all the exit routers. For additional considerations about the adoption of SAD routing in multihomed sites, the reader is referred to the Refs. [5,6].

It should be noted that SAD routing provides the multihomed hosts with a tool to determine the site exit path used to route their packets. This tool will be useful for the provision of fault tolerance functionalities since the selection of a source route implies the selection of an exit ISP. Because of SAD routing, when a host within the multihomed site changes the source address, the packets will be routed through a different ISP.

3.2. Mechanisms that require upgrading the hosts within the multihomed site

The next step proposed in the roadmap is the deployment of those mechanisms that only involve the multihomed site but they require upgrading the hosts within the site. In this category there are two mechanisms: the mechanisms for establishing new communications after an outage and the mechanisms for the provision of traffic engineering capabilities. Since fault tolerance is the preferred feature for most multihomed sites, will present the related mechanisms first and then we will move to traffic engineering issues.

3.2.1. Establishing new communications

In this section, we address the establishment of new connection in case of failures. If the failure arises in one of the providers or links close to Host2 (for example in ProviderK), and Host1 starts the communication, it can happen the following: Host1 accesses DNS to obtain the addresses of Host2. With the received addresses, the Address Selection procedure [7] selects a ⟨source address, destination address⟩ pair. Suppose that the destination address belongs to the address range delegated to the network of Host2 by ProviderK. The application tries to establish the communication using these parameters, but it is not possible, and it realizes of the communication problems (the transport connection establishment process indicates a failure, or a timer in the application expires). In this case, the behaviour expected for typical applications is to retry with another destination address from the set obtained from the DNS. Repeating this process, connectivity failures close to Host2 can be solved if it exists at least one valid path to the destination, with a time penalty of the sum of the time required for the timer expiration for all the explored paths.

If the failure occurs close to Host1, the exploration of alternative paths is achieved by the variation of source addresses. This would require modifications in the Address Selection mechanism, modifications that are currently under discussion. Note that in the proposed scheme, failure detection at the endpoints relies only on the information provided by timer expiration, and this information is not enough to determine if the failure has occurred in the destination endpoint (for example, because it has been turned off), in a location close to the destination endpoint (requiring changes in the destination address), or close to the source (requiring changes in the source address). Additionally, multiple failures may arise, restricting the combinations of source and destination addresses that are valid. Therefore, only the exploration of all the possible pairs ⟨source address, destination address⟩ assures connectivity if there is at least one valid path among the endpoints.

Taking into account all these issues, we propose a mechanism based on the retrial of different source address and destination address combinations by the initiator of the communication. Moreover, in order to provide a mechanism to establish new communications after an outage, it is possible to benefit from the capabilities provided by the SAD routing adopted for the provision of ingress filtering compatibility presented earlier.

Since the basic assumptions behind adopting SAD routing for multihoming support are that the source address is determined by the initiating host, and that each source address prefix determines an exit ISP, fault tolerance capabilities can be provided by the hosts themselves, based on a trial and error procedure. Considering that each source address available in a host is bound to an exit path, the host can try different exit paths by changing the source address.

The resulting behavior is the following:

- The initiating host (Host1) selects a pair of source and destination address.
- The intra-site SAD routing system routes the packet according to the source address through the corresponding site-exit router.
- If the selected destination is reachable through the selected source address, then the packet is forwarded towards the site exit router that leads to the ISP corresponding to the source address prefix selected.
- If the selected destination is unreachable, the packet is discarded and an ICMP Destination Unreachable is sent back to the host.
- The host selects a new pair of source and destination addresses and retries.

3.2.2. Tools for traffic engineering

As we remarked already, in IPv6 a multihomed site is bounded to the prefix delegated by the correspondent ISP, and the selection of the prefix used will determine the path used to reach the multihomed site. Consequently, Traffic Engineering (hereafter TE) will be heavily related to address selection mechanisms. In this section we will present a framework composed of various tools that can be successfully combined for providing TE capabilities in IPv6 multihomed sites that have multiple global prefixes configured.

We will next analyze the impact of address selection in the provision of TE capabilities. We will first analyze the case of ingress traffic and then we will move to the case of outgoing traffic.

– Ingress traffic (from the multihomed site's perspective) When multiaddressing is adopted, the multihomed site is reachable through a given route/ISP only through the proper prefix, so in order to reach the multihomed site through a given ISP, the correspondent prefix/address has to be used in the communication. This implies that the path used is determined by which address is used among the multiple addresses available for a multihomed host, and that TE capabilities for traffic flowing to the multihomed site will be heavily influenced by the address selection process.
– Egress traffic (from the multihomed site's perspective) In the multiaddressing scenario, because of ingress filtering [4], each ISP will only forward packets that carry the appropriate prefix in the source address. So, in order to avoid being discarded by ingress filters, packets will have to flow towards the Internet through the ISP associated with the prefix included in the source address. This implies that the selection of the address of the multihomed host that is used for the communication will also determine the ISP used for outgoing packets.

So, when multiple PA addresses are available in a multihomed site, the selection of the address of the multihomed host that is used for the communication determines both the ingress and egress path. It becomes relevant then to understand how the address selection is performed. Current default address selection algorithms are defined in RFC 3484 [7] and they define a set of rules and data structures that allow the host to select among the multiple addresses available. In particular, hosts have a policy table that provides the means to express policy considerations when selecting among multiple addresses. It is a longest prefix match table that takes an address (source or destination) as input, and returns two values: a label value and a precedence value. The label value is used to match destination addresses with source addresses. The precedence value is used to select destination address among a set of available destination addresses. Such policy table can be used to express preferences that reflect the TE considerations of the multihomed site. The main part that is still missing is a mechanism to automatically configure the policy table. We will next describe the proposed mechanism for such task, called NAROS [8].

The NAROS approach is a solution which allows sites to engineer their incoming and outgoing interdomain traffic without any manipulation of BGP messages. 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 so that 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 the NAROS service addresses the issues related to source and destination address selection and traffic engineering.

When a host initiates a connection with a correspondent, it must determine the best source address to use among its available addresses. The source address selection algorithm previously described 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 the host-centric multihoming case and only the policy table would provide guidance if it is properly configured. The principle that 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. The obtained information is stored in the policy table for future use.

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 can run over UDP or over another protocol like ICMPv6 [9]. The NAROS protocol contains only two messages: NAROS request and NAROS response.

The first message is used by a client to request which source address to use to reach a given destination. The parameters included in a NAROS request are at least the destination address of the correspondent and the source addresses currently allocated to the client. The NAROS server should only be contacted when the default source address selection procedure [7] cannot select the source address.

The NAROS response message is sent by a NAROS server and contains the source address 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 in the policy table during the announced lifetime.

So, considering that when a host selects a source address, it also selects the provider through which the packets will be sent and that the source address to use is selected by the NAROS service, this can naturally be used to perform traffic engineering.

For example, in order to equally balance the traffic among the multiple providers, a NAROS server can use a round-robin approach. Each time it receives a 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. Note that this 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 additional complexity.

4. Second stage: mechanisms that require support from both ends of the communications

4.1. Protocol for preserving established communications through outages

In this section we will present the last mechanisms of the IPv6 multihoming solution proposed in the roadmap. The mechanism described in this section allows the preservation of the established communications through outages. The deployment of such mechanisms requires modifications of both ends of the communication, implying deployment effort not only in the multihomed site that obtains direct benefits from the adopted mechanisms, but also from the external hosts. So, we next present a mechanisms for preserving established communications (such as a TCP connection-but not exclusively) in case of a failure of the network infrastructure. It would be desirable that this functionality could be provided to the applications in a transparent fashion, i.e. without requiring changes in the applications for benefiting from the service. We can also consider that it could also be beneficial if the service is also provided transparently to the transport layer (TCP, UDO and other protocols). To fulfill both requirements, it has been proposed that the multihoming functionality could be provided at the network layer. The following considerations are developed from the information available at several working documents [10–13].

A network layer multihoming solution would be in charge of changing the IP addresses used for routing the packets, to ensure that only addresses that define valid paths for a considered communication are used. The addresses included in the actual IP packets, addresses that are used for the routing of the packets, are known as locators. However, the transport layer usually employs IP addresses within the basic parameter that define a communication. TCP, for example, uses the source address and destination address, apart from source port and destination port, to identify uniquely a connection. These addresses have been usually provided by the application of the host that is starting the communication, and they can also be used by the applications for identifying the communication. When we use the IP addresses this way, we will call them identifiers. After establishing a communication using previously defined mechanisms, a multihoming solution will be in charge of managing different locators to ensure that connectivity between endpoints is preserved if there is at least one valid path between them, while keeping a single identifier for the upper layers.

The management of identifiers and locators will be performed in a entity at the endpoints, included in the network layer as a sublayer that will be known as identification sublayer. This sublayer will exchange the available locators that are available for each one of the endpoints, to make the locators ready to be used in case of failures. Following the example presented in the previous section, once established the communication between Host1 and Host2, either (for example Host2) could initiate an exchange in which it could inform the other about all the locally available locators, and the other would answer correspondingly. This task would be performed by a specific protocol for multihoming, and a new state will be generated for the participants in the information exchange. From now on, if a problem is detected when a given pair is used in the communication, each one of the hosts can modify the pair to find a valid path.

Tools for identifying flows even after a change in the locator pair are required. It can be necessary to include in the packets a flow identifier pre-accorded between both endpoints by means of the multihoming specific protocol. This identifier can be included in the packets in an IPv6 header extension.

To detect a locator pair that determines a path in which a failure occurs, there are several alternatives [12]. First, there are mechanisms that allow the detection of local problems, such as an interface that is no longer operational, etc. Additionally, we can rely on the information provided by upper layers to be able to detect problems in communications from explicit notifications of failure, or from the absence of positive confirmations. TCP, for example, can inform the network layer about communication problems if a TCP confirmation has not been received for some time. Finally, we can add specific signaling procedures for multihoming, sending packets that could check reachability between a given pair of locators. This procedure can be analogous to an ICMP ping (a request with its corresponding response), and it can be used for checking the state of the locators currently being used for the communication, in parallel to the data exchange, i.e. out of band. A timer will check if the responses are received within the appropriate period, and if this is not the case, a process for the selection of a new locator pair will be raised. The process selection will include a reachability test for different locator pairs. Once selected a new pair, this pair will be used for subsequent data packet exchanges. When the remote host receives packets with the new locators, this host will start a reachability test taken as candidate pair the locators received from the host that has initiated the change.

The ability to change locators while a communication is being held enables security problems. As a criteria for the analysis of the security offered by the new multihoming solutions, it is usually required that they should not enable vulnerabilities that are not possible in the current IPv4 infrastructure [14]. With the tools that have been presented so far, we can think of new attacks that are not possible when the identifier and locator functions are integrated in a single IPv4 address, as it is the current case. An example is

the time-shifting attack, in which an attacker, HostX, tries to communicate with Host1 hijacking the identity of Host2. To achieve this, HostX is placed in the path between one of the locators of Host1 and one locator of Host2, in order to be able to intercept all the packets that Host1 would send to Host2. Although this attack can be performed in the current IPv4 infrastructure, the multihoming tools presented above add a new option that was not possible before: the attacker HostX can use the multihoming protocol for indicating Host1 that after that moment, it can be accessed in another locator. In this way, the attacker is not forced to be present all the time in the path between Host1 and Host2. It should be noted that the time-shifting attack is not the only attack that is enables with the new multihoming tools.

4.2. Security features of the protocol

The main obstacle in defining a mechanism for the management of multiple locators in multihoming environments has been the provision of the appropriate security level. Mechanisms based on cryptography and addresses generated cryptographically have been proposed to avoid identity theft [15,16], but these solutions suffer from the high computational cost of performing asymmetric key operations, cost that can be intolerable in scenarios such as a server with a large number of requests per second. So the adoption of an alternative mechanism that guarantees the link between a set of locators with an identity without incurring in large computational costs is proposed. In this proposal, a multihomed host Host1, located in a network with different prefixes corresponding to different providers, generates interface identifiers (the 64 less significant bits of the IPv6 address) for its own address by performing a hash of the available prefixes. In this way, a 'signature' obtained from the prefixes assigned to the host is included in all its addresses. When a corresponding host Host2 establishes a communication using a particular address of Host1 (obtained for example in the DNS), and Host2 receives by means of the multihoming protocol the alternative locators of Host1, Host2 can check that the received locators are legitimate. To do so, Host2 performs a hash of the prefixes of the locators that should generate the interface identifier of the address originally used for establishing the communication. An attacker would require in the order of 2^63 operations (due to the number of bits of the hash) to obtain a set of prefixes different from the initially specified that fulfil the hash check, and at the same time include a locator of the attacker.

In more precise terms, the security architecture proposed for the multihoming protocol is based in the use of new type of addresses, called Hash Based Addresses [17]. Hash Based Addresses are a new type of global IPv6 addresses that incorporate a cryptographic one-way hash of the prefix-set available in the multihomed site into the interface identifier part. The result is that the binding between all the addresses of a multihomed host is encoded within the addresses themselves, providing hijacking protection. Through this tool, any node that is communicating with a multihomed node can efficiently verify that the alternative addresses proposed for continuing the communication are bound to the initial address through a simple hash calculation. In order to benefit from the proposed security mechanism, the addresses of each multihomed host have to constitute an HBA set. In a general multihoming scenario considered, a multihomed host attached to a link where N 64-bit prefixes [18] are available (P1::/64, P2::/64,…, PN::/64 1) generates the interface identifier part of each one of its addresses as a 64 bit hash of the prefix set available in the link and a random nonce. Including a random nonce enables the generation of multiple HBA sets associated to the same prefix set. After generating the interface identifier parts, the addresses of the HBA set are generated by prepending the different prefixes of the prefix set with the interface identifier parts. The output of the described procedure is a set of N HBAs that carry information about the prefixes available in the multihomed site within their interface identifier part. Each one of the generated addresses will have a different prefix from the input prefix set, while their interface identifier part will contain information about the complete prefix set in the form of a hash of the full prefix set. Because of their nature, each address contains information about all the other addresses of the set, and a receiver can easily verify if two addresses belong to the same set through a cost effective hash operation. After this verification, the receiver can securely use them interchangeably. In the next section we will describe how HBAs can be used to prevent time-shifted hijacking attacks in the Session Context Creation exchange.

4.3. Protocol walkthrough

In this section we will consider a case study to show the behavior of the proposed IPv6 multihoming solution in a common scenario. Consider two hosts that are capable of handling the IPv6 multihoming protocol presented before, namely HostX, holding N different prefixes (PX1,…PXN), and HostY being configured with M different prefixes (PY1,…PYM). The addresses assigned to each host have been generated according to the HBA specification, resulting in set {PX1:IX1… PXN:IXN} for HostX, and set {PY1:IY1…PYM:IYM} for HostY.

Typically, an application in HostX issues a DNS request for a name associated to HostY, obtaining in the request some subset of the addresses assigned to HostY. The regular address selection process for IPv6 specified by RFC 3484 [7] is used by HostX to select one of the addresses of HostY (PYJ:IYJ) as destination address for the outgoing packets, and one of its own address (PXK:IXK) as source address. These addresses selected at the beginning of the communication will also be used as endpoint identifiers for transport and application layers when required. If the path defined by

the selected addresses is valid, the transmission of upper layer data can proceed; otherwise, RFC 3484 specifies a procedure for establishing a new communication by the selection of new destination and source address pairs until a valid path is found. Note that the procedure of establishing the communication does not require any multihoming protocol exchange.

After a while, we can suppose that any of the entities involved in the communication (applications, transport or network layers) in HostX requests higher quality in terms of reliability, so it initiates the IPv6 multihoming protocol exchange to create at both endpoints the state required for recovering the communication in case of failure.

Next, some time later a failure in the network occurs, preventing communication through the provider delegating the PXK prefix to HostX. A timeout in HostY is raised after a period of T seconds without receiving any packet for the pair of identifiers considered. A Reachability Test Request packet is sent to HostX using the current locators to check the validity of the path that is being used. Since HostY receives no answer, it initiates the Alternative Locator Pair Exploration mechanism sending several packets with different source and destination addresses. As a result, responses are received for all the pairs that do not contain the PXK prefix. By selecting one of these pairs, for example <PXR:IXR, PYS:IYS>, HostX will define the locators to be used for sending data to the other endpoint. HostX will then receive data packets with the new locators, interpreting this change as a hint to verify the currently used locator pair through a Reachability Test. As this check fails, HostX will check for new locators. In this case, HostX can try to optimise the Exploration by checking reachability for just the locator pair received, although it is not required for both endpoints to use the same locators for the communication.

Some time later, the application at HostY decides to stop the communication, so the identity layer at HostY sends a Close Request to HostX, which will be acknowledged to allow proper state disposal.

5. Related work

Over 40 solutions for IPv6 multihoming have been proposed in the last few years. In this section we will describe the most relevant ones and compare them with the approach proposed in this paper. The Host Identity Protocol (HIP) architecture [15] presents some commonalities with the solution presented in this paper. In particular, HIP includes the creation of a HIP layer between the IP layer and the transport layer that would perform a mapping between the identifiers and the locators, just as in the Multihoming Sub-Layer proposed in this paper. The fundamental difference between these two approaches is the nature of the identifiers. In both cases, identifiers are cryptographic in nature, but in the case of HIP, the identifiers contain a one-way hash of a public key. Moreover, in HIP, a strict separation between locators and identifiers is proposed and the identifiers are not valid locators, as opposed to the HBA addresses described in this paper. Such characteristic of the HIP architecture results in some difficulties when supporting some types of applications, such as referrals and call-backs [13]. Moreover, because of the non-hierarchical nature of the identifier name space, it is not possible to deploy a directory service that stores the information about identifier to locator mapping. This means that it would be impossible in the short term to deploy a service that contains information about the set of locators associated with a given identifier. In addition to this, the HIP approach imposes an extensive usage of public key cryptography, which is expensive in nature. This is considered a problem especially for heavy loaded servers that have to maintain hundreds of thousands simultaneous communications. Because all these reasons we consider that in the short term, HIP is an inferior solution than the one presented in this paper. In order to overcome the limitations presented by the HIP solution concerning the support for referral and call-back applications, it is possible to adopt a middle-ground scheme where the identifiers are also valid locators and they carry a hash of a public key in the lower 64 bits, as in Cryptographic Generated Addresses [16]. Such approach would provide a similar support than the presented approach with respect to applications, but it would still impose the workload required by public key cryptography. The benefit of a CGA based approach over the HBA based approach is that it supports dynamic locator sets, i.e. the locator set is not predetermined and it can evolve through time. This is considered an advantage when providing support for mobile environments and renumbering events. However, it should be noted that HBA and CGA approaches are compatible, since it is possible to define addresses that are simultaneously CGA and HBA [17]. Other approaches propose the usage of regular IPv6 addresses both as locators and as identifiers. The difficulty with such approaches is the provision of the required protection against time-shifted hijacking attacks, as described before. In the case of Mobile IPv6 (MIPv6) based approaches [19,20], the security is achieved through the return routability procedure. Such procedure is inherently incompatible with the multihoming goal of fault tolerance, since the verification of the identity is based in the usage of reachability tests. This is so because once a failure has occurred, the locator that is being used as identifier will be unreachable and the return routability check will fail. Other approaches such as NOID [21] use a third trusted party to validate the mapping between the multiple IPv6 addresses. In the particular case of NOID, the DNS is used to store all the available IP addresses of a multihomed node. This approach does provide the required security features, but it requires the availability of direct and reverse DNS records for all the hosts within a multihomed site. Even this may be possible for some well-managed sites,

it does not seems feasible for small sites that do not even have their own domain name.

6. Conclusions

In this article we have presented a solution for IPv6 multihoming and a roadmap for its incremental deployment. The large availability of IPv6 addresses allows the deployment of multiaddressing configurations that circumvent the scalability problems that pose current IPv4 solutions. The proposed solution consists in four type of mechanisms: mechanism for the provision of ingress filtering compatibility, mechanisms for establishing new communications after outages, tools for traffic engineering and a protocol for preserving established communications through outages. In particular, it has been proposed a modification in the routing inside the multihomed networks to avoid packet discarding due to ingress filtering, and also variations in the Address Selection mechanism to allow the exploration of different ⟨source address, destination address⟩ pairs in case of failure. For the preservation of established communications larger changes are required such as a model in which identifiers and locators are split, the definition of a new protocol, new states in the hosts, and the deployment of failure detection mechanisms in the paths. The 128 bits of the length of the IPv6 address allows the inclusion of cryptographic information or a hash of relevant information to provide sufficient security when performing locator redirections. In particular, the usage of a new type of addresses called HBA that are cryptographic in nature and that incorporate a one-way hash of the prefix set available in the multihomed site in their interface identifier part. The result is that all the addresses available in a multihomed host are inherently bound to each other, and the host can securely use them interchangeably.

In addition, a roadmap for the deployment of the aforementioned mechanisms is proposed, based on the trade-off between deployment effort and benefit of each one of the mechanisms. The proposed roadmap consists in an initial stage where the mechanisms that can be locally adopted in the multihomed sites are deployed. Those include the mechanisms for the provision of ingress filtering compatibility, the mechanisms for establishing new communications and the tools for performing traffic engineering. The second stage include those mechanisms that require modification of both ends of the communication, in particular the protocol for preserving established communications through outages,

We conclude that the proposed architecture preserves the scalability of the global routing system, it does not introduce new vulnerabilities in the Internet and it is easy to adopt since it does not require complex management in the end-site. In particular with respect to the last point, it should be noted that none of the presented mechanisms require manual configuration, allowing poorly managed sites to easily deploy the proposed solution. Moreover, as opposed to the multihoming currently deployed in IPv4, the fault tolerance capabilities of the solution do not require complex configuration of BGP or other protocols. In the presented approach, fault tolerance support is directly implemented in the end-hosts and work without requiring user configuration. As opposed to the IPv4 case, IPv6 allows small networks-even residential ones- to benefit from multihoming. Even though there is not a clear IPv6 killer application yet, we can definitely expect that IPv6 multihoming support will contribute to IPv6 success.

References

[1] J. Abley, K. Lindqvist, E. Davies, B. Black, V. Gill, IPv4 Multihoming Motivation, Practices and Limitations, Internet Engineering Task Force (IETF), RFC 4116, 2005.
[2] C. Labovitz, A. Ahuja, A. Bose, Delayed Internet Routing Convergence, SIGCOMM 2000, 2000.
[3] G. Huston, Commentary on Inter-Domain Routing in the Internet, Internet Engineering Task Force (IETF), RFC 3221, 2001.
[4] P. Ferguson, D. Senie, Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing, Internet Engineering Task Force (IETF), RFC 2267, 1998.
[5] M. Bagnulo, A. García Martínez, J. Rodríguez, A. Azcorra. The Case for Source Address Dependent Routing in Multihoming, Proceeding of the International Workshop on Multimedia Interactive Protocols and Systems, Lecture Notes of Computer Science, Springer-Verlag, 2003.
[6] C. Huitema, R. Draves, M. Bagnulo, Ingress filtering compatibility for IPv6 multihomed sites, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2004.
[7] R. Draves, Default Address Selection for Internet Protocol version 6 (IPv6), Internet Engineering Task Force (IETF), RFC 3484, 2004.
[8] C. de Launois, O. Bonaventure, NAROS: Host-Centric IPv6 Multihoming with Traffic Engineering, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2003.
[9] A. Conta, S. Deering, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, Internet Engineering Task Force (IETF), RFC 2463, 1998.
[10] E. Nodmark, M. Bagnulo, Multihoming L3 Shim Approach, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2005.
[11] M. Bagnulo, J. Arkko, Functional decomposition of the multihoming protocol, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2005.
[12] J. Arkko, Failure Detection and Locator Selection Design Considerations, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2005.
[13] E. Nodmark, Shim6 Application Referral Issues, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2005.
[14] J. Abbley, B. Black, V. Gill, Goals for IPv6 Site-Multihoming Architectures, Internet Engineering Task Force (IETF), RFC 3582, 2003.
[15] R. Moskowitz, P. Nikander, P. Jokela, T. Henderson, Host Identity Protocol, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2005.

[16] T. Aura, Cryptographically Generated Addresses (CGA), 2004, Internet Engineering Task Force (IETF), RFC 3972, 2005.
[17] M. Bagnulo, A. García-Martínez, A. Azcorra. Efficient security for IPv6 multihoming, ACM Computer Communications Review, 35(2) (2005) 61–68.
[18] R. Hinden, S. Deering, Internet Protocol Version 6 (IPv6) Addressing Architecture, Internet Engineering Task Force (IETF), RFC 3513, 2003.
[19] D. Johnson, C. Perkins, J. Arkko, Mobility Support in IPv6, Internet Engineering Task Force (IETF), RFC 3775, 2004.
[20] M. Bagnulo, A. García-Martinez, I. Soto, A. Azcorra, A MIPv6-based Multi-Homing Solution, EUNICE 2003: 9th IFIP Workshop on Next Generation Networks, 2003.
[21] E. Nordmark, Multihoming without IP Identifiers, Internet Engineering Task Force (IETF), Internet Draft (work in progress), 2004.