The Paths Towards IPv6 Multihoming

Cédric de Launois[*] **and Marcelo Bagnulo**[†]

[*]Université catholique de Louvain (UCL), Belgium

http://www.info.ucl.ac.be

delaunois@info.ucl.ac.be

[†]Universidad Carlos III de Madrid (UC3M), Spain

http://www.it.uc3m.es

marcelo@it.uc3m.es

Abstract— **Multihoming, the practice of connecting to multiple providers, is becoming highly popular. Due to the growth of the BGP routing tables in the Internet, the way to get multihomed in IPv6 is required to allow route aggregation in order to preserve the scalability of the interdomain routing system. After several years of development, the IETF and the research community propose drastic changes in how site multihoming shall be achieved in IPv6. Those changes will potentially affect the fundamental multihoming mechanism, as well as intradomain routing, traffic engineering and even the behaviour of hosts within the multihomed site. This paper presents the required functionalities and the constraints imposed to the solutions to the IPv6 multihoming problem. It surveys the main multihoming approaches that have been proposed to the IETF over the period 2000 - 2005. It proposes a taxonomy, and compares the solutions according to their mechanisms, benefits and drawbacks. This survey also outlines the major steps that have led to a new multihoming architecture for IPv6.**

I. INTRODUCTION

The Internet connects today more than 18000 *Autonomous Systems* (AS) [1]. An autonomous system can be defined as a set of networks operated by the same technical administration. The large majority of autonomous systems do not allow external domains to use their infrastructure, except to reach them. These domains are named *stub* ASes. Autonomous systems that provide transit services to other ASes are called *transit* ASes. The Border Gateway Protocol (BGP) is used to distribute routing information among routers that interconnect ASes.

Internet connectivity takes a strategic importance for a growing number of companies. Therefore, many ISPs and corporate networks wish to be connected through at least two providers to the Internet, primarily to enhance their reliability in the event of a failure in a provider network, but also to increase their network performances such as network latency. *Site Multihoming* refers to those stub networks that connect to at least two different network service providers, while *ISP Multihoming* refers to transit providers that are multihomed. In today's IPv4 Internet, at least 60% of stub domains are multihomed to two or more providers [2], and this number is growing. Many sites are expected to also require to be

multihomed in IPv6, even end users with multiple interfaces to GSM, UMTS, or 802.11 networks.

The traditional approach for multihoming in IPv4 is to announce, using BGP, a single site prefix to each provider. When multihoming with BGP, the site can use provider-independent (PI) addresses or provider-aggregatable (PA) addresses.

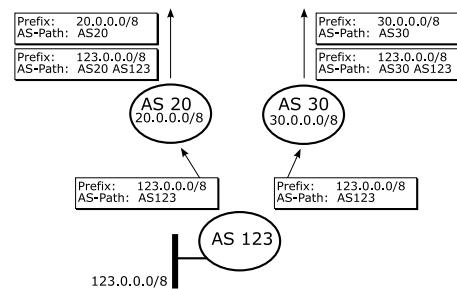### A. Multihoming with PI addresses

Fig. 1. IPv4 Multihoming using provider-independent addresses

In Figure 1, AS 123 was large enough to obtain and use provider-independent (PI) addresses. It announces its PI prefix to both its providers AS 20 and AS 30. Neither AS 20 nor AS 30 is able to aggregate the announcement made by AS 123. Therefore, both AS 20 and AS 30 announce to the global Internet the prefix of AS 123, in addition to their own prefix. These ISPs will propagate the route received to the global Internet. This provides the rest of the Internet with multiple paths back to the multihomed sites. It is clear that the use of PI addresses introduces an additional routing entry in the global routing system. Widespread multihoming in this manner presents scaling concerns [3], [4].

The use of provider-independent addresses has long been the preferred way to multihome in IPv4 [5]. A reason for this preference is that a site does not have to renumber if it changes of provider. Until the mid 1990s, it was relatively easy for a site to obtain a fairly large provider-independent address space from a Regional Internet Registry (RIR). Little justification was needed to obtain a /24 PI assignment. Due to the rapid depletion of the IPv4 address space, the RIRs no longer assign blocks as large as a /24 to small sites [6]– [8]. As a consequence, many sites are not able to obtain a PI assignment from their RIR.
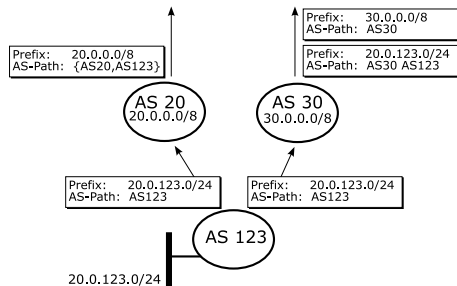
B. Multihoming with PA addresses

Fig. 2. IPv4 Multihoming using provider-aggregatable addresses

In Figure 2, AS 123 uses instead a single provider-aggregatable address space. This address space is assigned by the primary transit provider AS 20. AS 123 announces prefix 20.0.123.0/24 to both its providers AS 20 and AS 30. Here, AS 20 is able to aggregate this prefix with its own 20.0.0.0/8 prefix. AS 20 only announces the aggregate to the Internet. However, AS 30 cannot aggregate this prefix with its own prefix. Thus AS 30 still has to announce the prefix of AS 123 in addition to its own prefix. An awkward side effect is that almost all packets will now enter AS 123 via AS 30, due to the BGP decision process which favours more specific prefix advertisements.

Sites use PA addresses when their addressing requirements are not sufficient to meet the requirements for a PI address block by RIRs. The drawback is that the site is usually required to renumber if it decides to change of primary transit provider.

Even when PA addresses are used, multihoming with BGP in this manner still introduces an additional routing entry in the global BGP routing tables, as AS 30 cannot aggregate the prefix announced by AS 123.

Over the last ten years, the size of these routing tables has risen from 20,000 in 1995 to about 200,000 in 2005 [1], [9]. The current size of those tables causes operational issues for some Internet Service Providers, as it can decrease the packet forwarding speed and demands large memory space [10]. Moreover, several experts are concerned about the increasing risk of instability of BGP [11]. Hence, a new multihoming solution that ensures route aggregation is required for IPv6. This paper focuses only on *site* multihoming, as *ISP* multihoming has a widely accepted solution, which is to announce a set of routes with BGP to the upstream providers. This solution is considered acceptable in IPv6, since the number of transit providers is expected to remain reasonable.

After several years of development, the IETF and the research community have proposed drastic changes in how site multihoming shall be achieved in IPv6, compared to how it is done currently in IPv4. Those changes potentially affect the fundamental multihoming mechanism, as well as intradomain routing, traffic engineering and even hosts within the multihomed site. This paper outlines the steps that have led to a new multihoming architecture. It focuses on the design requirements and on the comparison of the major architectures that were proposed to the IETF over the period 2000-2005.

This paper is organised as follows. Section II states the problem of IPv6 site multihoming. It presents the required functionalities and the constraints imposed to the proposed solutions.

Next, this paper will follow a path in the decision tree that yields to the solution selected by the IETF. Sections III, IV and V focus respectively on the Routing, Middle-Box and Host-Centric approaches. As it will appear that Host-Centric is the most promising approach, several proposed solutions that belong to this class will be detailed in Section V. These solutions are classified into *Transport Layer* (Section V-B), and *Network Layer* (Section V-C) approaches, the latter approach being the one chosen by the IETF. Finally, at the end of the decision tree, several Network Layer solutions are presented, differing from the kind of identifier namespace. Among the solutions proposed, the SHIM approach is the one fostered by the IETF. It is detailed in Section VII.

Mechanisms, advantages and drawbacks of all solutions presented in this survey will be summarised in Table I and Table II.

II. PROBLEM STATEMENT

The overall issue of IPv6 site multihoming is to provide enough functionalities in order to fit the various motivations for multihoming, under several technical and non-technical constraints. The motivations for multihoming are described in section II-A, while the functionalities needed to meet these motivations are detailed in section II-B. The constraints are presented in section II-C.

A. Multihoming Motivations

Most sites request to be multihomed in order to protect themselves against failures of the links with their providers, as well as other failures within and beyond their providers. Sometimes, multihoming is used by a site to distribute its traffic between multiple transit providers, as a mean to achieve better network performances including delay, loss, jitter, or raw bandwidth. Multihoming is also often requested for some policy beyond technical scope, e.g. for cost or commercial reasons. A last motivation for multihoming is an economic, political or administrative independence with respect to the providers, especially if the site has its own provider-independent address space.

B. Multihoming Functionalities

The functionalities required for IPv6 multihoming solutions can be directly derived from the multihoming motivations. Two main functionalities can be distinguished: full fault-tolerance and traffic engineering capabilities. Fault-tolerance means that the exchange of packets between devices in the multihomed site and devices elsewhere on the Internet may proceed across re-homing events. In other words, an IPv6 multihoming solution should provide transport-layer surviv-ability across failure events. Additionally, an adequate set of traffic engineering functionalities is required to fulfil the load-sharing, performance, and policy motivations for multihoming.

C. Multihoming Constraints

Any multihoming solutions for IPv6 must respect several technical and non-technical constraints, as detailed in [12]. The main constraint is to preserve the size of the BGP routing tables in the Internet. A second constraint is that a multihoming solution should not preclude filtering procedures, for security reasons. The filtering consists in dropping the customers' packets entering in the ISP network that is coming from a source address not legitimately in use by the customer network [13], [14]. A third constraint is that a solution for IPv6 multihoming must not require cooperation between the providers of the multihomed site. Moreover, the solution should preferably not require any accommodation beyond that what a provider would do for a single-homed customer, so that home and enterprise networks can also get all multihoming benefits. This constraint is named here *multihoming independence*. Another constraint is that the impact on hosts, routers and the Domain Name System (DNS) should be limited, and that the multihoming mechanism should not be substantially more complex to deploy and operate than current IPv4 multihoming practices.

III. ROUTING APPROACHES FOR IPV6 MULTIHOMING

The main architectural approaches for multihoming can be classified according to the fundamental mechanisms they use to provide fault-tolerance, traffic engineering, route aggregation and multihoming independence. Three approaches can be distinguished: *Routing*, *Middle-Box*, and *Host-Centric*. This section focuses on Routing approaches. Section IV presents the solutions based on the use of middle-boxes. Finally, section V details several host-centric solutions. Mechanisms, advantages and drawbacks of all solutions presented in this paper are summarised later in table I.

The class of Routing approaches groups multihoming architectures that rely on the routing system in general to roughly provide fault-tolerance and traffic engineering functionalities. Routing mechanisms include the use of BGP, the filtering of BGP route advertisements, or the use of interdomain tunnels. IPv6 multihoming solutions that belong to this class are *IPv6 Multihoming with BGP* [15], *IPv6 Multihoming using Cooperation between Providers* [16] and *IPv6 Multihoming Support at Site Exit Router* [17].

A. IPv6 Multihoming with BGP

IPv6 multihoming with BGP adapts to IPv6 the traditional IPv4 multihoming method. The procedure is detailed in [15]. Like in IPv4, a site uses BGP to announce its own prefix to each provider. As explained previously and illustrated in Figure 3, this solution causes scalability problems, as each multihomed site introduces a new prefix in the BGP routing tables of all routers in the Internet, even if the site is using provider-aggregatable prefixes.

This multihoming solution is considered acceptable only for large ISPs and transit providers, which use provider-independent prefixes. It is not a solution for small ISP, home and enterprise networks, for scalability reasons but also because it requires the use of BGP.
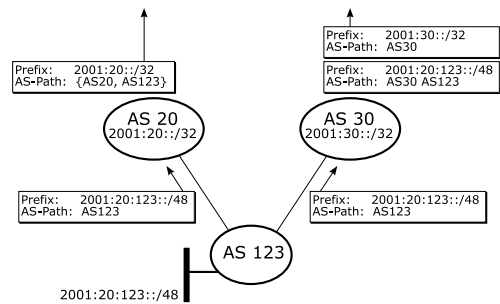

Fig. 3.   IPv6 Multihoming with BGP using provider-aggregatable prefixes

Fault-tolerance and traffic engineering are typically provided by adequate configuration of BGP and IGP. For instance, a multihomed site can engineer its outbound traffic by assigning appropriate IGP weights to its intradomain links, or it can use more complex techniques [18], [19]. Inbound traffic engineering is as difficult to control as with IPv4. AS-Path prepending, MED or community attributes can be used to roughly control the amount of traffic received from the providers [19]–[21].

B. IPv6 Multihoming using Cooperation between Providers

This solution relies on providers that cooperate to filter BGP routes, in order to enable route aggregation while still providing some fault-tolerance. It uses the existing routing protocols and implementations. The solution is named *IPv6 Multi-Homing with Route Aggregation*. [16], [22], but it could be used also with IPv4. Figure 4 illustrates a multihomed site that uses this mechanism. It is connected to two providers, AS 10 and AS 20. The multihomed site received a single provider-aggregatable prefix 2001:10:1::/48 from one of its providers, AS 10 in this example. This particular ISP is named the *primary ISP*.

Fig. 4.   IPv6 Multihoming through cooperation between providers

The multihomed site advertises its prefix to both AS 10 and AS 20. In order to preserve the interdomain routing, AS 20 propagates prefix 2001:10:1::/48 to AS 10 and to AS 10 only. AS 10 is able to aggregate this prefix with its own prefix 2001:10::/32, and it announces only the aggregated prefix 2001:10::/32 to the global Internet. AS 20 does not propagate prefix 200:10:1::/48 to the global Internet. This can be done for example by using BGP redistribution communities [18], [23], [24]. As a result of this routing advertisement, the traffic coming from the Internet and destined for the site is always routed through AS 10, since only AS 10 announced the prefix

of the site to the Internet. AS 10 will forward the traffic destined for its customer either directly through Link1 or via AS 20, according to some routing policy. The multihomed site can send its outbound traffic indifferently through AS 10 or AS 20.

If Link2 fails, both inbound and outbound traffic will flow through link1. Similarly, if Link1 fails, the inbound traffic will reach the multihomed site by taking the path AS 10 → AS 20 → multihomed site. The outbound traffic will take the reverse forwarding path.

Route aggregation is achieved because only the provider that assigned the PA prefix to the multihomed site aggregates and announces it to the Internet. This solution does not provide fault tolerance neither in case of a failure within the primary ISP (AS 10), nor for primary ISP Internet link failure (Link1). Additionally, if there is no direct link between AS 10 and AS 20, then the prefix 2001:10:1::/48 announced to AS 20 must be propagated to AS 10 through an intermediary transit provider. In such cases, a tunnel must be used, otherwise less aggregation is achieved and/or cooperation is needed between the transit providers, for instance through the use of the BGP Communities Attribute [25], [26]. This cooperation may conflict with their commercial interests, and may become unmanageable if the number of multihomed sites using this mechanism increases. This solution also forces the client to depend on its primary provider.

A similar solution is described in [27], where a group of providers administrates cooperatively one prefix and one ASN for their multihomed customers. Each customer is assigned a subprefix, e.g. a /48, based on the current assignment rules. However, the group of providers advertises to the global Internet only the aggregated prefix, e.g. a /32.

C. IPv6 Multihoming Support at Site Exit Router

This routing solution is based on the use of tunnels and multiple prefixes. It is described in RFC 3178 [17]. The multihomed site is assigned one prefix per provider. In the example illustrated in Figure 5, AS 65001 obtained prefix 2001:10:1::/48 from AS 10 and prefix 2001:20:1::/48 from AS 20. The two prefixes are advertised by the site exit routers RA and RB to every host inside AS 65001. These prefixes are used to derive one IPv6 address per provider for each host interface. Route aggregation is achieved by announcing to a given provider only the prefix allocated by this provider, so that each provider is able to perform route aggregation. For instance, AS 65001 advertises prefix 2001:10:1::/48 only to AS 10, and AS 10 announces only its own IPv6 aggregate 2001:10::/32 to the global Internet. Besides route aggregation, an advantage provided by this solution is that no service related to multihoming is required from the transit providers, as the site is not really multihomed, but rather single-homed to each provider.

Redundancy is provided by using secondary links, established between RA and AS 20, and between RB and AS 10. In Figure 5, RA advertises prefix 2001:20:1::/48 towards AS 20 over the secondary link, which is usually an IP-over-IP tunnel. Similarly, RB advertises prefix 2001:10:1::/48 towards AS 10
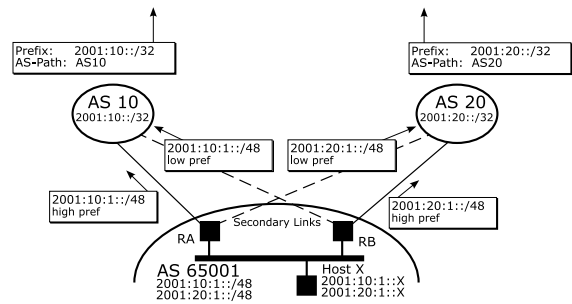

Fig. 5.   IPv6 Multihoming support at site exit router

over the secondary link. In normal conditions, secondary links are advertised by the routing protocol with a low preference, so that the primary links are used. When a failure occurs on a primary link between the site and its ISP, normal operation of the routing protocol ensures that the routing advertisement corresponding to this particular path is withdrawn. In this case, the path using the secondary link becomes a valid option for the routers.

This architecture provides route aggregation and is able to preserve the established TCP connections across link failures. The main concerns are that it does not cope with the failure of any of the upstream ISPs, and that it forces each ISP to configure tunnels. Moreover, in the case of a long-term failure, the traffic that flows through the secondary link should be switched to the primary link of the valid provider. This requires some mechanism to prevent the use of addresses belonging to the failed ISP.

IV. MIDDLE-BOX APPROACHES FOR IPV6 MULTIHOMING

Middle-Box approaches provide multihoming functionalities through services offered by intermediary boxes between multihomed hosts and the Internet, for instance a NAT box. Multihoming architectures that belong to this Middle-Box class include *Multihoming with NAT* [28], *Multihoming Aliasing Protocol* [29], and *Multihoming Translation Protocol* [30].

A. IPv6 Multihoming with NAT

IPv6 Multihoming with NAT relies on the use of Network Address Translation to direct packets towards a working provider. Typically, a NAT router is installed at the edge of the network, and knows which provider is up and which is not. Based on this knowledge, the NAT router substitutes the source IP address of an outgoing packets with an IP address belonging to the prefix of a operational provider. Figure 6 illustrates a site that is multihomed by using NAT. The site received two IPv6 prefixes, one from each of its providers. Hosts within the site may use addresses from either the first or the second provider. When a failure occurs, the source addresses contained in outgoing packets can be rewritten by the NAT box, in order to pass the filters applied by the newly selected provider.

Multihoming with NAT allows route aggregation and provides complete independence with respect to the providers. The site does not need to run BGP. The NAT box can also
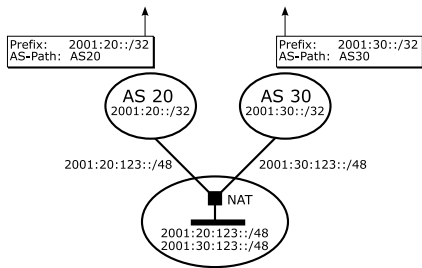
Fig. 6.   IPv6 Multihoming with NAT

be used to somewhat control the amount of traffic sent and received through each provider, as it is often done today in IPv4 [31].

However, NAT is not considered as a good engineering practice in an IPv6 Internet, because it has many architectural implications [32]. In particular, NAT alters packets and is not sufficient to achieve transport-layer survivability. Indeed, if a failure occurs that affects some connection, it is not possible to intercept and continue the connection, since the outgoing public IP addresses cannot be modified without breaking the TCP session. A solution would be to establish some cooperation with a NAT box installed in the remote site, so that the new IP address can be rewritten back to its original form when a packet reaches the destination site. This idea is used by the MHTP and MHAP solutions, described in the next section.

The use of middle boxes breaks the leading design principle for Internet protocols, called the *end-to-end principle* [33]– [35]. According to this principle, it is beneficial to limit the storage of state related to established connections to the involved end nodes. NAT breaks this principle, as state information about the ongoing connections is stored in middle-boxes along the path. The result is a reduced fault-tolerance as a these NAT boxes become single points of failure for that connections.

For all these reasons, the IETF considers this approach not suitable for IPv6 site multihoming, although it may be interesting for use as a transition mechanism, or for small residential networks.

B. Multihoming Aliasing and Translation Protocols

Multihoming Translation Protocol (MHTP) [29] and its variant *Multihoming Aliasing Protocol* (MHAP) [30] propose to set up middle-boxes at the edges of the multihomed sites. The middle-boxes, called *endpoints* use a protocol to convert provider-independent addresses to provider-aggregatable addresses when leaving the multihomed site, and convert the addresses back to the original form when reaching the edge of the destination site. Hence a fully routable and aggregatable space is used in the core of the Internet, while a provider-independent space is used at the edge of the Internet, e.g. in multihomed sites. The drawback is that an additional routing table (the MHTP routing table) must be maintained for multihomed networks. This adds an additional layer of indirection, shifting the scalability issue of site multihoming to a separate protocol. Moreover, it is difficult to ensure in an

Internet-wide environment that the addresses will be written back, and that no intermediate router will need to access the original addresses, e.g. to send an ICMP message. In addition, it is not evident that this approach can be made reasonably secure, though no security analysis was ever performed on this approach.

V. HOST-CENTRIC APPROACHES FOR IPV6 MULTIHOMING

Host-Centric Multihoming groups all solutions that rely on the use of multiple prefixes and host capabilities to provide fault-tolerance, traffic engineering and route aggregation. Many proposed solutions belong to this class : *SIM* [36], *NOID* [37], *CB64* [38], *SHIM* [39], *WIMP* [40] and *HIP* [41]. Several transport protocols (*SCTP* [42], *DCCP* [43], *Multi-homed TCP* [44], and *TCP-MH* [45]) have been adapted to operate in such an environment.

The architecture common to all Host-Centric multihoming approaches is first described in section V-A. Next, Host-Centric approaches are further divided into *Transport Layer* and *Network Layer* approaches, described respectively in sections V-B and V-C. Finally, implications on traffic engineering inside the multihomed site are outlined in section V-D. Mechanisms, advantages and drawbacks of Host-Centric solutions are summarised in table II.

A. Architecture

Figure 7 illustrates a standard IPv6 multihomed site. Two Internet Service Providers, AS 10 and AS 20, provide connectivity to the multihomed site AS 65001. AS 65001 received one prefix per provider, e.g. 2001:10:1::/48 from AS 10 and 2001:20:1::/48 from AS 20. The two prefixes are advertised by the site exit routers RA and RB to every host within AS 65001. These prefixes are used to derive one IPv6 address per provider for each host interface. Route aggregation is achieved, because AS 65001 advertises prefix 2001:10:1::/48 only to AS 10, and AS 10 only announces its own IPv6 aggregate 2001:10::/32 to the global Internet.

Fig. 7.   Host-Centric IPv6 Multihoming

1) Source Address Dependent Routing: To respect ingress filtering policies applied by the providers [14], the site must ensure that all outgoing packets with a source address within prefix 2001:10:1::/48 are sent through AS 10. Similarly, outgoing packets with a source address within prefix 2001:20:1::/48 must be sent through AS 20, otherwise they would be dropped when entering in AS 10. As a consequence, the source address selected by a host determines the upstream provider used.

Fig. 8. Source address dependent routing between the site exit routers

The simplest solution to respect this constraint is to have only one site exit router connected to all providers. This router selects the exit link on the basis of the source address contained in the packet [46]. Another solution is to implement source address dependent routing, either in the whole multi-homed sites, or simply in a connected domain that includes all the site exit routers [46], as illustrated by Figure 8. Tunnels can be used if the site exit routers are not directly interconnected.

2) Preservation of Established Connections: The difference with IPv6 Multihoming Support at Site Exit Router is that fault-tolerance is not provided by the use of backup links. Instead, it is based on enhanced host capacity to detect the failure of a path, and to switch from one provider to another. In Figure 7, suppose for instance that some failure occurs within AS 10. The host detects this failure, for example by examining the number of packet losses. To switch from the failed provider to the other provider, the host simply selects another source address for its outgoing packets. This would break the transport connections, unless some new protocol or mechanism is used by the host to preserve its established connections. This can be done in practice by using an enhanced TCP like Multi-hom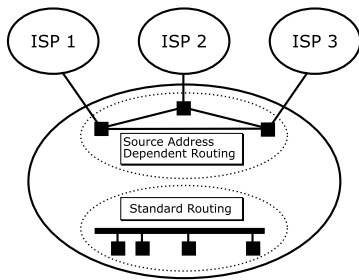ed TCP [44] or TCP-MH [45], or a dedicated transport-layer protocol like SCTP [42], [47]–[49] or DCCP [43], [50], [51], or by designing something new below the transport layer like HIP [41], SHIM [39], SIM [36], NOID [37], CB64 [38], or WIMP [40].

As applications have many different requirements for the quality of their network connection, an advantage brought by this solution is that each application can decide by itself if the connection is good enough. If needed, an application may switch to another provider by simply using another source address. The major drawback is that new mechanisms are required to preserve the established connections, in both source and destination hosts. This solution also heavily change how traffic engineering is achieved as hosts can decide which provider they use to send their outgoing traffic. This will be largely discussed in this thesis. Further issues related to this class of solutions are described in [52]. However, a quick look to Table I shows that Host-Centric approaches provide all major functionalities required, at the price of some modifications to end-hosts. This trade-off is considered acceptable, especially if the required modifications also produce benefits outside the framework of IPv6 multihoming.

B. Transport Layer Approaches

As previously stated, the mechanisms for the provision of multihoming support can be located in the transport layer or the network layer of the stack. Current transport protocols, e.g. TCP or UDP, identify the endpoints involved in a communication through their IP addresses. If one address changes, the transport-level flow breaks. With the Host-Centric solutions, hosts have several addresses. They need to be able to use them interchangeably during the lifetime of a flow in order to survive outages affecting any of those addresses. Transport Layer approaches suggest the support of multiple addresses per endpoint in the transport layer, so that an address can be substituted with another without breaking the communication. A few current transport protocols already support this, like SCTP [42] and DCCP [43]. For those protocols, only minor modifications [50], [53] are required to adapt them to the multihoming scenario. Older transport protocols, like TCP or UDP, do not currently support the use of multiple addresses, and require substantial modifications, like TCP-MH [45].

It is usually believed that the transport layer has a better understanding than the network layer of which address is working and which is not. The transport layer naturally obtains information on the quality of different paths. However, working at the transport level requires a different mechanism for every transport protocol. With respect to security, cookie based protections may be enough to ensure that no new security threats are introduced. The reason is that attacks to transport layer solutions are performed on a per connection basis, in contrast to network layer solutions.

1) TCP-MH: The existing TCP is not designed to manipulate multiple addresses in one TCP session. If a network outage occurs and the access-line associated with the local and remote IP addresses is down, the TCP session will finally time out and terminate.

An extension to TCP has been proposed in [45], where SYN segments contain all the IP addresses available to reach the source node. TCP-MH also defines MH-Add and MH-Delete options in order to convey local address information from the sender to the receiver over an established TCP connection. These options are used one at a time during a connection to add or remove usable IP addresses. When an outage is detected, the endpoints switch to another available pair of IP addresses. A serial number is added in the MH-Add and MH-Delete options so that these options are more difficult to fake for an attacker trying to hijack an existing TCP connection, but many other security issues still exist [54].

TCP-MH is not the first proposition to enhance TCP. In 1995, *Multi-homed TCP* [44] already suggested to identify packets belonging to the same connection by using a context identifier that is sent in a TCP option, rather than using the source and destination addresses and ports. Outgoing packets are sent to one or more addresses from which data has recently been received for the same connection.

2) SCTP: The Stream Control Transmission Protocol (SCTP) is a new, reliable, connection-oriented transport protocol [42], [47], [48]. It can be used as an alternative to TCP and UDP. A relationship is created and maintained between two endpoints of an SCTP association until all packets have been

successfully transmitted. SCTP allows data to be partitioned into multiple streams that have the property of independently sequenced delivery, so that a message lost in any one stream will only initially affect delivery within that stream, and not delivery in other streams.

A core feature of SCTP is the ability for a SCTP endpoint to support multiple IP addresses. SCTP endpoints exchange their lists of addresses during the initiation of an association. No IP address can be added or deleted once the association has been established, although an extension to SCTP can provide this feature [55]. Each endpoint is able to receive messages from any of the addresses associated with the remote endpoint. However, a single address is chosen as the *primary* address and is used as the destination for normal transmission. Each endpoint monitors the reachability of the secondary addresses of its peer so that it always knows which addresses are available for the failover. The monitoring is done by sending a heartbeat packet to an idle destination address, which the peer acknowledges. A secondary address is used when continued failure to send to the primary address is noticed, until heartbeat packets determine that the primary address is reachable again.

Issues and discussion on the applicability of SCTP to the multihoming problem are presented in [49], [53]. From the security viewpoint, SCTP uses a random verification tag as a weak security mechanism to avoid packet injection. SCTP protects itself against TCP SYN flooding attacks by remaining stateless during the handshake in order to prevent state-exhaustion attacks. Security issues are discussed in details in [54].

3) DCCP: The Datagram Congestion Control Protocol (DCCP) [43], [51] did not originally support multihoming. An extension to DCCP [50] provides primitive support for multihoming and mobility via a mechanism for transferring a connection endpoint from one address to another. The moving endpoint must negotiate this support beforehand. When the moving endpoint gets a new address, it sends a DCCP-Move packet from that address to the stationary endpoint. Next, the stationary endpoint changes its connection state to use the new address. DCCP support for mobility is intended to solve only the simplest multihoming and mobility problems; for instance, there is no support for simultaneous moves.

C. Network Layer Approaches

Network Layer approaches suggest to support multiple addresses in an intermediate layer between the transport and the network layers. More precisely, this intermediate layer is located above the IP routing sub-layer (that performs network related functions like forwarding), but below the IP endpoint sub-layer (that performs end-to-end functions like fragmenting and IPsec). The whole protocol stack, including the new layer, is depicted in Figure 9 [56]. This new layer 3.5 aims at separating two entirely separate functions that are included in an IP address : the location of a node, and the identity of the node. The *locator* of a node specifies how to reach the node. It specifies a network attachment point, in term of the network topology. A locator is mostly used to forward packets in routers. The *identifier* of a node is a label at the

IP layer, which is presented to upper layers. This is illustrated on Figure 9. An identifier is used for distinguishing one node from another and is independent from the node's attachment to the network. A node can have multiple identifiers, but each identifier is supposedly globally unique.

Current IP addresses are both locators and identifiers, because they contain topological significance and act as a unique identifier for an interface. Identifiers form a new intermediate namespace between the two currently global namespaces that the Internet has created : Internet Protocol (IP) addresses and Domain Name Service (DNS) names.

The separation between locators and identifiers allows applications to only use identifiers, which are mapped to locators by the intermediate layer. When a locator is no longer valid, the identifier is mapped to another locator. This change of locator is transparent for the upper layers, since applications and transport protocols bind only to the identifier, which never changes. Consequently, this approach is available for any transport protocol, including the installed base of TCP. However, endpoints using this approach require additional mechanisms to coherently map the identifiers presented to the upper layers and the IP addresses actually contained in the packets. This mapping between identifiers and locators may be vulnerable to redirection attacks if no proper protection is provided [56], [57]. Such a vulnerability is introduced when an attacker can benefit from the mapping mechanism to induce a victim to believe that she is communicating with the owner of a given identifier, while she is actually exchanging packets with a locator that does not belong to the owner of the perceived identifier. In other words, a redirection attack consists in creating a false mapping between an identifier and a locator.

Several Intermediate Network Layer approaches have been proposed, differing from the kind of identifier namespace. Some proposals suggest the creation of a new identifier namespace, of cryptographical nature or not. Others promote the use of regular IP addresses as identifiers. In addition, hybrid proposals suggest the use of cryptographical addresses as identifiers. We will now present each one of them.

1) New Cryptographic Identifier Namespace: Several proposals suggest the creation of a new cryptographic identifier namespace. Among those, we can find the *Host Identity Protocol* (HIP) approach to multihoming [41], [58]–[60], and the *Strong Identify Multihoming* [36] (SIM). Both HIP and SIM implement the separation between identifiers and locators by defining a separate name space and a new layer between the network and transport layers. This new structure insulates the transport layer protocols from the networking layer, thereby allowing transport sessions to remain unaffected even if the underlying IP addresses change. They both propose the creation of a new 128-bit identifier as the cryptographic hash of a public key associated to the endpoint. The public key is typically stored in the DNS, using a new DNS Resource Record.

The result is a secure binding between the identifier and the associated key pair. This allows the node to use the corresponding private key to sign the control packets that convey alternative address information. The trust chain is
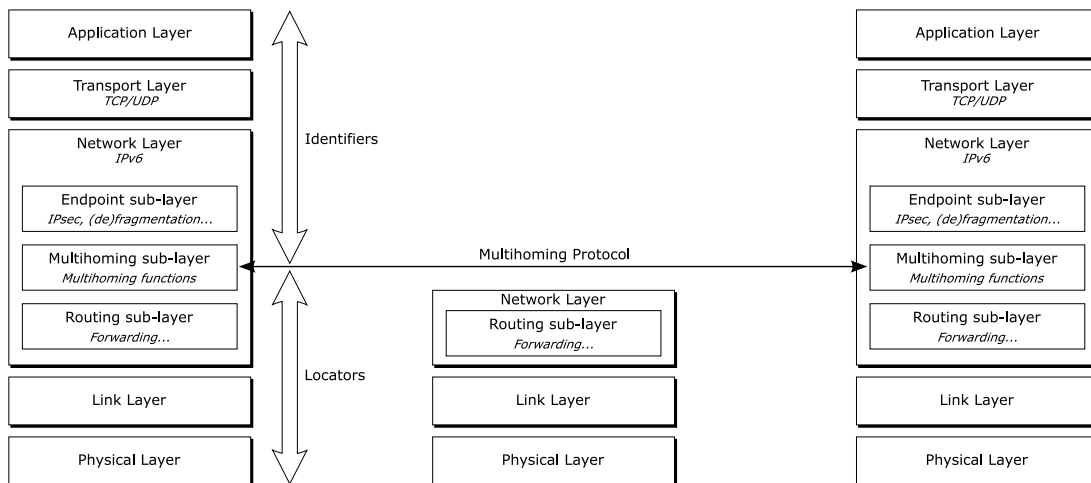
Fig. 9. Multihoming layer in the protocol stack

the following : the identifier used for the communication is securely bound to the key pair because it contains the hash of the public key, and the alternative address is bound to the public key through the signature. This approach effectively protects against redirection attacks. Additional protection against flooding attacks is obtained through a reachability test before actually sending packets to the alternative locators.

A main difficulty identified for this approach is the high cost of public key operations. HIP is a four-way handshake, requiring public key cryptographic operations. It provides protection from Man-In-the-Middle attacks. While such a heavy exchange makes sense for applications where hosts have a fairly long term relationship, it may be too heavy for short term transactions, such as web browsing, where such protection is not required. SIM limits the use of public key signatures only to the time of locator prefix changes for a host or when two hosts claim to use the same identifier. Another difficulty is the support for referrals and call-backs when embedding the identifiers into applications protocols [61]. This last problem is related to the extreme difficulty of building a directory service that maps identifiers to locators when the identifier namespace is flat.

2) New Ephemeral Identifier Namespace: The protection of the identifier is important because it represents the identity of the owner. A possible approach to avoid the security issues is to simply remove this functionality from the identifier. The *Weak Identifier Multihoming Protocol* (WIMP) [40] proposes to use ephemeral 128-bit identifiers, which are only valid as long as the flow is active. Once the flow is over, the identifier is meaningless, hence worthless. So there is no need to protect it. However, this is only possible for the identifier of the source endpoint, but not for the identifier of the destination endpoint. Indeed, in order to be able to establish a communication with a given target, a stable identifier of the target is required. The stable identifier proposed by WIMP is a hash of the Fully Qualified Domain Name (FQDN). The result is that source identifiers are worthless, so there is no need to protect them, while the destination identifiers are intrinsically bound to the FQDN of the target, providing the required protection.

During WIMP session establishment, WIMP introduces a separate header into the data packets, between the IP and TCP/UDP headers that contains information about the WIMP session. WIMP does not introduce a separate header into all IPv6 packets. Instead, once a WIMP session is established, the IPv6 FlowID is used to hold an identifier for the WIMP host-pair context associated with a given packet. The FlowIDs serve as a convenient *compression tag* without increasing the packet size.

In order to prevent redirection attacks WIMP relies on the ability to verify that the entity requesting redirection indeed holds the successor values of a hash chain and is able to combine a divided secret that is sent via parallel paths. WIMP can be divided into two exchanges : context establishment and re-addressing exchange. The former exchange establishes a state for both communication end-points. The re-addressing exchange is used to update the locators belonging to the communicating parties.

The main difficulty with the WIMP approach is that referrals and call-backs are not supported, due to the ephemeral nature of the identifier. Another issue with this approach is that it depends on the DNS system.

3) Plain IPv6 Addresses as Identifiers: HIP, SIM and WIMP propose to create a new namespace, completely separated from the locators namespace. Instead, *Multihoming without IP Identifiers* (NOID) [37] suggests that the identifier of a host may be chosen from its set of regular IPv6 addresses. Since the address used as an identifier by the upper layers is not intrinsically bound to a public key, an external trusted entity is required to secure the binding between the identifier and the locators. NOID relies on the DNS infrastructure to verify the relationship between a given locator, the corresponding FQDN and the set of locators for the host. More precisely, the initiating node uses a DNS query to obtain all the available addresses of the target node. Next, the initiator selects an identifier among the obtained addresses. The remaining addresses are used as alternative locators to establish the flow. The target node obtains the set of IPv6 addresses available for the initiator by querying the reverse

DNS tree. NOID makes use of flow IDs so that mapping to the correct identifier at the receiving end can be accomplished, without relying on the locators in the packet.

The main difficulty of this approach is the dependence on the DNS, especially on the proper population of the reverse DNS tree, which may be difficult to achieve for unmanaged networks.

4) Hybrid Approaches: Addresses with Cryptographic Features as Identifiers: Hybrid approaches attempt to achieve the benefits of the previous approaches without their limitations. They propose the use of addresses as identifiers, as in NOID, in order to properly support referrals and call-backs. However, the addresses used as identifiers are not regular addresses. Instead, they contain cryptographical information in the interface identifier part, providing a secure binding between the identifier and the alternative locators. One approach is *Multihoming using 64-bit Crypto-Based IDs* (CB64) [38], where the address of a multihomed node is a Cryptographic Generated Address (CGA) [62], [63], that contains a 64-bit hash of a public key in its interface identifier. In this case, the set of alternative locators can be authenticated through a signature with the corresponding private key.

In order to prevent redirection attacks, this protocol relies on the ability to verify, using public key cryptography as in SIM, that the entity requesting redirection indeed holds the private key where the hash of the corresponding public key hashes to the ID itself. Hence, CB64 does not use DNS for verification as in NOID. However, the cost of those public key operations involved is a limitation for CB64.

An alternative approach is based on the use of Hash Based Addresses [64]. *Multihoming L3 Shim Approach* (SHIM) [39], [56] suggests that information about the multiple prefixes is included within the addresses themselves. This is achieved by generating the interface identifiers of the addresses of a host as hashes of the available prefixes and a random number. The multiple addresses are next generated by appending the different network prefixes to the generated interface identifiers. The result is a set of addresses, called Hash Based Addresses (HBAs), that are inherently bound. A cost efficient mechanism is available to determine if two addresses belong to the same set : given the prefix set and the additional parameters used to generate the HBA, a single hash operation is enough to verify if an HBA belongs to a given HBA set. No public key operations are involved in the verification process, as long as the prefix set is stable. Protection against flooding is obtained through a reachability test that verifies the willingness of the target to receive traffic through the alternative locators. An incremental approach to IPv6 multihoming, that uses the SHIM approach, is described in [65]. We will futher detail the SHIM approach in Section VII, as this is the solution chosen by the IETF.

D. Implications on Traffic Engineering

The use of multiple addresses introduces a new architectural approach to engineer the traffic. As explained in section V-A, the source address selected by a host determines the upstream provider used. Hence, in order to control its outgoing traffic, the multihomed site must instruct its hosts how they should select their source addresses. This traffic engineering approach is said to be based on host capability.

In practice, hosts use the source address selection algorithm described in [66] to select an appropriate address. The selection relies on a *policy table*, that can be filled with additional rules. However, the default source address selection is arbitrary when the host has several global-scope IPv6 addresses as in Host-Centric solutions. Load sharing can be achieved by filling in the policy tables of the hosts, either in a fully dynamic fashion, as proposed in [67], or more or less statically by using an enhanced DHCP.

VI. Looking Towards the Future of IPv6 Multihoming

During the last years, the IETF has made several explicit or implicit architectural decisions regarding IPv6 multihoming. The main decision is to go down the path of developing the Host-Centric approaches. The IETF made this choice in 2003. It can be explained by looking at Table I.

This table summarises the mechanisms, advantages and drawbacks of the three approaches to IPv6 Multihoming : Routing, Middle-Box, and Host-Centric approaches. By taking a look at the features provided by each approach, we can observe that Routing approaches either do not allow route aggregation, or cannot provide complete fault-tolerance. Moreover, Routing approaches typically require running BGP, and do not provide Site-ISP independence. Middle-Box approaches provide many required features, but basically fail to preserve the End-to-End Principle [33]–[35]. Moreover, IPv6 multihoming with NAT does not preserve the transport-layer flows in case of failure, and MHAP or MHTP shows security concerns. Hence, the Middle-Box approach is not considered as a suitable solution for IPv6 site multihoming, although it may still be interesting as a transition mechanism, or for small residential networks.

In Table I, Host-Centric approaches appear to be the most promising IPv6 multihoming architectures, provided that functionalities are added to end hosts. This is considered acceptable as many hosts already require to be updated for the new IPv6 Internet, and because Routing approaches for IPv6 multihoming can provide transition mechanisms. As a consequence of this decision, it can be expected that the use of several IPv6 addresses on each end-host will become widely prevalent on the IPv6 Internet. This is a drastic architectural change compared to today's IPv4 Internet, where hosts are typically identified by a single IPv4 address.

Table II summarises the mechanisms, advantages and drawbacks of the main Host-Centric solutions. The Host-Centric architecture requires that end hosts be able to switch between IPv6 addresses without breaking transport-level flows. Many Host-Centric solutions were proposed. Among all proposed mechanisms, the IETF promotes Intermediate Network Layer approaches, as they can provide multihoming support to any transport level protocol such as TCP and UDP. Moreover, Transport Layer approaches appear to have some difficulties

	Routing Class			Middle-Box Class		Host-Centric Class
	Multihoming with BGP	Provider Cooperation	RFC 3178	Multihoming with NAT	MHAP, MHTP	Host-Centric*
Mechanism IP support	IPv4+IPv6	IPv4+IPv6	IPv4+IPv6	IPv4	IPv6	IPv6
Fault-tolerance	R	R	R+TUN	MB	MB	H
Traffic engineering capability	R	R	R	MB	MB	H
Route aggregation	N/A	SA	MP	MP	MP	MP
Complete Independence	N/A	N/A	N/A	MP	MP	MP
Feature Route aggregation		✓	✓	✓	✓	✓
Scalability			✓	✓		✓
Traffic engineering capability	✓	✓	✓	✓	✓	✓
Link fault tolerance	✓	✓	✓	✓	✓	✓
ISP fault tolerance	✓			✓	✓	✓
Transport-layer survivability	✓	✓	✓		✓	✓
Stable configuration in case of long term failure	✓			✓	✓	✓
Site - ISP Independence				✓	✓	✓
ISP - ISP Independence	✓		✓	✓	✓	✓
No modification to hosts	✓	✓	✓	✓	✓	

R = Based on routing system, H = Based on host capability, MP = Based on the use of multiple prefixes, MB = Based on the use of a middle-box, SA = Based on selective route announcements, TUN = Based on the use of tunnels, N/A = Not available, ✓ = provides this feature. * Groups many Host-Centric solutions.

TABLE I

OVERVIEW OF IPV6 MULTIHOMING APPROACHES

	Transport Layer Approaches			Network Layer Approaches					
	SCTP	TCP-MH	DCCP	NOID	CB64	SHIM	SIM	HIP	WIMP
Mechanism IP version support	4+6	4+6	4+6	4+6	6	6	6	4+6	6
Layer	4	4	4	3.5	3.5	3.5	3.5	3.5	3.5
ULID namespace	IP	IP	IP	IP	IP+	IP+	ID	ID	ID
Layer-4 applicability	TCP+UDP	TCP	UDP	ANY	ANY	ANY	ANY	ANY	ANY
Secure change of locators	PK/key	Cookie	PK/key	DNS	PK	HBA/PK	PK	PK	DNS/Ephemeral
Feature DNS independence	✓	✓	✓		✓	✓	✓	✓	
Allows referrals	✓	✓	✓	✓	✓	✓			
No new identifier namespace	✓	✓	✓	✓	✓	✓			
Protection against redirection	✓	✓	✓	✓	✓	✓	✓	✓	✓

IP = plain IP address space, ID = new IP address space, IP+ = plain IP with modified interface identifier PK = uses Public Key cryptography, key = uses symmetric key operations, DNS = uses DNS, Ephemeral = security provided through the use of ephemeral identifiers, HBA = uses Hash Based Addresses. ✓ = provides this feature.

TABLE II

HOST-CENTRIC APPROACHES: OVERVIEW OF MECHANISMS FOR THE PROVISION OF IPV6 MULTIHOMING

for protecting against *Man-In-The-Middle* (MITM) attacks. Among the Intermediate Network Layer approaches, SIM, HIP, WIMP, LIN6 and E2E use a new identifier namespace, which creates concerns for supporting referrals and call-backs when embedding the identifiers into applications protocols [61]. A look at this table shows that SHIM and CB64 are the most promising solutions, due to their security features, and their low requirements on the infrastructure.

SHIM can be seen as a superset of CB64, since SHIM supports multiple security mechanisms. In particular, SHIM can provide a more efficient support when the set of prefixes allocated to a multihomed site is stable. For this reason, the IETF has decided by the end of 2004 to foster the SHIM approach.

VII. THE SHIM APPROACH

As explained in the previous sections, the *Multihoming L3 Shim Approach* (SHIM) [39], [56] proposes the use of an intermediate layer located above the IP routing sub-layer, but below the IP endpoint sub-layer. This approach uses, at least initially, routable IP locators as the identifiers visible above the SHIM layer. This ensures that all upper layer protocols can operate unmodified in a multihoming environment, as they always see a stable IPv6 address. The locator used in the address fields of the packets can change over time in response to failures affecting the original locator. This is illustrated

on Figure 10. In this figure, Host X has addresses IP1(X) and IP2(X), and Host Y has addresses IP1(Y), IP2(Y) and IP3(Y). The stable source and destination addresses seen by transport and upper layers are IP1(X) and IP2(Y), while the actual addresses used in the packets are IP2(X) and IP3(Y). The mapping between the stable and actual addresses is done by the new SHIM layer.
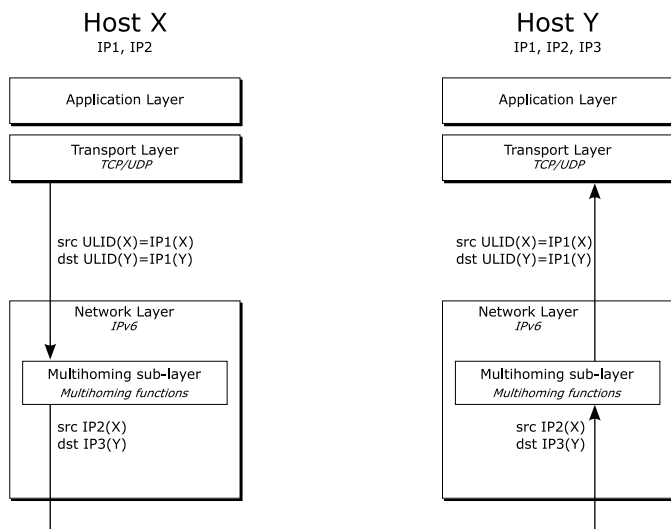

Fig. 10. Mapping with changed locators

The SHIM approach is best explained by describing the sequence of events that occur when a multihomed capable Host X starts talking to another multihomed capable Host Y.

When Host X wants to initiate a communication with Host Y, it first typically issues a DNS request for a name of Host Y. It receives in the DNS response some or all the addresses assigned to Host Y. Host X uses the default address selection algorithm [66] to select both the source and destination addresses that will be used for its outgoing packets. These initial source and destination addresses will also be used as endpoint identifier for all transport and application layers on Host X and Host Y. So far, no multihoming protocol exchange is needed.

At some point in time, one of the host, e.g. Host X, may request to take advantage of multihoming, e.g. in order to obtain a higher reliability. Hence, it initiates the SHIM protocol exchange. This exchange will fail if Host Y does not support the SHIM protocol. If it succeeds, Hosts X and Y will exchange their respective set of available addresses. In order to prevent redirection attacks, Host X uses the HBA mechanism [64] described in Section V-C.4 to ensure that the additional addresses given by Host Y are compatible with the initial address of Host Y, i.e. that all addresses of Y belong to the same HBA set. At this point in time it is possible for both hosts to change to a different address in the set.

Suppose that a failure of a provider prevents Host Y from receiving packets from Host X. A timeout is raised on Host Y, and a reachability test packet is sent to Host X to check if the path is still available. If no answer is received, Host Y initiates an address pair exploration procedure by sending several test packets to Host X with different source and destination addresses, until a reply packet is received. When Host X receives packets from Host Y with new addresses, it also checks the currently used addresses, and switches to a new address pair if needed. It is not required that both Host X and Host Y use the same address pair for communicating. This address exploration procedure is explained in [68], [69], and is still being discussed at the IETF.

VIII. CONCLUSION

Constraints for the scalability of the Interdomain routing system have led the IETF and the research community to propose drastic changes in how site multihoming shall be achieved in an IPv6 Internet. This paper has reviewed and compared all major architectures that were proposed for IPv6 site multihoming, their mechanisms, advantages and concerns. Host-Centric multihoming, the approach promoted by the IETF for IPv6 multihoming, introduces fundamental changes to the behaviour of hosts within the multihomed site. It also affects the intradomain routing and traffic engineering mechanisms In particular, the use of several IPv6 addresses per end hosts introduces a major architectural change compared to today's IPv4 Internet, where hosts are typically identified by a single IPv4 address. Fortunately, these changes also bring wide opportunities to develop multihoming for small and residential networks.

IX. BIOGRAPHIES

Cédric de Launois obtained his degree in computer science and engineering in 2001 from Université catholique de Louvain (UCL), Belgium. In 2005, he gained his Ph.D. degree from the same university. His main research interests include IPv6 multihoming and traffic engineering.

Marcelo Bagnulo received his Electrical Engineering degree from Universidad de la Republica Oriental del Uruguay in 1999, and the Doctor degree in 2003 from the University Carlos III de Madrid - Spain. He is currently working as a Research and Teaching Assistant at Universidad Carlos III de Madrid. He is involved in the design of the SHIM6 protocol, the IPv6 site multihoming solution currently being defined at the IETF.

REFERENCES

[1] G. Huston, "BGP Routing Table Analysis Reports," http://bgp.potaroo. net, May 2004.

[2] S. Agarwal, C.-N. Chuah, and R. H. Katz, "OPCA: Robust interdomain policy routing and traffic control," in *Proceedings of OPENARCH*, 2003.

[3] G. Huston, "Commentary on Inter-Domain Routing in the Internet," RFC 3221, IETF, December 2001.

[4] R. Atkinson and S. Floyd, "IAB Concerns and Recommendations Regarding Internet Research and Evolution," RFC 3869, IETF, August 2004.

[5] G. Huston, "Architectural Approaches to Multi-Homing for IPv6," Internet Draft, IETF, October 2004, <draft-ietf-multi6-architecture-02.txt>, work in progress.

[6] RIPE NCC, "Smallest RIPE NCC Allocation / Assignment Sizes," Document ID: ripe-345, https://www.ripe.net/ripe/docs/smallest-alloc-sizes. html, April 2005.

[7] ARIN, "IP Address Space Allocated to ARIN," https://www.arin.net/ reference/ip_blocks.html, May 2005.

[8] APNIC, "Allocation sizes within APNIC IPv4 address ranges," https: //www.apnic.net/db/min-alloc.html, May 2005.

[9] G. Huston, "Analyzing the Internet BGP routing table," *Internet Protocol Journal*, March 2001.

[10] ——, "Commentary on Inter-Domain Routing in the Internet," RFC 3221, IETF, December 2001.

[11] R. Atkinson and S. Floyd, "IAB Concerns and Recommendations Regarding Internet Research and Evolution," RFC 3869, IETF, August 2004.

[12] J. Abley, B. Black, and V. Gill, "Goals for IPv6 Site-Multihoming Architectures," RFC 3582, IETF, August 2003.

[13] P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," BCP 38, RFC 2827, IETF, May 2000.

[14] F. Baker and P. Savola, "Ingress Filtering for Multihomed Networks," BCP 84, IETF, March 2004.

[15] K. Lindqvist, "Multihoming in IPv6 by multiple announcements of longer prefixes," Internet Draft, IETF, December 2002, <draft-kurtis-multihoming-longprefix-00.txt>, work in progress.

[16] J. Jieyun, "IPv6 Multi-Homing with Route Aggregation," Internet Draft, IETF, August 2000, <draft-ietf-ipngwg-ipv6multihome-with-aggr-01.txt>, work in progress.

[17] J. Hagino and H. Snyder, "IPv6 Multihoming Support at Site Exit Routers," RFC 3178, IETF, October 2001.

[18] B. Quoitin, S. Uhlig, C. Pelsser, L. Swinnen, and O. Bonaventure, "Interdomain traffic engineering with BGP," *IEEE Communications Magazine*, May 2003.

[19] S. Uhlig, "Implications of Traffic Characteristics on Interdomain Traffic Engineering," Ph.D. dissertation, Université catholique de Louvain, March 2004.

[20] O. Bonaventure, P. Trimintzios, G. Pavlou, B. Quoitin, A. Azcorra, M. Bagnulo, P. Flegkas, A. García-Martínez, P. Georgatsos, L. Georgiadis, C. Jacquenet, L. Swinnen, S. Tandel, and S. Uhlig, *Quality of Future Internet Services, Cost263 final report*, ser. LNCS. Springer-Verlag, September 2003, no. 2856, ch. Internet Traffic Engineering, pp. 118–179.

[21] B. Quoitin, C. Pelsser, O. Bonaventure, and S. Uhlig, "A performance evaluation of BGP-based traffic engineering," *International Journal of Network Management (Wiley)*, vol. vol. 15, no. 3, May-June 2004.

[22] M. Bagnulo, A. García-Martínez, A. Azcorra, and D. Larrabeiti, "Survey on proposed IPv6 multi-homing network level mechanisms," Internet Draft, July 2001, <draft-bagnulo-multi6-survey6-00.txt>.

[23] B. Quoitin, S. Tandel, S. Uhlig, and O. Bonaventure, "Interdomain traffic engineering with Redistribution Communities," *Computer Communications Journal*, vol. vol. 27, no. 4, pp. 355–363, October 2003.

[24] B. Quoitin, S. Uhlig, and O. Bonaventure, "Using redistribution communities for interdomain traffic engineering," in *3d COST 263 International Workshop on Quality of Future Internet Services (QoFIS 2002)*, vol. LNCS 2511. Springer-Verlag, October 2002.

[25] P. Traina, R. Chandrasekeran, and T. Li, "BGP Communities Attribute," RFC 1997, IETF, August 1996.

[26] O. Bonaventure and B. Quoitin, "Common utilizations of the BGP community attribute," Internet Draft, IETF, June 2003, <draft-bq-bgp-communities-00.txt>, work in progress (expired).

[27] K. Toyama and T. Fujisaki, "Operational Approach to achieve IPv6 multihomed network," Internet Draft, IETF, February 2004, <draft-toyama-multi6-operational-site-multihoming-00.txt>, work in progress (expired).

[28] P. Akkiraju and Y. Rekhter, "A Multihoming solution using NATs," Internet Draft, November 1998, <draft-akkiraju-nat-multihoming-00.txt>, work in progress.

[29] M. Py, "Multi Homing Translation Protocol (MHTP)," Internet Draft, IETF, November 2001, <draft-py-multi6-mhtp-01.txt>, work in progress.

[30] ——, "Multi Homing Aliasing Protocol (MHAP) intro," Internet Draft, IETF, March 2003, <draft-py-mhap-intro-00.txt>, work in progress.

[31] D. Allen, "NPN: Multihoming and Route Optimization: Finding the Best Way Home," *Network Magazine*, February 2002.

[32] T. Hain, "Architectural Implications of NAT," RFC 2993, IETF, November 2000.

[33] R. Bush and D. Meyer, "Some Internet Architectural Guidelines and Philosophy," RFC 3439, IETF, December 2002.

[34] B. Carpenter, "Internet Transparency," RFC 2775, IETF, February 2000.

[35] J. H. Saltzer, D. P. Reed, and D. D. Clark, "End-To-End Arguments in System Design," *ACM Transactions on Computer Systems*, vol. vol. 2, no. 4, pp. 277–288, November 1984.

[36] E. Nordmark, "Strong Identity Multihoming using 128 bit Identifiers (SIM/CBID128)," Internet Draft, IETF, October 2003, <draft-nordmark-multi6-sim-01.txt>, work in progress.

[37] ——, "Multihoming without IP Identifiers," Internet Draft, IETF, July 2004, <draft-nordmark-multi6-noid-02.txt>, work in progress.

[38] ——, "Multihoming using 64-bit Crypto-based IDs," Internet Draft, IETF, October 2003, <draft-nordmark-multi6-cb64-00.txt>, work in progress.

[39] E. Nordmark and M. Bagnulo, "Multihoming L3 Shim Approach," Internet Draft, IETF, January 2005, <draft-ietf-multi6-l3shim-00.txt>, work in progress.

[40] J. Ylitalo, V. Torvinen, and E. Nordmark, "Weak Identifier Multihoming Protocol (WIMP)," Internet Draft, IETF, January 2004, <draft-ylitalo-multi6-wimp-00.txt>, work in progress.

[41] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host Identity Protocol," Internet Draft, IETF, February 2005, <draft-ietf-hip-base-02.txt>, work in progress.

[42] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson, "Stream Control Transmission Protocol," RFC 2960, IETF, October 2000.

[43] E. Kohler, M. Handley, and S. Floyd, "Datagram Congestion Control Protocol (DCCP)," Internet Draft, IETF, November 2004, <draft-ietf-dccp-spec-09.txt>, work in progress.

[44] C. Huitema, "Multi-homed TCP," Internet Draft, IETF, May 1995, <draft-huitema-multi-homed-01.txt>, work in progress (expired).

[45] A. Matsumoto, M. Kozuka, and K. Fujikawa, "TCP Multi-Home Options," Internet Draft, IETF, October 2003, <draft-arifumi-tcp-mh-00.txt>, work in progress.

[46] C. Huitema, R. Draves, and M. Bagnulo, "Ingress filtering compatibility for IPv6 multihomed sites," Internet Draft, IETF, October 2004, <draft-huitema-multi6-ingress-filtering-00.txt>, work in progress (expired).

[47] L. Coene, "Stream Control Transmission Protocol Applicability Statement," RFC 3257, IETF, April 2002.

[48] L. Ong and J. Yoakum, "An Introduction to the Stream Control Transmission Protocol (SCTP)," RFC 3286, IETF, May 2002.

[49] L. Coene, "Multihoming issues in the Stream Control Transmission Protocol," Internet Draft, May 2002, <draft-coene-sctp-multihome-04.txt>, work in progress (expired).

[50] E. Kohler, "Datagram Congestion Control Protocol Mobility and Multihoming," Internet Draft, IETF, July 2004, <draft-kohler-dccp-mobility-00.txt>, work in progress.

[51] S. Floyd, M. Handley, and E. Kohler, "Problem Statement for DCCP," Internet Draft, IETF, October 2002, <draft-ietf-dccp-problem-00.txt>, work in progress (expired).

[52] C. Huitema, R. Draves, and M. Bagnulo, "Host-Centric IPv6 Multihoming," Internet Draft, February 2004, <draft-huitema-multi6-hosts-03.txt>, work in progress.

[53] L. Coene and J. Loughney, "Multihoming: the SCTP solution," Internet Draft, IETF, January 2004, <draft-coene-multi6-sctp-00.txt>, work in progress.

[54] T. Aura, P. Nikander, and G. Camarillo, "Effects of Mobility and Multihoming on Transport-Protocol Security," in *IEEE Symposium on Security and Privacy*, Berkeley, California, May 2004.

[55] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, and P. Conrad, "Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration," Internet Draft, IETF, February 2005, <draft-ietf-tsvwg-addip-sctp-11.txt>, work in progress.

[56] M. Bagnulo, A. García-Martínez, and A. Azcorra, "Efficient Security for IPv6 Multihoming," *ACM SIGCOMM Computer Communications Review*, vol. vol. 35, no. 5, April 2005.

[57] E. Nordmark and T. Li, "Threats relating to IPv6 multihoming solutions," Internet Draft, IETF, January 2005, <draft-ietf-multi6-multihoming-threats-03.txt>, work in progress.

[58] R. Moskowitz and P. Nikander, "Host Identity Protocol Architecture," Internet Draft, IETF, June 2004, <draft-moskowitz-hip-arch-06.txt>, work in progress (expired).

[59] P. Nikander, "Considerations on HIP based IPv6 multi-homing," Internet Draft, IETF, December 2003, <draft-nikander-multi6-hip-00.txt>, work in progress (expired).

[60] P. Nikander, J. Arkko, and T. Henderson, "End-Host Mobility and Multi-Homing with Host Identity Protocol," Internet Draft, IETF, February 2005, <draft-ietf-hip-mm-01.txt>, work in progress.

[61] E. Nordmark, "Multi6 Application Referral Issues," Internet Draft, IETF, January 2005, <draft-ietf-multi6-app-refer-00.txt>, work in progress.

[62] T. Aura, "Cryptographically Generated Addresses (CGA)," RFC 3972, IETF, March 2005.

[63] G. O'Shea and M. Roe, "Child-proof Authentication for MIPv6 (CAM)," *ACM Computer Communications Review*, vol. vol. 31, no. 2, April 2001.

[64] M. Bagnulo, "Hash Based Addresses (HBA)," Internet Draft, IETF, December 2004, <draft-ietf-multi6-hba-00.txt>, work in progress.

[65] M. Bagnulo, A. García-Martínez, A. Azcorra, and C. de Launois, "An Incremental Approach to IPv6 Multihoming," *To appear in Computer Communications*, 2005.

[66] R. Draves, "Default Address Selection for Internet Protocol version 6 (IPv6)," RFC 3484, IETF, February 2003.

[67] C. de Launois and O. Bonaventure, "NAROS : Host-Centric IPv6 Multihoming with Traffic Engineering," Internet Draft, May 2003, <draft-de-launois-multi6-naros-00.txt>, work in progress.

[68] J. Arkko, "Failure Detection and Locator Selection Design Considerations," Internet Draft, IETF, January 2005, <draft-ietf-shim6-failure-detection-00>, work in progress.

[69] I. van Beijnum, "Shim6 Reachability Detection," Internet Draft, IETF, July 2005, <draft-ietf-shim6-reach-detect-00.txt>, work in progress.

Prefix: 20.0.0.0/8
AS-Path: AS20

Prefix: 123.0.0.0/8
AS-Path: AS20 AS123

Prefix: 30.0.0.0/8
AS-Path: AS30

Prefix: 123.0.0.0/8
AS-Path: AS30 AS123

AS 20
20.0.0.0/8

AS 30
30.0.0.0/8

Prefix: 123.0.0.0/8
AS-Path: AS123

Prefix: 123.0.0.0/8
AS-Path: AS123

AS 123

123.0.0.0/8

Prefix: 30.0.0.0/8
AS-Path: AS30

Prefix: 20.0.123.0/24
AS-Path: AS30 AS123

Prefix: 20.0.0.0/8
AS-Path: {AS20,AS123}

AS 20
20.0.0.0/8

AS 30
30.0.0.0/8

Prefix: 20.0.123.0/24
AS-Path: AS123

Prefix: 20.0.123.0/24
AS-Path: AS123

AS 123

20.0.123.0/24

Prefix: 2001:20::/32
AS-Path: {AS20, AS123}

Prefix: 2001:30::/32
AS-Path: AS30

Prefix: 2001:20:123::/48
AS-Path: AS30 AS123

AS 20
2001:20::/32

AS 30
2001:30::/32

Prefix: 2001:20:123::/48
AS-Path: AS123

Prefix: 2001:20:123::/48
AS-Path: AS123

AS 123

2001:20:123::/48

Prefix: 2001:10::/32
AS-Path: {AS10, AS123}

Prefix: 2001:10:1::/48
AS-Path: AS20 AS123

Prefix: 2001:20::/32
AS-Path: AS20

Primary ISP

AS 10
2001:10::/32

AS 20
2001:20::/32

Prefix: 2001:10:1::/48
AS-Path: AS123

Prefix: 2001:10:1::/48
AS-Path: AS123

Link1

Link2

AS 123

2001:10:1::/48

Prefix: 2001:10::/32
AS-Path: AS10

Prefix: 2001:20::/32
AS-Path: AS20

AS 10
2001:10::/32

AS 20
2001:20::/32

2001:10:1::/48
low pref

2001:20:1::/48
low pref

2001:10:1::/48
high pref

2001:20:1::/48
high pref

Secondary Links

RA

RB

AS 65001
2001:10:1::/48
2001:20:1::/48

Host X
2001:10:1::X
2001:20:1::X

```
Prefix:    2001:20::/32
AS-Path:  AS20
```

```
Prefix:    2001:30::/32
AS-Path:  AS30
```

AS 20
2001:20::/32

AS 30
2001:30::/32

2001:20:123::/48

2001:30:123::/48

NAT

2001:20:123::/48
2001:30:123::/48

```
Prefix:    2001:10::/32
AS-Path:  AS10
```

```
Prefix:    2001:20::/32
AS-Path:  AS20
```

AS 10
2001:10::/32

AS 20
2001:20::/32

```
Prefix:    2001:10:1::/48
AS-Path:  AS65001
```

```
Prefix:    2001:20:1::/48
AS-Path:  AS65001
```

RA   RB

AS 65001
2001:10:1::/48
2001:20:1::/48

Host X
2001:10:1::X
2001:20:1::X

ISP 1   ISP 2   ISP 3

Source Address
Dependent Routing

Standard Routing

Application Layer		Application Layer
Transport Layer *TCP/UDP*		Transport Layer *TCP/UDP*
Network Layer *IPv6* Endpoint sub-layer *IPsec, (de)fragmentation...* Multihoming sub-layer *Multihoming functions* Routing sub-layer *Forwarding...*	Network Layer Routing sub-layer *Forwarding...*	Network Layer *IPv6* Endpoint sub-layer *IPsec, (de)fragmentation...* Multihoming sub-layer *Multihoming functions* Routing sub-layer *Forwarding...*
Link Layer	Link Layer	Link Layer
Physical Layer	Physical Layer	Physical Layer

Identifiers

Locators

Multihoming Protocol

Host X
IP1, IP2

Host Y
IP1, IP2, IP3

Application Layer

| Transport Layer |
| *TCP/UDP* |

src ULID(X)=IP1(X)
dst ULID(Y)=IP1(Y)

Network Layer
IPv6

| Multihoming sub-layer |
| *Multihoming functions* |

src IP2(X)
dst IP3(Y)

Application Layer

| Transport Layer |
| *TCP/UDP* |

src ULID(X)=IP1(X)
dst ULID(Y)=IP1(Y)

Network Layer
IPv6

| Multihoming sub-layer |
| *Multihoming functions* |

src IP2(X)
dst IP3(Y)