Host Identity Specific Multicast

Zsolt Kovácsházi, Rolland Vida
{kovacshazi, vida}@tmit.bme.hu
Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics
H-1117, Budapest, Magyar tudósok krt. 2.

Abstract – Multicasting is a key technology for both users and service providers as it enables important bandwidth savings, and thus lower costs, for content distribution and group communication. The current network level multicast solutions have several weaknesses, such as the unsolved problem of access control and accounting, or the handling of mobility. These issues are partly related to a problem embedded in the use of the IP addresses themselves. Currently an IP address is both an identifier of who I am and where I am. The Host Identity concept tries to solve this problem by introducing a new, unique identifier, the Host Identity Tag, to answer the "Who?" question; the IP addresses will deal then only with the "Where?" question. This concept was used, up until now, mainly for unicast communications. However, it has many advantages that can make it an interesting solution for multicasting as well. In this paper we propose therefore the Host Identity Specific Multicast (HISM) model; we present all the architectural elements of the model, and show how it handles access control and accounting, multicast mobility, and how it provides native network layer multicast support in mixed, IPv4-IPv6 environments. We believe that by adopting the HISM concept, multicasting can finally reach the large scale deployment it deserves.

Index Terms – multicast, host identity, mobility, IPv4-IPv6 traversal, authentication

I. INTRODUCTION

The Internet has two important global namespaces: Internet Protocol (IP) addresses and Domain Name Service (DNS) names. These two namespaces have a set of features and abstractions that have powered the Internet to what it is today. They also have a number of weaknesses. One of the most important issues is the dual role of IP addresses, serving both as end-point identifiers and locators. There are many problems related to this duality, like the handling of mobility and multi-homing, or the transition from IPv4 to IPv6. In all these cases a node has several IP addresses (be it IPv4 or IPv6, home address or care-of address, from the address pool of one ISP or the other), and each time the address is changed all the correspondent nodes have to be alerted about it.

Changing the IP address of a node can be particularly sensible to handle if that node is a multicast source, and thus the root of a multicast tree (depending on the multicast routing algorithms that is used). This address change can occur if we have a multi-homed source or a mobile source, which can often be the case in a video conferencing session or in e-learning applications for example. Handling this source address change is a particularly costly operation in the current multicast models, as it requires either reconstructing the entire multicast tree or introducing some sort of bi-directional tunneling.

On the other hand, if a node, and in particular a multicast source or a multicast receiver, is identified by a permanent Host Identity Tag, instead of an ever-changing IP address, many of the above mentioned problems can be more easily handled.

In this paper we present a new multicast model based on the Host Identity concept. We describe all the elements of the architecture, and present the operation of the associated protocols. The goal of the paper is to show the viability of the concept. We do not present any simulation results, the efficiency of our solution is not compared in any way to other approaches, as simply there are no existing methods to build for example a native dual-stacked multicast tree. However, the concept could be later validated by implementing the model on the Planetlab [1] or GENI [2] test networks.

In the next section, we first present the problems of the current network layer multicast solutions that hindered their large scale deployment. In section III we then briefly describe the Host Identity namespace and the Host Identity Protocol (HIP) [3], the mechanisms our new multicast model will be based on. Section IV presents the details of the Host Identity Specific Multicast architecture we propose. Finally, section V concludes the paper.

II. MULTICASTING PROBLEMS

Multicasting is currently not yet deployed on an Internet scale, despite being continuously labeled as one of the most promising technologies for more than 15 years already. One of the reasons for this lack of deployment is the unsolved problem of access control and accounting. The traditional IP Multicast model (labeled today as ASM – Any-Source Multicast), as first defined in the PhD thesis of Steve Deering [4], was an open one, where anyone could join and leave the group without any control, anyone could become a source, even without being a member, and there was no global knowledge about the group membership, neither at the source nor at any other node.

Even though the later introduced Source-Specific Multicast (SSM) model, grown out from the Express [5] approach, modified some of these characteristics, a clean multicast access control mechanism, together with an associated accounting and charging scheme, are still missing. There have been some attempts to solve this problem, e.g., through the Multicast Control Protocol (MCOP) [6] which proposed an authentication scheme of the multicast receivers. However, MCOP has several drawbacks, the most important one being the fact that it is only capable of authenticating the subnet of the user and not the user itself.

Another interesting question regarding multicast data transmission is the handling of mobility, together with the corresponding IP address change. Or we can further enhance this problem by considering not only mobility but every kind of IP address changes. That can include multi-homed nodes changing from one access network to another, or dual-stack nodes changing from one IP version to another.

From the receiver's point of view mobility itself can be handled in several ways in a multicast environment. The most often cited approaches are Bidirectional Tunneling and Remote Subscription [7], but various hybrid methods, combining these two techniques also exist.

In Bidirectional Tunneling the mobile host connects to multicast groups via its home network, with the help of its home agent. It uses Mobile IP bidirectional tunneling for communications. A mobile station joining a foreign network first sends a binding update message to its home agent; then, a tunnel is created. From then on, the mobile station can join multicast groups just as if it were in its home network. It sends its multicast signaling message to its home agent making the home agent join the group. The home agent will terminate the multicast tree and forward data over the tunnel to the mobile host. As the mobile host joins a new sub network, it informs its home agent of its new location. The home agent then refreshes the tunnel end point to the new care-of-address of the mobile station.

In Remote Subscription, when the mobile host is in a foreign network, it uses the foreign network's local multicast router to join the multicast groups. It sends its group management reports to that router, and performs all multicast procedures the same way static nodes of that visited network do. When joining another network, the mobile station rejoins the multicast groups, this time through the local multicast router of the new network.

Building a tunnel or a new branch of the multicast tree, might be a time-consuming operation (depending on the tunnel's or the branch's length), period during which the client does not receive any data. An even harder problem to solve is multicast source mobility, an issue the majority of the solutions do not address. Source mobility, or simply the change in the source address due to any other possible reason, is an extremely sensible problem in the SSM model. As opposed to ASM, an SSM channel is identified by the source address as well; thus, only packets sent from that specific address are forwarded by the routers along the tree. All the packets coming from different addresses, including the source's possible new addresses, are dropped.

The solution is again to use a tunnel or reshape the tree, but as opposed to the case of receiver mobility reshaping here means not only adding a new branch, but rebuilding the entire SSM the source is the root of. This can take significant time in which the clients will not receive any data from the given multicast flow. An interesting hybrid solution to address both source and receiver mobility is M-HBH [8]; it is based on a recursive unicast delivery scheme to provide the multicast service, it eliminates encapsulation and tunneling, and reduces triangular routing. However, it provides significant performance enhancement only for small groups.

The third problem in multicast communication we want to deal with is the handling of receivers having different IP address versions. For unicast communication there are several solutions for IPv4/IPv6 traversal, e.g., using dual-stack routers, tunneling, 6to4 techniques, etc. While these methods can be applied to unicast communication, building native dual-stacked multicast trees is not yet solved. The problem is that the receiver node cannot create a valid join message if the source address is in a different IP version. The problem also exists if the source and the receiver have the same version of IP addresses, but there is a part of the network between them that only supports the other type of addresses. We can build a multicast tree that passes across that network segment through a tunnel, but in this case we loose the native multicast support.

As we have mentioned before, currently there are two different IP Multicast models in use, ASM and SSM. This might be also a problem for participants (either server or client side) not supporting both models, as they will be able to communicate only with those other participants that use the same model they are supporting. For example a client that does not implement IGMPv3 [9] or MLDv2 [10], the latest versions of group management protocols capable of source filtering in IPv4 and IPv6 respectively, will not be able to join an SSM session.

We believe that all the problems mentioned throughout this section can be efficiently solved by the introduction of a new network level multicast model based on the Host Identity concept. Using Host Identities provides a straightforward solution for authenticating not only the multicast receivers but the sources as well, mobility and multi-homing can be transparently handled, and native multicast support can be extended to mixed IPv4-IPv6 environments as well. Also, using the new model as an overlay on top of the existing ASM and SSM models can enable users of the two different IP multicast versions to communicate with each other. But before describing the details of the model we propose, let us first briefly present the main concepts of the Host Identity namespace and the corresponding Host Identity Protocol (HIP).

III. The Host Identity namespace and Protocol

IP addresses were created to serve two roles. One of the roles is identifying the host, the other one is giving the necessary information for routing purposes. These roles were bound together for many years, because of the lack of mobility. However, today mobility handling is a must for both services, applications or communication devices. We are living in the world of mobile workstations, intelligent mobile phones, etc.

A new namespace, based on the Host Identity concept, tries to solve the aforementioned problems, by separating the two roles of IP addresses. The Host Identity namespace consists of Host Identifiers (His), which are the public key of an asymmetric key-pair. Each host will have at least one, but typically several Host Identities. Each Host Identity uniquely identifies a single host, i.e., no two hosts have the same Host Identity. The Host Identity, and the corresponding Host Identifier, can be either public (e.g., published in the DNS) or unpublished. Client systems will tend to have both public and unpublished Identities [11] [12]. The Host Identifiers take on the role of end-point identifiers, while the IP addresses will only be used as locators, for routing to the host [3] [13].

There is a subtle but important difference between Host Identities and Host Identifiers. An Identity refers to the abstract entity that is identified. An Identifier, on the other hand, refers to the concrete bit pattern that is used in the identification process.

For the Host Identity namespace a new protocol, called the Host Identity Protocol (HIP), and a cryptographic exchange, called the HIP Base Exchange, are introduced.

The HIP Base Exchange is a two-party cryptographic protocol used to establish the communication context between hosts [14]. The Base Exchange is a Sigma-compliant [15] four packet exchange. The first party is called the Initiator and the second party the Responder. The four-packet design helps to make HIP DoS (Denial of Service) resilient. The protocol exchanges Diffie-Hellman keys in the second and third packets, and authenticates the parties in the third and fourth packets. Additionally, the Responder starts a puzzle exchange in the second packet, with the Initiator completing it in the third packet before the Responder stores any state from the exchange.

The HIP protocol provides for limited forms of trust between systems, enhances mobility, multi-homing, and dynamic IP renumbering, helps in protocol translation/transition, and reduces the vulnerability to certain types of denial-of-service (DoS) attacks. When HIP is used, the actual payload traffic between two HIP hosts is typically, but not necessarily, protected with IPsec. The Host Identities are used to create the needed IPsec Security Associations (SAs) and to authenticate the hosts. When IPsec is used, the actual payload IP packets do not differ in any way from standard IPsec-protected IP packets.

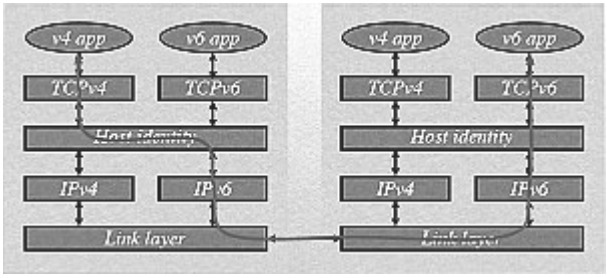

Figure 1. Host Identity – IP address resolution

There are two main representations of the Host Identity, the full Host Identifier (HI) and the Host Identity Tag (HIT). The HI is a public key and directly represents the Identity. Since there are different public key algorithms that can be used with different key lengths, the HI is not good for use as a packet identifier, or as an index into the various operational tables needed to support HIP. Consequently, a hash of the HI, the Host Identity Tag (HIT), becomes the operational representation. It is 128 bits long and is used in the HIP payloads and to index the corresponding state in the end hosts.

The host operation systems are extended by a new host identity layer. This layer is placed between the network and transport levels. Transport level ports are bound to the host identities and not to the IP addresses. This can be seen on Figure 1.

IV. Host Identity Specific Multicast

After this short introduction on the Host Identity namespace and protocols, let us now see how these concepts could be used to provide a multicast service. In the following we describe thus the elements of a new network layer multicast model we propose, called Host Identity Specific Multicast (HISM).

A. Identifiers

First we introduce the different identifiers used in our model. The concept is mainly based on the Host Identity namespace identifiers. In the currently used ASM or SSM models we have two identifiers:

- Source IP address – S
- Multicast group address – G

Actually in ASM we only have the multicast group address, as we use the (*, G) information to build shared trees, the * denoting the fact that any source can send data on that shared tree. As opposed to this, in the SSM model we use the (S, G) information to build source specific trees, the source S being the root of such a tree.

As we have mentioned before, the main problem with the source or receiver IP address change is that we have to either partially or totally rebuild the multicast tree. That is why we introduce the Host Identity Tag (HIT) to identify the multicast source and receivers, instead of their IP

address. The HIT is a 128 bit long identifier, just like an IPv6 address. However, this identifier remains unchanged even if the IP address of the host (either the source or the receivers) changes. This means that we, if the multicast tree is built based on the HIT information, we do not have to rebuild each time some host changes its address. We use the following notations:

- HIT-R: Host Identity Tag of the Receiver
- HIT-S: Host Identity Tag of the Source

This is a good solution if we want to solve the problem of mobile or multi-homed multicast sources and receivers. But our goal is to create such a multicast model in which any kind of address change, including the change from one IP version to another, can be handled. That is why we introduce another identifier, called the Session ID – SID. The SID should replace the multicast group address and it is supposed to be IP version independent. The SID is a 26 bit long identifier, from which any application or architectural element can easily create either an IPv4 or an IPv6 multicast address, if needed. The mapping should be done in the following way:

- IPv4: 1110 | 11 | 26 bit SID
- IPv6: FF | FF | fill pattern or identifier pattern | 26 bit SID

In these mappings we took into account the IP address ranges reserved for multicast addresses in IPv4 and IPv6 respectively; inside these ranges we further reserved a sub-range dedicated to HISM SIDs. In the IPv6 mapping for example the identifier pattern should represent a unique multicast address range so that an application could easily identify that it is a Host Identity Specific Multicast address it is working with. In the IPv4 mapping the fifth and the sixth bit serve this purpose.

Taking all these into account, we can now define a HISM channel as being identified by the (HIT-S, SID) identifier pair, following the model of SSM identifiers. Such a channel identifier preserves all the advantages of the SSM-like identification, while getting rid of the variable source IP address.

In Table 1 we summarize the identifiers used in the different multicast models. Please note that although SSM could be seen as a subset of ASM, SSM and ASM work on distinct address ranges and are not compatible with each other; e.g., a (*, G) ASM join message cannot be sent for a G address reserved for SSM channels [16]. Our HISM model will act as an overlay on top of the existing ASM and SSM models, being compatible with both of them. The details of how this compatibility is achieved will be presented in a later section.

B. The Model

We now introduce the main functions embedded into the Host Identity Specific Multicast model. Then, we present the different architectural elements in more detail.

Table 1. Identifiers used in the different multicast models

	Multicast group/channel identifier	Compatibility
ASM	(*, G)	ASM only
SSM	(S, G)	SSM only
HISM	(HIT-S, SID)	ASM and SSM

The traditional network level multicast communication model involves the following steps:

1. The application gives the source address – multicast address (S, G) couple (according to the SSM model), or just the multicast address G (in the ASM model) it wants to listen to.

2. The system creates the necessary entries of the multicast addresses in the operation system's registries.

3. The multicast receiver node sends out group management join messages (actually IGMP or MLD Reports), depending the IP version of the multicast stream.

4. This message arrives to the first multicast router (Designated Router – DR).

5. The DR starts the tree building process, by sending a PIM Join message (or another signaling message, depending on the multicast routing protocol that is used).

6. The PIM Join message reaches the DR of the source, or an intermediate router that is already part of the multicast tree. At this point the join process is terminated, and multicast data starts flowing along the newly constructed tree branch.

As opposed to this, in the Host Identity Specific Multicast model the communication is established through the following steps:

1. The application gives the (HIT-S, SID) couple (according to the SSM model), or the SID (in the ASM case) it wants to listen to.

2. The Host Identity layer knows what kind of addresses the client has (IPv4 and/or IPv6); based on this it creates the necessary entries of the multicast addresses in the operation system's registries (mapped from the SID).

3. Group management joining based on the (HIT-S, SID) couple. The join message should contain the HIT-R identifier of the receiver as well; it will be used for the authentication that is described later. To enable this, we propose a new, unified group management protocol that will also be described in a later subsection.

4. This join message arrives to the DR, the first multicast router. The DR then initiates the authentication process. The authentication can be handled on the router itself or on another dedicated server. For the authentication we use the following information: HIT-S, SID, HIT-R.

5. The authentication server discovers the current IP address of the source from the HIT-S tag it received. It also checks, based on the HIT-R tag, whether the client is authorized to receive the stream or not. The authentication server also has a register about the dual-stack edge routers for IP version traversal purposes. Thus, the server gives back the following information to the DR:

- Client is authorized or not
- Source IP address(es)
- IP addresses of (optimal) dual-stack edge routers to the source

6. Based on these information, the DR starts the tree building process:

- Normal PIM Join is performed if the IP versions of the receiver and the source are the same
- If not, then tree building starts as if the dual-stack edge router would be the multicast source; PIM Join messages are thus sent towards this edge router. However, we have to signal that this router is eventually not the real source of the tree and we need address conversion. We also have to put in the message the real IP address of the source we want to listen to. The details of how these modified signaling messages are built and handled will also be given later.

7. The dual-stack edge router handles the IP version conversion and it starts building the other part of the multicast tree

8. The Join message finally reaches the DR of the source DR, or an intermediate router that is already part of the multicast tree. At this point the join process is terminated, and multicast data starts flowing along the newly constructed tree branch.

Figure 2 shows the above described process. The numbers on the arrows indicate the steps of the joining process.

The Host Identity Specific Multicast (HISM) model tries to take advantage of the Host Identity namespace, by using host identities instead of multicast addresses or channels. With this model, we can create a unified multicasting architecture on top of the currently existing ASM and SSM schemes. This model supports the authentication of both the source and the receiver parties, enabling the deployment of accounting and charging schemes.

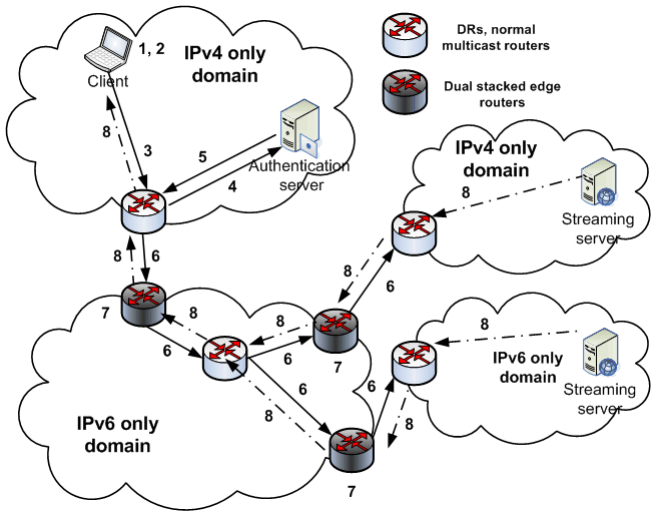

Figure 2. Communication in the Host Identity Specific Multicast model

As mentioned in chapter II, in the SSM model when the multicast source changes its IP address we have to rebuild the entire multicast tree. With the introduction of the Host Identity Tags as source identifiers, the trees do not have to be rebuilt but only properly maintained. This can solve the problem of source mobility or that of multi-homed sources.

All these important features have also a cost, namely the fact that the Host Identity concept requires the upgrade of all host stacks with the Host Identity layer being introduced between the TCP and the IP layers. However, Host Identity layers are not necessary in the routers. For the operators this is a very important feature of the HISM concept: it gives them an enhanced multicasting solution without requiring any new hardware to be deployed in the core. The new elements can be implemented by software upgrades, which is cheaper and easier to solve. In the following we present in more details the different architectural elements of the HISM model.

C. Changes in the Applications

At the application level we have to be able to give the (HIT-S, SID) information of the multicast channel we want to receive. If an application supports IPv6 communication than the HIT-S can be easily given; it is a 128-bit identifier, just as an IPv6 address, so the application does not have to be modified. Providing the 26-bit long Session ID is a bit more complicated. Basically there are two possibilities to solve this problem:

- either the application itself should be made capable of handling Session IDs;
- or the Host Identity layer should recognize that a multicast address has been given in a multicast application and convert it to a Session ID. This approach should be used in case of older applications that are not capable of handling the identifiers introduced by the HISM model.

D. Group Management

Group management protocols are used by the receivers to signal their wish of joining or leaving a multicast group. A multicast capable host has to support some kind of group management protocol. The most used such protocols are the Internet Group Management Protocol (IGMP) for IPv4 hosts, and the Multicast Listener Discovery (MLD) protocol for IPv6 hosts. The latest versions of these protocols, IGMPv3 [9] and MLDv2 [10] respectively, support source filtering, i.e., receiver do not only specify the multicast group they want to join or leave, but also the sources they want to listen to. This capability is needed for the Source Specific Multicast model.

Both IGMP and MLD are soft-state protocols, i.e., the local multicast router periodically checks whether or not there are any clients interested in the multicast streams. Thus, the router sends out Query messages, while the receivers answer with Reports.

As there are different identifiers and extra information the HISM capable client has to send to the multicast routers, we propose a new group management protocol to support HISM. This group management protocol will be a unified protocol, i.e., it would both support HISM based multicasting (independently from the IP version) and if necessary older IPv4 and IPv6 ASM or SSM multicasting. The main advantage of a unified group management protocol is that all stack implementations should need to run one protocol and would be able to communicate in both IP versions. In this new unified group management protocol we introduce the support of the Host Identity Tags and Session IDs.

An important feature of the new protocol is that it also supplies the HIT-R tag of the receiver. This is needed later for the authentication process. When in compatibility mode with the older models, supplying the HIT-R tag is only optional, but it still can be used for the same reason.

Let us now see how these functionalities can be included in the protocol. Our departure point is the MLDv2 protocol. The router periodically send *query* messages, to discover the multicast listeners on its interfaces. These are called General Query messages. The hosts interested in multicast flows have to send back *report* messages. In these reports the clients should send now, instead of the usual (S,G) pairs, the necessary information for joining a HISM session, namely the (HIT-S, SID, HIT-R) identifier triplet.

The modified message formats for group management queries and reports are shown in Figure 3 and Figure 4. In the figures we highlighted the most important changes when compared to the format used by MLDv2.

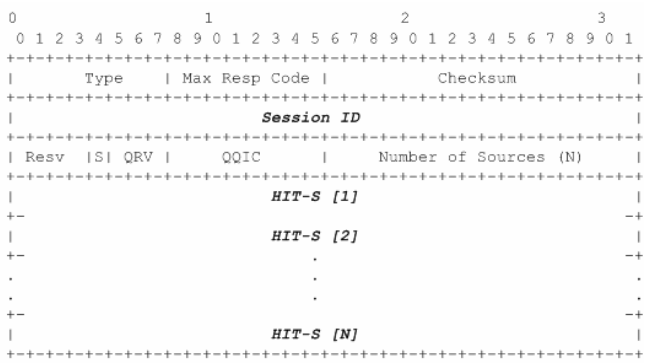

Figure 3. New group management query

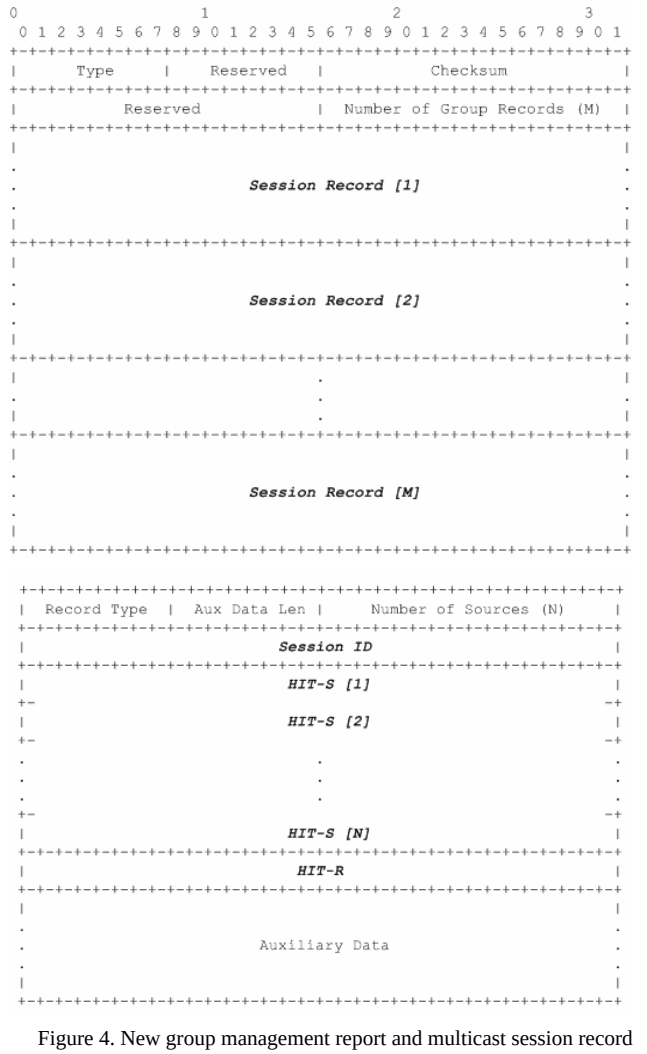

Figure 4. New group management report and multicast session record

E. Authentication

The next element of the HISM model is the authentication of the user. We propose an authentication functionality that is based upon the MCOP (Multicast Control Protocol) solution [6].

Thus, the hosts try to join a multicast session identified by a (HIT-S, SID) couple instead of the traditional (S, G) pair. As we previously mentioned, the Report message used for

this purpose will contain the HIT-R tag as well. This can be very effective for authenticating a single host and not just the subnet it is currently located on, as it was done in MCOP. A HIP base exchange can be introduced if both sides, source and receiver, need authentication from the other. Always using the streaming server itself in this process can of course affect its performance; thus, some kind of a proxy might be needed at the source side.

After the first multicast router (DR) receives a new group management join message concerning the (HIT-S, SID) identifier pair, it first sends the (HIT-S, SID, HIT-R) information to the authentication server. The server has to validate the client first. After the validation is successful, it has to retrieve the actual source address of the server based on the HIT-S tag. How this retrieval is done depends on how the HIT – IP address pairs are stored and transmitted. The most common way should be to get this information from the DNS, where a new Resource Record (RR) should be created. There is currently a proposal for storing Host Identity information in the DNS [12]. The record suggested by this proposal can store the Host Identity, the HIT and the domain name of the Rendezvous Servers. It can be further enhanced if we use another RR for the HISM multicast information. Another important information the authentication must keep track is the address of the dual-stack edge routers.

The authentication server has to contain the following information:

- SID
- HIT-S
- Current multicast source IP address
- Dual-stack edge router IP address
- The list of connected DRs
- The list of already checked HITs

After retrieving all the necessary data, the server sends back the following information:

- If IP versions are the same for the client and the server: *(Client authentication successful or not, Source IP address);*
- If IP versions differ: *(Client authentication successful or not, Source IP address, Dual-stack edge router IP address)*

After a session is started it will be the task of the authentication server to check periodically, or on demand, the current IP address of the source. The source itself can signal an address change alerting directly all the concerned parties, or it can be the authentication server detects and alerts the DR about the change in the address of the source. The authentication is handled in a soft-state manner. This means that the clients must periodically re-authenticate themselves. Different timings can be used with fixed and mobile clients for optimizing network performance.

F. Multicast Routing

The most important element we must deal with is multicast routing. Our work is based on the PIM-SSM protocol [17]. The following areas of the protocol's core functionality must be enhanced in order to support the HISM model:

- First hop router (DR) functions for enabling authentication and authorization
- Core PIM routing functions for building new (HIT-S, SID) based trees, abd thus enabling dual-stack trees. This is needed to support mobile or multi-homed sources and clients.
- Dual-stack edge router functionalities

1. DR Functions

First hop routers or Designated Routers (DR) play a significant role in a subnet's multicast communication. They are responsible for registering the user, starting the authentication process, and finally starting the building of the multicast tree. If the domains have several multicast capable routers than one of them will be chosen as a DR. This is useful because without this mechanism more than one multicast router could start the tree building, which could lead to multiple copies of the stream arriving to the domain.

Thus, at the DR side we must first handle the different group management messages arriving from the clients. Now instead of the normal (Source Address, Group Address) pair the DR receives (HIT-S, SID) information. Based on these it should automatically start the authentication process by sending the necessary information to a designated server or do the authentication itself. After the authentication is done, the DR receives the following data:

- Client is authorized or not
- Source current IP address
- IP addresses of edge routers to the source (the nearest to the client, most likely at the edge of it's domain)

If the client is authorized then the DR starts the tree building process by sending a Join messages towards the source. The message format can be seen on Figure 5. There is only one additional flag used, compared to the conventional PIM Join format. This is the Conversion bit, which is used to signal whether or not an address conversion will be concerning at least one of the source addresses.

2. Core PIM routing

We want to build multicast trees that are both independent from the IP versions used in the communication and can significantly enhance the routing efficiency even if an IP address change occurs at the source. This is the reason why we now use the HIT-S and the SID information as tree states in the multicast routers. The core routers do not have a Host Identity layer that could resolve the HITs into IP addresses.

This means that we must explicitly tell the multicast routers the needed IP addresses, as they can route the packets only based on them. The encoded group and source formats are going to carry all additional information.

On Figure 6, the new HISM compatible encoded group format can be seen. Instead of the multicast address it now carries the Session ID. The HISM compatible encoded source format, which can be seen on Figure 7, has several additions. In this we transmit all the necessary information for the tree building (HIT-S) and routing (the source IP address). We also forward here the dual-stack edge router information for the next multicast routers. The Conversion bit signals that this source address will need an IP version conversion and the edge router address should be used for routing (it is a temporary destination address). On the following figures all changes or extensions are written in bold and italic. All flags are stored in fields that are originally reserved in the PIM-SSM specification.

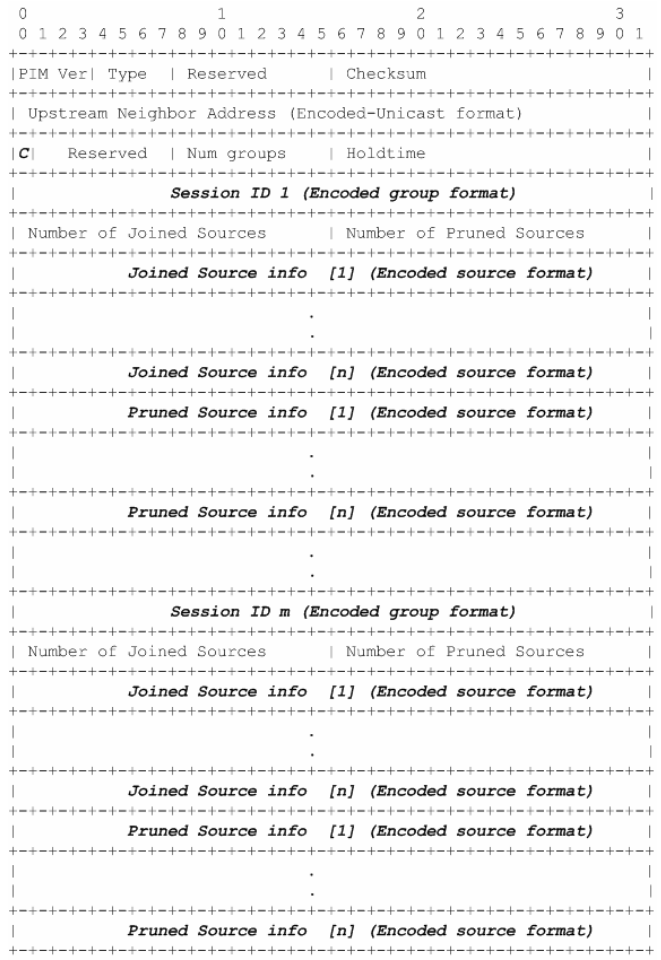

Figure 5. HISM compatible PIM-join/prune message

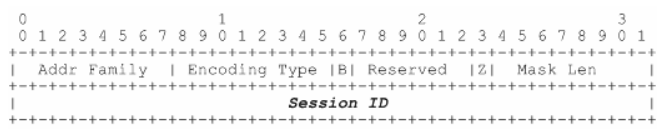

Figure 6. HISM compatible encoded group format

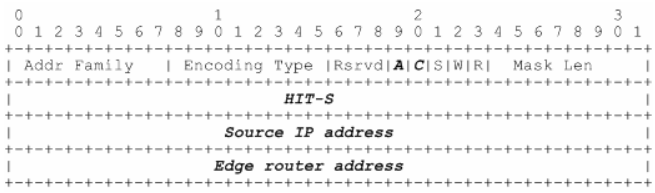

Figure 7. HISM compatible encoded source format

After we have built the multicast tree, we have to continuously check if the source IP address has changed or not. Now that we build the trees based on the HIT-S, we do not have to rebuild the tree, but the destination address of the join/prune messages will be updated. We use the Address change bit to signal that there has been a change in the source's IP address. If the IP address change resulted in IP version change as well, then we use both flags.

Thus, if the source changes its address, the DRs on the receiver subnets will map the HIT-S to the new IP address. The PIM Join messages will reflect then this change. However, the intermediate PIM routers will not be aware of anything, they will handle the same (HIT-S, SID) multicast tree they dealt with before. The changed source IP address will of course result, if necessary, in a reshaping of the last portion of the tree that leads to the new location of the source. However, this is done in a transparent manner. If for example the source receives from the DHCP server a new IP address on the same subnet, the tree will basically not change at all; in the SSM model this address change would trigger the reconstruction of the entire tree.

3. Dual-stack edge routers

In the HISM model we are capable of creating native dual-stack multicast trees. This means that both IPv4 and IPv6 capable hosts are able to receive an IPv4 or IPv6 multimedia stream at the same time without having to use unicast methods at all. Without the HISM support we can not even address the source if it's using another type of IP address. But even if the receiver and the source address types match, there could be intermediate network portion that use a different IP version, hindering thus the communication.

The problem could be solved through the use of some kind of tunneling methods. By doing so, we can build multicast trees across domains having different IP versions. However, this would mean that in some domains we will use unicast for the data forwarding, and if there were other users in those domains hooked on the same multicast tree, they would have to build another multicast tree for themselves. Also, the tree might comprise several tunnels established in the same domain, loosing the efficiency of multicasting. An example for such a multicast tree can be seen on Figure 8.

The dual-stack edge routers always know to which other edge router they must route the packets when the destination is not in their domain. We use this functionality to always change to the native version of the current domain and signal in our join/prune messages the necessary changes.

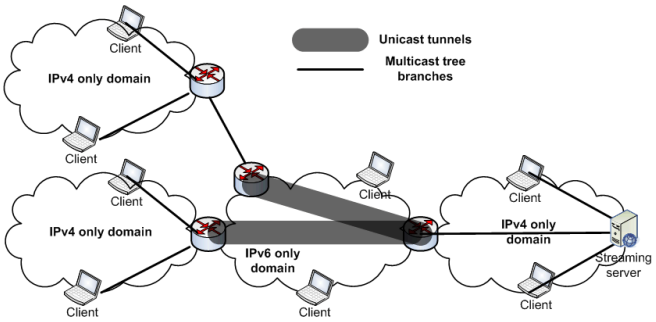

Figure 8. Non-native dual IP multicast trees

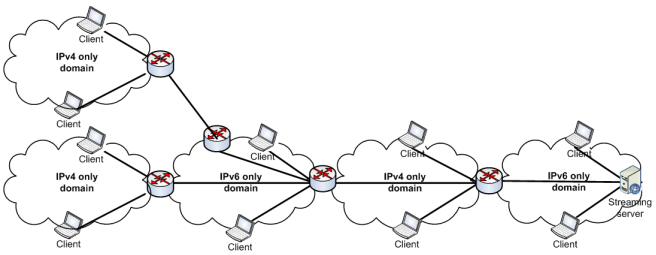

Figure 9. Native HISM based dual IP multicast trees

Figure 9 shows a native dual stacked multicast tree. Every domain can have their own clients without the need of an own multicast tree. Every time a domain uses a different IP version we will use the edge router address instead of the real source address for routing. At the edge router we are able to route the packets natively towards the next domain or to the source itself. It is always the edge router's responsibility to check if conversion is needed, and to change the data packets from one IP version to another on their way from the source to the receivers.

V. Conclusion

In this paper we presented a new multicast model that provides support for building native dual-stacked multicast trees and makes tree rebuilding unnecessary in case the IP address of the source changes (due to mobility, multi-homing, dynamic address allocation, etc.). The model also provides authentication functionalities that can be crucial for both service and content providers. We also presented a brief overview of a new unified group management protocol. We believe that by adopting the HISM concept, multicasting can finally reach the large scale deployment it deserves.

References

[1] Planetlab – An open platform for developing, deploying and accessing planetary scale services, http://www.planet-lab.org/

[2] GENI – Global Environment for Network Innovations, http://www.geni.net/

[3] R. Moskowitz, P. Nikander, P. Jokela, T. Henderson, "Host Identity Protocol", Internet Draft, draft-ietf-hip-base-06.txt, June 2006.

[4] S. Deering, "Multicast Routing in a Datagram Internetwork", *Ph.D. thesis*, Stanford University, Palo Alto, CA, USA, December 1991.

[5] H. Holbrook and D. Cheriton, "IP Multicast Channels: Express Support for Large-Scale Single-Source Application", in *Proceedings of ACM SIGCOMM'99*, Cambridge, MA, USA, September 1999, pp. 65-78

[6] R. Lehtonen, J. Soini, J. Majalainen, H. Vatiainen, M. Tammi, "MCOP operation for first hop routers", Internet draft, draft-lehtonen-mboned-mcop-operation-02.txt, December 2004.

[7] D. Johnson, C. Perkins, J. Arkko, "Mobility support in IPv6", RFC 3775, June 2004.

[8] R. Vida, L. Costa, S. Fdida, "Mobile Hop-by-Hop Multicast Routing", Computer Networks, Elsevier, Volume 44, Issue 6, April 2004, pp. 789-812.

[9] S. Deering, B. Cain, I. Kouvelas, B. Fenner, A. Thyagarajan, "Internet Group Management Protocol version 3", RFC 3376, October 2002.

[10] R. Vida, L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

[11] R. Moskowitz, P. Nikander, "Host Identity Protocol (HIP) Architecture", RFC 4423, May 2006.

[12] R. Moskowitz, J. Laganier, "Host Identity Protocol (HIP) Domain Name System (DNS) Extensions", Internet Draft, draft-ietf-hip-dns-06.txt, February 2006.

[13] T. Henderson (editor), "End-Host Mobility and Multihoming with the Host Identity Protocol", Internet Draft, draft-ietf-hip-mm-04.txt, June 2006

[14] P. Jokela, R. Moskowitz, P. Nikander, "Using ESP transport format with HIP", Internet Draft, draft-ietf-hip-esp-03.txt, June 2006.

[15] Krawczyk, H., "SIGMA: The 'SIGn-and-MAc' Approach to Authenticated Diffie-Hellman and Its Use in the IKE-Protocols", in Proceedings of CRYPTO 2003, pages 400-425, August 2003.

[16] Z. Albanna, K. Almeroth, D. Meyer, M. Schipper, "IANA Guidelines for IPv4 Multicast Address Assignments", RFC 3171, August 2001

[17] S. Deering, "Protocol Independent Multicast – Sparse Mode (PIM-SM)", RFC 2363, June 1998.