Network Working Group
INTERNET DRAFT

Expires: 1,2000

Dina Katabi
John Wroclawski
MIT LCS
June, 1999

A Framework for Global IP-Anycast (GIA)
<draft-katabi-global-anycast-00.txt>

Status of this Memo

This document is an Internet-Draft and is in full conformance with allprovisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed athttp://www.ietf.org/shadow.html.

Abstract

This document describes GIA, an architecture for a scalable Global IP-Anycast service. In contrast to previous approaches, which route IP-anycast through the unicast routing system, GIA provides IP-anycast with its own routing protocol. To scale, GIA pushes the overhead of anycast routing to the edge of the network and off-load the middle routers of the burden of storing anycast routes.

GIA's main contribution is its interdomain routing protocol, which is based on promoting the quality of an anycast route according to its level of usage. The protocol generates two types of routes; default low-cost anycast routes, which consume no bandwidth or storage space; and high-quality shortest-path anycast routes that are customized according to the beneficiary domain's interests.

1.0 Introduction

IP-anycast is a network service whereby receivers that share the same characteristics are assigned the same

anycast address. A sender interested in contacting a receiver with those characteristics sends its packet to the anycast address and the routers conspire to deliver the packet to the nearest receiver to the sender, where nearest is defined according to the routing system measure of distance.

Anycast' s early applications are service oriented. In particular, RFC 1546 [19] proposes anycast as a means for service location and host auto-configuration. As an example of using anycast for service location, a set of replicated ftp servers is assigned the same anycast address. By contacting the anycast address a user in France who wants to retrieve a file is directed to the European server while a user in the States is directed to the American server. As an example for using anycast for host auto-configuration, one can imagine assigning the same anycast address to all Domain Name Servers. In that case, a host that is moved to a new network needs not be configured with a new DNS address since it can use the global anycast address to access the local DNS server anywhere.

Recently, new anycast applications that do not directly provide a service to an end user have been developed. For example, [5] uses a global anycast service as an infrastructure to develop an inter-domain multicast routing scheme, while [16] uses anycast as a method to develop more efficient intradomain multicast trees.

Currently, there is no scalable design for providing a global ip anycast service. The major obstacle that faces research in this area is anycast's defiance of hierarchical aggregation. Actually, an anycast address, similarly to a multicast address, represents a group of nodes that share a particular characteristic. As such, there is no reason to expect anycast group-topology to be hierarchical or to comply with the underlying unicast topology. Unfortunately, this means that to provide a global anycast service we need to advertise each anycast group to the entire Internet. Thus, routing tables will have a new component - anycast host routes - which grows proportionally to the total number of anycast groups in the Internet. Given that the routers in the backbones already suffer from the large size of their routing tables, this approach does not scale.

The problem can be partially alleviated by use of scoped anycast addresses for groups whose members are confined to a particular network region. However, many anycast applications such as the one in [5] require a global anycast service for which no scoped anycast address works. Moreover, sometimes even when the anycast group is

currently confined to a particular region there is an incentive to use a global anycast address. For example, a company that provides an online service might use an anycast address to have its customers access the nearest online office. Although the company online offices might currently cover only the east cost, the company would like to use a global address, since a scoped anycast address prevents future expansion to the west coast or to Europe.

2.0 Background and Related Work

Anycast was introduced to the Internet literature by RFC 1546 **[19]. This document motivated the need for an anycast** service as a means for service discovery and host auto-configuration. The document also pointed out the major difficulties associated with IP-anycast, which are the stateless nature of anycast and its defiance of hierarchical aggregation. The authors suggested approaches for building TCP connections on top of anycast addresses, and discussed possible addressing schemes. They considered it "wiser to use a separate class" to assign anycast addresses than to carve them from the existing unicast address space.

When the Internet Engineering Task Force considered finding a replacement for IPv4, all major candidates adopted anycast as an addressing mode (Pip [8], SIPP [14], IPv6 [13]). In particular, IPv6 allocates anycast addresses from the unicast address space making them indistinguishable from their unicast counterparts. Each anycast group is confined to a particular topological region with which it shares the prefix. Within the region identified by the shared prefix, each member of the anycast group must be advertised as a separate entry in the routing system (commonly referred to as a "host route"); outside the region, the anycast address may be aggregated into the routing advertisement for the shared prefix. Note that global anycast groups have a prefix of null and must be advertised as separate routing entries throughout the entire Internet.

3.0 Requirements

Our objectives are to implement a scalable IP-anycast service that complies with the semantics defined in RFC 1546**. Thus, we want a service that satisfies the following** characteristics:

- Global: In a global anycast service, group members can potentially exist anywhere in the Internet and be accessible to senders in their neighborhood. Although

confining each anycast group to a configured region satisfies some range of applications it leaves a space that can't be filled unless there is a global anycast service. For example, providing a host with the ability to configure itself wherever it is plugged into the Internet is infeasible unless we assign the same anycast address to all of the dynamic-host-configuration servers.

- Scalable: A service is scalable when the overhead in terms of storage, traffic and processing is manageable in a large heterogeneous inter-network. Thus, scalability is not achieved by readily using fewer resources, it also depends on having a good topological alignment between resource consumption and resource availability.

- Efficient: The anycast service is efficient when there is a high probability that an anycast packet is received by the nearest group member to the sender.

4.0 Design Rationale

The traditional belief that IP-Anycast should be routed similarly to its unicast counterpart has hampered the service, and a routing scheme that recognizes the characteristics of anycast and benefits from them to scale the service is needed.

Actually, given that most anycast groups represent services, it seems inefficient that a domain has to spend equal amount of network resources on learning storing and maintaining different anycast routes. For example, why does an MIT router spend equal resources on the route to the replicated CNN web server and the route to the replicated Algerian news agency? The first route is used every day by some MIT user, while the second one is rarely used if ever. In fact, anycast's primary usage (to date) is service discovery (host auto-configuration is a form of service discovery). Thus, one can extrapolate information about other networked services' access pattern to anycast. One well-studied example is the web's access pattern, which reveals a reasonable amount of locality of interests that justifies the use of proxies. Therefore, it seems that although anycast is not amenable to hierarchical aggregation, the service's potential applications make it amenable to caching. In particular, the authors think that at any given time there is a predictable set of anycast groups that hosts in a domain access with high probability and that this set is much smaller than the total number of anycast groups in the Internet. This means that at a particular domain anycast routes are not equally valuable, and that a good anycast routing protocol devotes more

resources to build good routes to repeatedly accessed anycast groups.

GIA's design allows a domain to discover, store and maintain efficient routes to anycast groups repeatedly accessed by users in this domain, while supporting a cheap fallback mechanism to send packets to unpopular groups. In fact, the fallback mechanism does not consume any additional network resources because it is based on mapping the anycast topology to the underlying unicast topology by which all routers in the Internet know how to forward packets.

5.0 Design Details

This section describes the details of the architecture. Note that GIA is concerned only with routing and addressing issues and does not address the data-link layer or the transport layer issues.

5. 1 Address Architecture

Our architecture assigns anycast its own address space but it allocates anycast addresses to domains according to the unicast hierarchy. More precisely, an anycast address is a concatenation of an anycast indicator, the unicast prefix of the home domain, and the group ID (see Figure 1). The anycast indicator is a fixed length prefix that differentiates anycast addresses from their unicast and multicast counterparts, such that anycast packets can be recognized andforwarded by the anycast forwarding protocol. One possible anycast indicator is the bit-pattern '11110'. On the other hand, the home domain's prefix identifies the Internet domain that owns the unicast space from which this anycast address is derived. It is the same as the unicast prefix carried by the routers inside the home domain. In contrast to the anycast indicator, the home domain field has a variable length that depends on the size of the domain's unicast address space. GIA requires each anycast address to be associated with an Internet home domain, and it requires that the home domain contain at least one machine configured with the anycast address. Note that the anycast address is still global and can be assigned to machines anywhere in the Internet (as opposed to IPv6 architecture where all group members reside inside the region or the domain with which they share the prefix). Finally, the group ID identifies a particular anycast address among the anycast addresses associated with the same home domain. Note that when shifting off the anycast indicator the anycast address becomes similar to a unicast address from the home domain (see Figure 2).

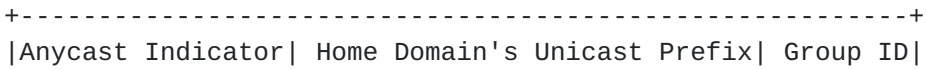


Figure 1: The syntax of an IP-anycast address

Figure 2: Shifting the anycast indicator off, an anycast address becomes a unicast address. Shifting is not performed on the packet but on the lookup variable

The above architecture means that every Internet domain is allocated an anycast address space that is proportional to its unicast address space (In fact, for the case of IPv4, if the anycast indicator is 11110 then domains whose unicast prefix is smaller than x.x.x.x/27 are not allocated any anycast address space. However, we don't know of any Internet domain whose prefix is smaller than x.x.x.x/24.)

The domain might use the anycast addresses to provide global services or it might lease them to end users providing an on-line service. For example, assume X is a company that provides an online service and that wants its customers to use an anycast address to locate the nearest on-line office to them. It is likely that X has a main office connected to the Internet somewhere and consequently can use one of the anycast addresses associated with its network for its on-line service, in which case the home domain for the anycast group is X's domain. If X does not have its own domain, X can lease an anycast address from its service provider. In this case the group's home domain is the provider domain. Anyway, if company X grows in the future and opens a new online office, then the new office can use the same anycast address and be accessible to customers in its neighborhood.

On the other hand, for well-known anycast addresses used for host auto-configuration (such as the group of all DNS) the home domain should be either in one of the backbones or a virtual domain advertised by the backbones.

5.2 Anycast Address Assignment

The process according to which a domain D assigns an anycast address out of its allocated address space to an end user is domain dependent and can be the same as the one used for assigning unicast addresses. For example, the

anycast address might be assigned manually by the administrator, or the domain might have special address assignment servers that lease anycast and unicast addresses with certain lifetime to end-users. The design requires the user to setup at least one machine with the anycast address in the home domain. However, the user can assign the address to other machines anywhere in the Internet.

In addition to the aforementioned address assignment procedure, some anycast addresses will be 'well-known addresses', assigned by the Internet Assigned Numbers Authority (IANA) to particular auto-configuration groups such as all Dynamic Host Configuration Servers or all DNS.

5.3 Anycast Address Advertisement

Anycast reachability information is generated and propagated only by routers. A host that wants to join an anycast group has to ask its next hop router to advertise the address on its behalf, which can be achieved by adding a new message type to either IGMP [26] or the Neighbor Discovery protocol [25]. A router that receives such a request processes it through some security checking procedure (to be designed as a future work), and if compliant it marks the address to be advertised according to the anycast routing protocol adopted by the domain. The router uses a keepalive mechanism to ascertain the availability of the anycast member. It never advertises an address after the member becomes inaccessible.

5.4 Anycast Routing

>From an edge domain perspective, an anycast group is classified according to the following classifications:

- Internal anycast group: An internal anycast group to domain D is a group for which D has internally at least one member. Note that, all groups are internal to their home domain. However, groups might be internal to domains other than their home domains.
- External Anycast group: An external anycast group to domain D is a group for which D has no members.
- Popular anycast group: A popular anycast group in Domain D is an external group that clients in domain D repeatedly access.

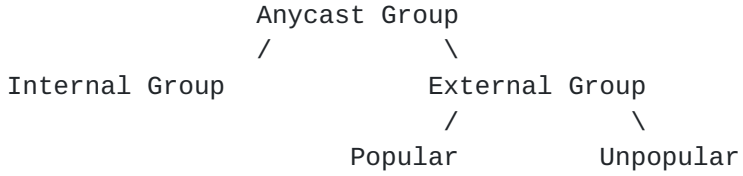


Figure 3: Anycast group classification from an edge domain perspective

Since GIA generates routes whose quality differs according to their anticipated usage, it distinguishes between routing internal anycast groups, routing external-popular anycast groups, and routing external-unpopular anycast groups.

5.4.1 Routing Internal Anycast Groups

Internally, each domain routes its internal groups using its own unicast routing protocol, which means that each member is advertised as a separate entry in the routing system, and each router knows the nearest anycast member. This holds the implied assumption that the number of internal anycast groups in a domain stays manageable. Since this number can be administratively controlled, we expect each domain to control this number to stay within the limits of locally available bandwidth and storage space.

Intradomain routing protocols based on the Distance-Vector algorithm such as RIP work without any modification [19]. For protocols based on the Link-State algorithm to work correctly, routers should abstain from routing through an anycast address. For example, assuming A is an anycast group, router R1 in Figure 4-a should not mistake the topology as that in Figure 4-b and should not try to route packets sent to R5 through A. To solve the problem, a large cost is assigned to virtual links connecting anycast nodes to their local networks, such that they are not used in building routes unless the anycast node is the destination.

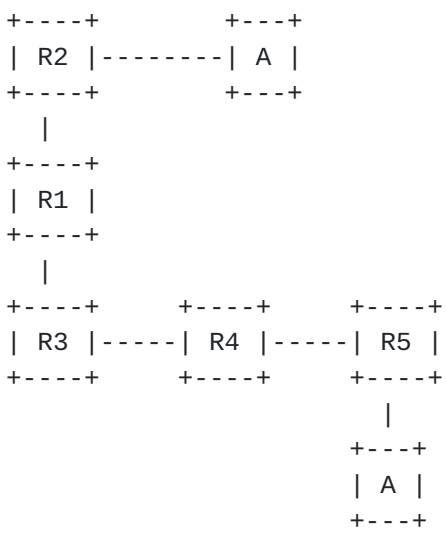


Figure 4-a

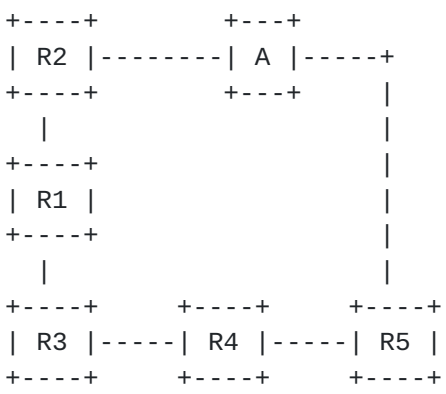


Figure 4-b

Figure 4: Applying the link state algorithm directly on the topology in 4-a may introduce false topologies

Internal groups are not advertised to other domains in the Internet. Sections 5.4.2 and 5.4.3 describe how users in other domains access those groups (For those users the groups are external.)

5.4.2 Routing External Unpopular Anycast Groups

In GIA, unpopular anycast groups need not be routed. Thus, if the number of unpopular anycast groups is considerably larger than the number of popular groups, the system, without degrading the service, can make a large saving by using cheap default sub-optimal routes to forward packets destined to unpopular anycast groups.

The basic characteristic of a default route is that it doesn't consume any bandwidth to generate, and doesn't need any storage space in the routing tables. To understand how such a route exists recall that an anycast address is a concatenation of the anycast indicator, the unicast prefix of the home domain and the group ID. Also, recall that the architecture requires the user who leased the anycast address to set up at least one machine with that address in the home domain. Thus, in the worst case any router that can distinguish anycast addresses can forward any anycast packet to its home domain. To do so the router shifts the anycast indicator off and forwards the packet according to its unicast routing table. Note that the router leaves the destination address in the packet intact; it only shifts the variable according to which it is looking up the address in its routing table.

Thus, a packet destined to an unpopular group is forwarded toward its home domain. However, depending on the popularity distribution of its corresponding group the packet follows one of three possible routes. First, if the packet crosses a domain that has a member of the anycast group then the packet is delivered to that member. Second, if the packet crosses a domain that has this group as a popular group and consequently knows a shorter route to a group's member then the packet continues its journey along the enhanced route. Finally, if neither of the aforementioned cases is encountered, then the packet eventually hits the home domain and is delivered to the nearest member there.

5.4.3 Routing External Popular Anycast Groups

At the core of GIA's architecture is generating routes to popular anycast groups. This task is performed by border routers of a domain and it is implemented as an integrated part of BGP (the Border Gateway Routing protocol). It is decomposed into 3 protocol-building blocks; monitoring popular anycast groups; learning an anycast route; and finally maintaining a learned route. We address each of these three components separately.

- Monitoring popular anycast groups

To monitor popularity, each border router keeps track of the number of times an anycast packet is forwarded along a default route. The border router keeps this information in a list of pending addresses. Periodically, the border router checks its list and decides on the most popular addresses to search for. This number should not exceed the maximum number of addresses that can be included in one search message. (This number is around 1000 and it is limited by the maximum size of a BGP message). Addresses included in the search are deleted from the list. Other addresses in the list have their popularity multiplied by an aging factor and are kept for consideration at the time of the next search. When the popularity of an address in the list falls below certain threshold the address is discarded. The time between two searches, which we call the search interval (SI), should be jittered to prevent synchronized search messages.

- Learning an Anycast Route

In contrast to unicast interdomain routing, which is based on advertising unicast prefixes to all Internet domains, GIA adopts an on-demand reactive inter-domain routing approach. The incentive for choosing a reactive approach is

the need for a design in which routers in the backbones do not store any external anycast routes. This objective makes it hard to design a proactive routing protocol. Consider unicast routing as an example. In unicast the routing information propagates from children domains to parent domains (up the hierarchy), then from parent domains to other children domains (down the hierarchy). Thus, for a downstream router to forward a packet along a certain route the upstream routers in the parent domain have to store the route. One can imagine a scheme in which upstream routers propagate reachability information without storing it, and downstream routers tunnel the packets to the router that generated the routing advertisement. However, such a design means that any routing advertisement is broadcast to all domains in the Internet because the upstream border routers can not tell whether they have already advertised the same or a better route for this address. A second less important factor for choosing a reactive approach is that the fact that an anycast group is replicated in multiple domains in the Internet increases the probability of finding the nearest group member by exploring a small neighborhood around the interested domain.

The route learning process makes use of the TCP connections a BGP router has with its peers [21]. It involves adding two new message types to BGP; the search message, and the reply message.

A search for a set of popular anycast groups is triggered by the exit BR (border router) towards the groups' home domains, which receives the anycast packets in the absence of a learned route. This BR, which we call the originator BR (OBR), monitors the popularity of the groups according to the scheme described in the previous section. At the beginning of a search interval (SI) the OBR generates a search message for all of the popular groups for which there is no learned route and broadcasts it to all of its peers. (Here 'peers' refers only to border routers whether they are internal or external peers. Some domains ran iBGP to disseminate external routes to internal routers. In this case, those peers should be excluded from the search.) Once the search is generated the OBR sets a timer and waits for replies.

The search itself is a scoped domain-by-domain broadcast that explores the neighborhood of a domain looking for members of the popular anycast groups. The search message whose format is shown in Figure 4 contains the IP-address of the OBR, a sequence number, a path vector which, at first, is set to the OBR's AS number (AS_Path field), and the set of anycast groups the OBR is searching for (Network

Layer Reachability Information field). In addition the message contains a TTL field, which is initialized to the maximum number of domain-hops the message is allowed to travel. Based on the study of the inter-domain topology provided in [10], the authors think that the TTL should be set to 3 or 4 domain-hops. This number was chosen because most edge domains are less than 3 or 4 domain hops from the backbone. The AS_PATH field is used to collect a vector of Autonomous Systems that separate the OBR from a domain that contains a member of a popular anycast group. It is also useful to prevent the search message from looping. On the other hand the TTL scopes the search to a certain neighborhood around the searching domain.

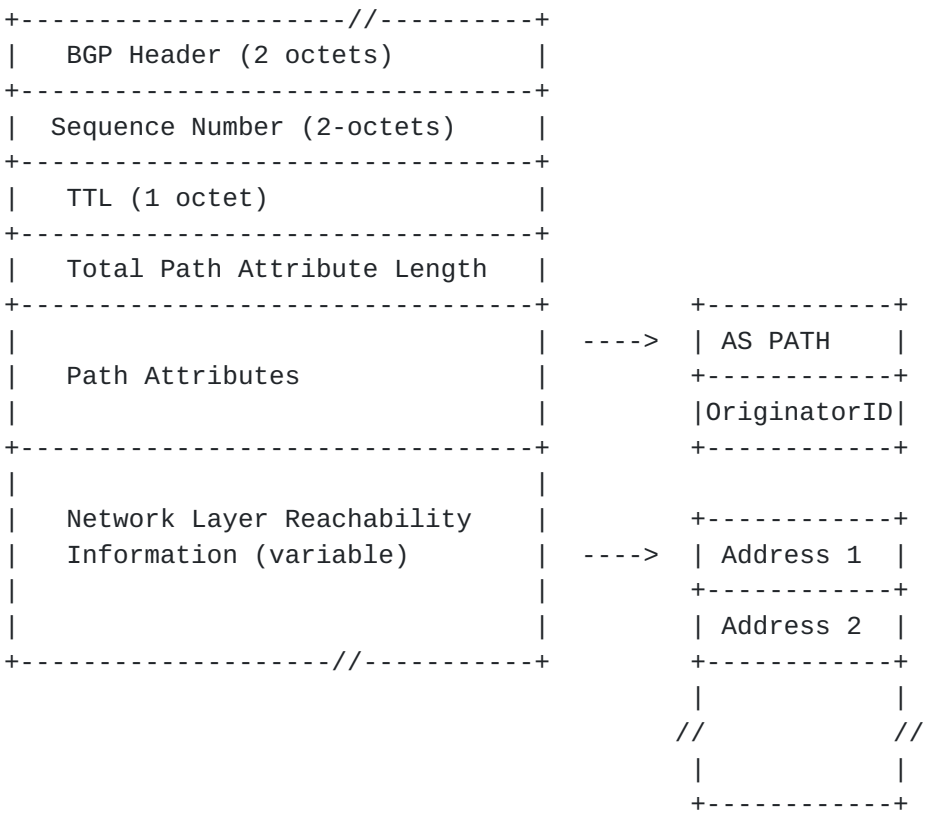
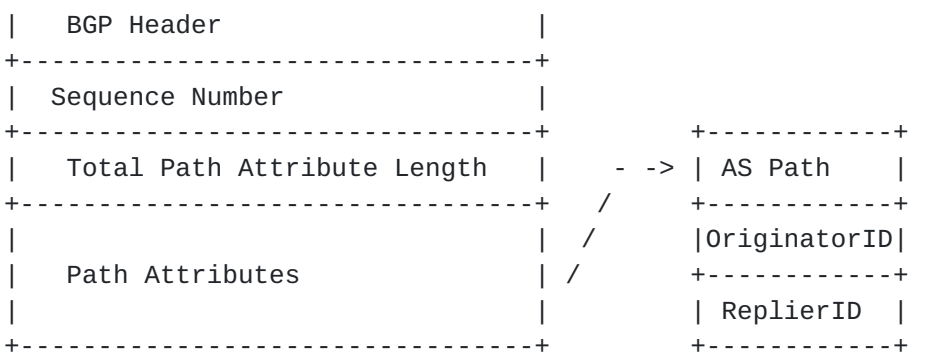


Figure 5: The Format of the Search Message



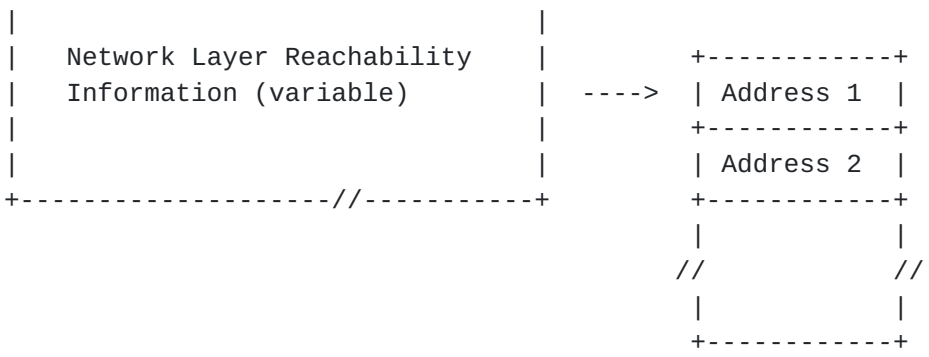


Figure 6: The Format of the Reply Message

A BR that receives a search message from an internal peer propagates the message to all of its external peers with no further processing. A BR that receives a search message from an external peer looks up the anycast addresses in the search message in its routing table. For all groups that are internal to the replying BR's domain, the BR sends a reply message, which relays the path vector in the original search message after appending the receiving BR's AS number. In addition, the reply includes the original search sequence number and the receiving BR's IP-address. The reply is sent directly to the OBR. Groups for which no internal member is found, are looked up in the set of learned anycast routes. For each anycast group for which the receiving BR has a learned route, it adds its AS to the vector-path in the search and concatenates the resulting AS-sequence with the vector-path associated with the learned anycast route, and sends this vector-path to the OBR in a reply message. The replier field in the reply message is set to the router from which the existing route is learned. All addresses for which the receiving BR is able to send a reply are removed from the search message. If there are still anycast groups to search for, the receiving BR decrements the TTL of the search, checks that the TTL did not reach zero, and propagates the search to all of its peers.

Finally, to prevent a search message from looping, GIA requires a BR that receives from an external peer a search whose vector-path include its own AS to ignore the search message. Moreover, to reduce the number of messages emanating from a search, we require each BR to maintain a table of all (OBR, sequence number, shortest vector-path up to present) heard of in the last two search intervals. For each search it receives, the BR propagates the search only if it contains a vector-path whose size is smaller than all the vector-paths with the same (OBR, sequence number) pair seen so far. Although, storing a table of (OBR, sequence number, shortest vector-path) consumes some memory at a

border router, the size of the memory needed is around 2*(the number of BRs in a neighborhood), which is much smaller than all anycast groups in the Internet. In addition, the lookups in this table are not on the critical path of unicast data packets.

After sending a search message an OBR sets a timer and waits for replies. When the timer expires, the OBR checks all the received replies and chooses the one that has the smallest vector-path. The OBR checks its list of pending addresses and deletes any address for which it found a route. If the list contains an address for which no route was found, the OBR multiplies the popularity of that address by a decaying factor to decrease it chance in being included in a new search.

The learned routes are kept in a cache of popular anycast routes. Also, the routes are advertised to all internal peers as if they were learned from a BGP update message. Depending on the domain's policy the routes might be injected into internal routers' routing table or kept only at border routers.

A stored external anycast route contains the path-vector, which describes the set of domains the route traverses, and the unicast address of the destination BR. Both are extracted from the reply message. The vector is kept to answer search messages issued by neighboring domains looking for a route to this anycast group. The unicast address of the destination border router (ReplierID Field) is used to tunnel all subsequent anycast packets to the neighboring domain that has a group member.

Figure 7 illustrates a possible scenario for a search message. The border router OBR in domain 1 sends to its peers BR2 and BR3 a search message solicitating routes for groups A and B. Since BR2 has a route for group B, it sends a reply back to OBR informing it of the availability of the route. However, since BR2 has no route for group A it propagates to its peer the search message, which now contains only a query for A. Eventually the search for A hits BR5 in domain 4, which has internally a group member. Thus, BR5 sends directly a reply to the OBR. When OBR's timer fires it examines the replies it received and decides that the nearest member for group B is through BR2 and that the nearest member of group A is through BR5. Having learned the routes, OBR tunnels subsequent packets addressed to groups A and B to BR5 and BR2 respectively, which decapsulate them and deliver them to the local members. It is also possible (though unnecessary) for OBR to inform its intradomain routing component (RIP for example) to inject the routes into

domain 1.

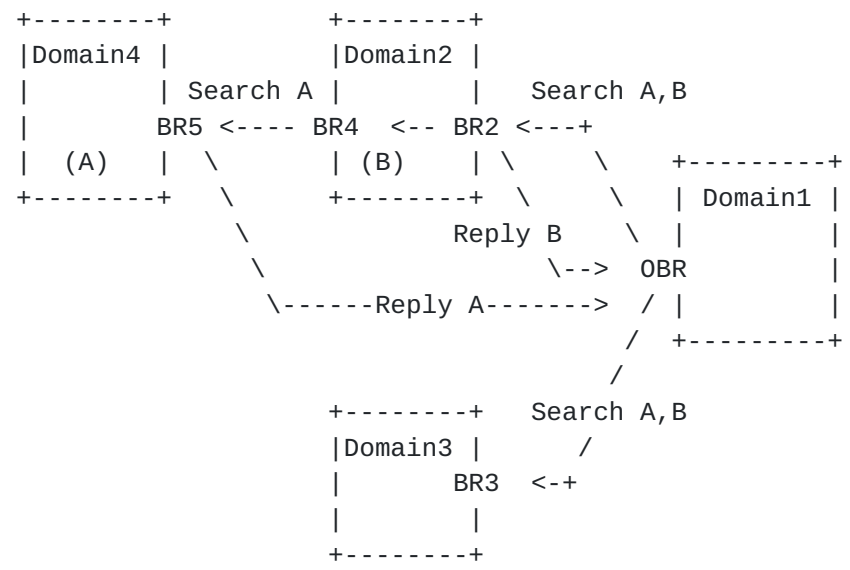


Figure 7: Learning routes to popular anycast groups

- Maintaining a Learned Route

A learned route becomes invalid in the following cases. First, when the originating domain loses connectivity to the destination domain. In this case, the OBR discovers the loss of connectivity from its unicast routing table and initiates a new search. The second case happens when the nearest anycast member crashes or leaves the group. In this case the originating domain can't discover the invalidity of the route directly and keeps tunneling the packets to the learned BR. However, when those packets get to the destination domain, the receiving BR discovers that there is no local anycast entry. Thus, it forwards the packets according to its best knowledge of the route. (Most likely the BR will forward the packets to their home domain. However, it might be the case that after the local anycast member crashed, the domain has learned a route to some other nearby member.) Also, it sends an ICMP message to the BR that tunneled the packet informing it of the invalidity of the learned route. A BR that receives such an ICMP treats the message similarly to a route withdrawal received via BGP. Finally, as a consequence of the route being withdrawn, the OBR schedules the group to be considered in the next search.

On the other hand, a learned route might be withdrawn even when it is still valid. This is necessary to allow caching of new popular anycast routes while maintaining an upper bound on the size of cached popular anycast routes. Thus, we require the exit border router toward the destination

domain to check a route level of usage and discards learned routes that are no longer popular. The algorithm for discarding learned routes that are no longer popular is an area for further research.

6.0 Discussion and Evaluation

- How does GIA affect the routing tables?

In contrast to the traditional approach for global anycast, where the routing tables grow proportionally to the total number of anycast groups, the growth in the routing tables in GIA is manageable. In fact, routers in an edge domain store routes to internal anycast groups and popular ones. Both numbers are controllable by the domain's administrator and should be much smaller than the total number of anycast groups. On the other hand, routers in the backbones, which usually maintain a large routing table only maintain anycast routes for their internal groups (if there are any).

In addition, the fact that anycast addresses are distinguishable from unicast addresses means that anycast routes can be maintained in their own table separated from unicast addresses. As a result, the existence of an anycast service does not slow down the unicast forwarding process. Moreover, the anycast routing table can be much simpler and allow faster search and insertion than the unicast routing table because it does not need to account for the longest match.

- What is the overhead of using GIA?

The control traffic overhead is dominated by the number of search messages, which is a function of the following parameters:
- The maximum number of domain-hops the search message explores (the TTL field in the search message).
- The network inter-domain topology specially the average edge degree.
- The number of popular anycast groups.
- The correlation of popularity in neighboring domains. (or in other words, the similarity in popular groups between neighbor domains)
- The number of members in an anycast groups and the groups' topology.

A thorough evaluation of the search overhead needs to investigate the effect of all of these parameters on the number of search messages, which is beyond the scope of this document. Instead, we try to provide some intuition

about why a search might consume less bandwidth than advertising the group (the proactive approach). One reason is that we search only for popular groups. A second reason is that we search only the neighborhood of a domain. The third and most important reason is that a search is more stable than an anycast advertisement through BGP updates. In other words, once we find a route, we don't need to perform a new search as long as the forwarding path does not change. Usually, the forwarding path stays the same for days [10,20]. Unfortunately, this is not the case for advertisement. The study of BGP routes in [15] shows that on average each prefix generates 100 updates a day at a BGP router. Thus, the frequency of search is multiple order of magnitude less than that of the equivalent advertisement.

The second source of overhead in GIA is the processing time of the control messages. The major concern here is that processing the search and reply messages might affect the BGP router performance and slow down its processing of the unicast updates. to prevent that, BGP routers might assign higher priority to processing unicast updates. On the other hand, note that processing search and reply messages is logically independent from processing the update messages and can be performed by a separate CPU. In addition, searches are allowed to explore only a limited neighborhood around an edge domain. Thus, only few of them reach the backbone and most of them are processed by routers at the edges of the network where the traffic is not as intense.

The last source of overhead is storage space. In addition to the anycast routing tables discussed above GIA requires a BR to store a list of (OBR, sequence number, shortest vector-path) seen in the past 2*Search_Interval seconds. However, this list is on the order of the number of border routers in a neighborhood, which is much smaller than a routing table.

- What is the average path length of an anycast packet in GIA to the shortest path, where the shortest path is the one found by unicast routing?

The routes to internal anycast groups are similar to those generated by unicast routing. Thus, for internal anycast groups, the average path length in GIA to the shortest path is equal to 1.

On the other hand, the path to external anycast groups, on average, is longer than the shortest path. The difference is due to the existence of packets destined to unpopular groups and to the possibility of a search failure.

To estimate 'Average[external path in GIA / shortest path]' we ran simulation on 100 graphs generated using GT-ITM network graph generator [24]. All graphs are generated using Transit-Stub edge connection method and have a size of 208-node, an average degree around 3, and a diameter of **11. These quantities (except the number of nodes) are** chosen according to [10], which studies the interdomain topology in the Internet. Members in an anycast group are assigned randomly to domains as long as no two members of the same group are assigned to adjacent domains. The home domain of an anycast group is also chosen randomly. Note that each node in this simulation represents a domain.

Our simulation shows that if the fraction of anycast traffic sent to popular group is 80% of all anycast traffic generated by the domain, and the search has a TTL of 3, then, as the number of the members in the anycast group vary between (0.005*number of domains) and all the domains in the simulation the below inequality stays valid:
1.05 < Average[external path in GIA/ shortest path] <1.15
Thus, the average performance of GIA is significantly good. (for further information about the simulation please contact the authors.)

Note that the simulation does not assume any correlation between where the service is popular and where the members exist. In practice, providers of an online service try to establish servers in network regions where the service is popular. This enhance GIAÆs performance further, but it is not a necessary assumption for the design to perform well.

- What is a typical working environment for GIA?

In practice, the authors think that there are going to be three major types of anycast groups: First, well-known groups that are used for auto-configuration purpose such as the one representing the set of DNS servers. These groups will be widely represented and internal to most domains and will cause hardly any searches. For the few domains that have to search for a DNS server the search will be satisfied in one domain-hop or at most in 2 domain-hops and the wide spread of the group will cause the messages emanating from a search to die close to their origin. Second, groups that represent an internationally replicated service such as the CNN web server. These groups will have a much smaller number of members distributed with reasonable spread in the Internet. Because these groups are popular in the majority of domains, a search usually succeeds in one or 2 domain-hops and generates few messages. Third, groups

that are regionally popular such as a local TV broadcaster. Most of the searches soliciting these groups will spring in their neighborhood and will locate them without involving the whole Internet in the search. Some search might spring from random regions in the network soliciting groups with random topology in which case the simulation results provided above give a rough estimate of the possible number of messages. Definitely, the above is not an enumeration of all possible group types. It is rather an attempt to understand GIAÆs working environment based on the currently proposed anycast applications.

7.0 Deployment issues

This section addresses the following two deployment issues:

- Changes to routers

To deploy GIA in a transit domain we need to change the border routers to participate in route learning and to change the internal routers to shift the anycast indicator off when they have no route to the anycast group. However, changing the internal routers is not crucial for the design. In fact, the same effect can be achieved by having the border routers inject the unicast interdomain routing information internally after shifting the anycast indicator in. We suggest this solution as an intermediate step until the domain upgrades the internal routers to understand the anycast address syntax.

On the other hand, deploying GIA in an edge domain requires integrating popularity monitoring, route learning, and route maintenance in the border routers. It also requires changing the internal routers to understand the syntax of an anycast address. However, most edge domains have only one border router, which makes it unnecessary to change the internal routers. When the internal routers receive a packet destined to an external unpopular anycast group they treat it as a unicast packet and realize that they have no route for it; thus, they forwarded to the border router. The border router, which is GIA-enabled, shifts the anycast indicator off and forwards the packet according to its unicast routing table. For the case of edge domains that have more than one border router, an intermediate stage similar to the one described for the transit domain case can be adopted.

If GIA is deployed in an IPv6 environment; therefore, the aforementioned changes can be incorporated to the routers while upgrading them to be IPv6 enabled.

- Crossing Regions that are not GIA-enabled

During the deployment phase, the Internet will contain both GIA-enabled and non-GIA-enabled regions. We would like a domain in a GIA-enabled region to forward packets addressed to an unpopular anycast group towards their home domain even if the home domain is separated from this domain by a non-GIA-enabled region. One possible solution is to configure the border routers at the periphery of a GIA-enabled region to encapsulate anycast packets leaving the region in unicast packets destined to the unicast address resulting from shifting the anycast indicator off. In addition, the border routers set the transport protocol field in the IP packet to a special protocol number that identifies these encapsulated anycast packets. The packets cross the non-GIA-enabled region safely heading toward the home domain. Once they cross the border of a second GIA-enabled region the border router recognizes them as encapsulated anycast packets. The BR decapsulates the packets, which now complete their path according to the scheme described in the previous sections.

8.0 Other Issues

- Scoped Anycast Addresses

The design in the above sections addresses only the case of global anycast groups. One can imagine providing a class of scoped anycast addresses by changing the anycast address syntax to the following:

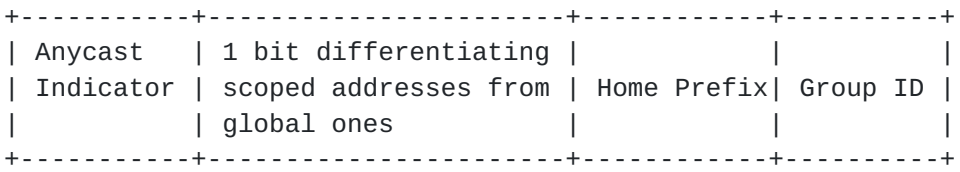


Figure 8: A possible change in the Syntax of the anycast address to provide scoped anycast groups

In this case the scope of the address is the home domain and the routers outside the home domain will never initiate a search for a scoped anycast group.
Another possibility is to use a sub-space of the domain's unicast address space for anycast groups whose scope is confined to the domain (in an approach similar to that proposed in IPv6).

- Long domains

The design assumes that the distance between any border

routers in a domain is roughly equivalent. However, this is not always the case. For example, a domain that connects Europe with North America would have BRs in both continents. It makes sense for a BR in Europe that received a search message to propagate the search only to its European peers. This issue can be addresses by providing the border routers in such domains with some proximity information to guide them in propagating search messages. The size of this information is on the order of the number of BRs in a domain. Thus, the authors think it is scalable. Besides, providing the BRs with this information is an easy task since the providers know these facts about their networks.

- A different measure of distance

As described above, the architecture computes the length of an external route in terms of the number of domain-hops. This is the measure of distance used by the unicast inter-domain routing. However, the architecture is flexible enough to handle different types of distance measures. For example, an ISP can occasionally run Ping between all pairs of border routers and store the results in a table at the border routers. A search message can collect this information and measure a path length using latency.

- The Transit Domain Case

The design as described in the previous sections carries the assumption that anycast packets addressed to external anycast groups are generated in edge domains. This assumption is needed because it is hard for Transit domains to monitor popularity. More precisely, in a transit domain, a BR forwarding an anycast packet along a default route can not decide whether the packet has been generated in the domain and consequently should increase the popularity of its group, or the packet is a transit one.
To address this issue we distinguish between two cases. In the first case, the transit domain contains hosts that access some external anycast groups. In this case the above assumption stays valid because the hosts are usually grouped on a local network that is connected to the core of the domain through one router. Therefore, the host network can be regarded as an edge domain given that we ran iBGP on the router connecting it to the core of the transit domain. In the second case routers inside the transit domain send packets to popular external anycast groups. Here, we need a mechanism to distinguish anycast packets generated internally from transit anycast packets. One possible solution is to have border routers tag transit anycast packets when they enter the domain such that they don't

interfere with the domain's popularity-monitoring task (yet this issue needs further study).

- Variations on the route learning protocol

One possible variation on the route learning protocol is to never discard previously learned anycast routes. These routes are deleted form the anycast routing table but stored in some secondary memory, which is not on the forwarding path. Later, if the group becomes popular again its route is reinserted into the routing table after checking that it is still valid.
A second variation is to do an expanding ring search instead of sending the search immediately three or four domain-hops.

9.0 Security Considerations

Security considerations are not discussed in this document (yet).

10.0 Acknowledgments

This work benefited from discussion with D. Clark and T. Shepard. In addition, B. Priyantha and R. Hariharan provided comments on the draft and M. Kasbekar and S. Mneimneh helped with the simulation. Finally, the authors would like to thank **C. Partridge, T. Mendez, and W. Milliken for writing RFC 1546.**

11.0 Authors' Addresses:

Dina Katabi
MIT Laboratory for Computer Science
545 Technology Square
Cambridge, MA 02139
nora@lcs.mit.edu
617-253-3147
617-253-2673 (FAX)

John Wroclawski
MIT Laboratory for Computer Science
545 Technology Square
Cambridge, MA 02139
jtw@lcs.mit.edu
617-253-7885
617-253-2673 (FAX)

12.0 References

1. E. Basturk, R. Haas, R. Engel, D. Kandlur, V. Peris, and D. Saha, "Using Network Layer Anycast for Load Distribution in the Internet". Global Internet, Dec. 1998.

2. S. Bhattacharjee, M. H. Ammar, E. W. Zegura, N. Shah, and Z. Fei, "Application Layer Anycasting". In Proc of INFOCOM'97, Apr. 1997.

3. J. Bound, P. Roque, "IPv6 Anycasting Service: Minimum Requirements for End Nodes". Work in progress.

4. M. Doar, "A Better Model for Generating Test Networks". IEEE Global Telecommunications Conference/GLOBECOM'96, London, Nov 1996.

5. D. Farinacci, L. Wei, and J. Meylor, "Use of Anycast Clusters for Inter-Domain Multicast Routing". Internet-Draft, Mar. 1998.

6. Z. Fei, S. Bhattacharjee, M. H. Ammar, and E. W. Zegura, "A Novel Server Technique for Improving the Response Time of a Replicated Service". In Proc. of INFOCOM'89, Apr. 1998.

7. P. Francis, "A Call for An Internet Wide Host Proximity Service (HOPS)," Aug. 1998.

8. P. Francis, "Pip Near-term Architecture" May 1994.

9. P. Francis, S. Jamin, V. Paxon, L. Zhang, D. F. Gryniewicz and Y. Jin, "An Architecture for a Global Host Distance Estimation Service". In Proc of INFOCOM'98, Apr. 1998.

10. S. V. Fuller, T. Li, J. Yu, and K. Varadhan "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation," RFC1519, Sep. 1993.

11. R. Govindan and A. Reddy, "An Analysis of Internet Inter-Domain Topology and Route Stability". Technical report USC-CS-96-642, department of computer science, University of Southern California, In Proc of INFOCOM'97.

12. J. D. Guyton and M. S. Schwartz, "Locating nearby Copies of Internet Servers". In Proc. of SIGCOMM'95, Aug,1995.

13. R. Hinden, S. Deering, "IP version 6 Addressing Architecture". RFC 2373, July 1998.

14. R. Hinden, "Simple Internet Protocol Plus," White Paper, RFC1710, Oct. 199

15. J. Itoh, "Disconnecting TCP connection toward IPv6 anycast address". Work in progress, Oct. 1998.

16. D. Katabi, " The use of IP-Anycast to Construct Efficient Multicast Trees," Master Thesis, Sep. 1998.

17. C. Labovitz, G. R. Malan, and F. Jahanian, "Internet Routing Instability," In Proc of SIGCOMM'97, Sep. 1997.

18. K. Moore, J. Cox, and S. Green, " Sonar - a Network proximity Service," Internet-Draft, Feb 1996.

19. C. Partridge, T. Mendez, and W. Milliken, "Host Anycasting Service," RFC1546, Nov. 1993.

20. V. Paxon, " End-to-End Routing Behavior in the Internet," In Proc. of SIGCOMM'96, Aug. 1996.

21. Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)," RFC 1771, March 1995.

22. M. Stemm, R. Katz, and s. Seshan, "Shared Passive Network Performance Discovery."

23. J. Veizades, E. Guttman, C. Perkins, and S. Kaplan, "The Service Location Protocol," RFC 2165, June 1997.

24. W. Zegura, K. Calvet, and S. Bahattacharjee, "How to Model an Inter-network," In Proc. of INFOCOM'96, Apr. 1996.

25. T. Narten, E. Nordmark, and W. Simpson, " Neighbor Discovery for IP Version 6 (IPv6)," RFC 2461, Dec. 1998.

26. W. Fenner, "Internet Group Management Protocol, Version 2," RFC 2461, Nov. 1997.