Deploying Lightweight Anycast Services Based-on Explicit Multicast Routing for Evolved Internet

Wen-Kang Jia
College of Photonic and Electronic Engineering
Fujian Normal University
Fujian, China
wkjia@ fjnu.edu.cn

XueFeng Dong
College of Photonic and Electronic Engineering
Fujian Normal University
Fujian, China
xfdong@fjnu.edu.cn

***Abstract*—** As a one-to-one-of-many communication method, anycast provides a dependable service framework for clients to select one of the nearest servers in an anycast group in evolved Internet. Deploying the Internet-scale application-layer anycast is proven to be infeasible unless the scalability issues are satisfied. To handle these challenges, we focus on application-layer anycast. Proposed framework is a comprehensive solution which allows an anycast service has its unique domain-name (a.k.a FQDN), which is associated to multiple replicated servers' unicast addresses in the worldwide. With DNS query/reply service, and through Explicit Multicast (Xcast)-based replica probing, ranking, and selection procedures, clients finally reach the nearby server. Compared to previous works, proposed solution has lowered the address resolution latency, replica probing latency, connection establishment latency, and signaling cost, while simplifying the anycast deployment by having it use a large number of medium-scale anycast groups. Besides, the deployment complexity of both anycast clients and servers, are expected reduction.

Keywords— Anycast; DNS; Internet; Explicit Multicast (Xcast)

I. INTRODUCTION

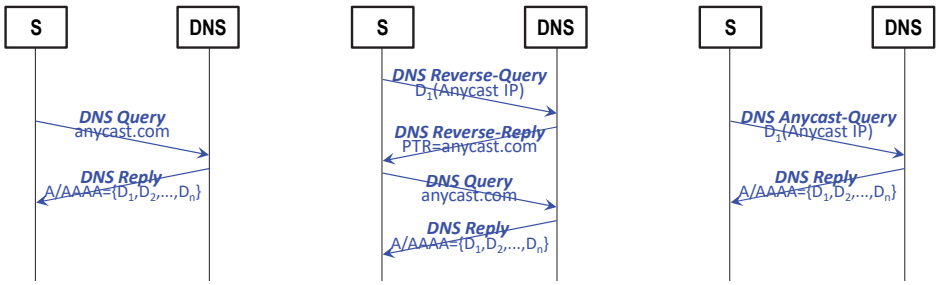
An Internet-scale anycasting service is an important building block for various IP-based dependable service framework and distributed applications. Since it was originally proposed in RFC 1546[1] during the early 1990s, IP anycast has been a promising technique for simple, efficient, and robust discovery and access of network services. IP anycast is an novel IP addressing mode whereby multiple geographically disperse servers are response to the same IP anycast address and provide the same service, with the result that IP datagram from a client will reach to optimal one of the servers, ideally the "topologically nearest available" to the client. In general, the client of the service does not care which instance of the anycast group it contacts. The specific rules in optimal anycast server are expressed principally in terms of specific performance index, e.g., end-to-end latency, routing hop-counts, server available/load, and network bandwidth/throughput, etc.

However, anycasting faces several challenges in the current Internet: Firstly, it should be scalable in terms of either number of groups or group size. Secondly, it must minimize the anycast query latency. And thirdly, the selected anycast destination must be optimizing to the client. The researches on anycast can be classified into two categories: network-layer and application-layer [2-4]. For economic reasons, and because it is believed that the amount of routing state needed to fully deploy anycast is infeasible in current Internet, network-layer anycast has been deployed only in private IP networks. As mentioned above, either of which restrict and hinder the application and development of network-layer anycast services on the Internet.

II. PROPOSED SCHEME

A new scheme on implementing dependable service in IP networks is proposed in this article. The essence of proposed framework is based on DNS services [5] and Xcast-enabled networks, which consists of two major components: 1) a DNS for providing anycast domain-name/address resolution function, and (2) a middleware of client or proxy server for providing anycast replica probing, ranking and selection function. Both above components couple smoothly to the anycast client and provide fast anycast access capacity between client and optimal one of anycast servers. It is also capable of supporting the Internet-scale anycasting by accessing the *Fully Qualified Domain Name (FQDN)* of anycast group, instead of anycast address.

Whereas network-layer anycast support relying on the use of anycast addresses, our application-layer anycast support makes directly use of anycast domain-names (ADNs). An ADN uniquely identifies a collection of unicast addresses and potentially dynamic, which constitutes an anycast group. An anycast domain-name/address resolution function is to configure a DNS server with an ADN that maps to multiple unicast addresses, each representing a unique destination containing the duplicated content. The application-layer anycasting service is thus to map an ADN into one or more IP unicast addresses. When a client queries the DNS server to resolve an ADN, the DNS server indicates a client to the entire unicast address list. An important feature of application-layer anycasting is that it does not require modifications to network layer operations. In other words, since anycast addresses are part of the IP address space, proposed framework can be allowed an anycast group to have its own unique FQDN and without occupied any IP anycast address over the Internet. Theoretically, the quantity of anycast group (including the massive FQDN and its sub-domain) and group size per anycast are seemingly infinite. Therefore, proposed scheme not only relieves IPv4 address space exhaustion problem but also solves scalability problem of anycast on the current Internet.



(a) Type-I: *domain-Name to Unicasts (N2U)* query

(b) Type-II: *Anycast to domain-Name to Unicasts (A2N2U)* query

(c) Type-III: *Anycast to Unicasts (A2U)* query

Fig. 1. The 3 query-types of anycast location discovery through DNS service.

While traditional multicast [6] schemes are scalable for very large multicast groups, they have scalability problems for large distinct multicast groups. Since some multicast services are required only for the close collaboration of small teams, eXplicit multicast (Xcast) [7] was proposed as a new multicast category with complementary scaling properties. Xcast achieves this by encoding the list of destinations explicitly in the packet header, instead of using a multicast group address. Thus, based on existing unicast routing information, the intermediate router can be stateless for forwarding, instead of relying on multicast routing information. Xcast can efficiently support a very large number of distinct small multicast groups and thus can play an important role in making multicast applications applicable for small groups [8].

Inside the anycast client of proposed scheme, an anycast middleware alters the TCP/IP model by replacing between the application and transport layers with middleware protocols. Anycasting middleware protocols abstract away from the difficulties of using raw anycast, and allow high-layer TCP and UDP connection to occur.

A. Service Location Discovery

To acquire the entire candidates' locations of all anycast replicas is the preliminary stage of the application-layer anycast. In contrast to IP-based communication model of "address-centric", an IP anycast address is utilized to identify an anycast group. By reason that a separate anycast address class was discussed previously [9]; considering that the current constraints on the available IPv4 address space, that is not likely to happen. Therefore, anycast addresses are suggested to be allocated from the available unicast address pool. In the Internet, when a unicast IP address is shared by many hosts, it is known as an anycast address. Accordingly, it cannot identify the address as anycast by itself.

To resolve the anycast address into its corresponding unicast address list in proposed scheme, we can make each anycast group own its unique ADN in the Internet. The DNS service translates human friendly names into IP addresses. DNS uses a hierarchical system where clients consult recursive DNS resolvers that identify and query an authoritative DNS resolver to discover the translated IP address. Using existing DNS resolution service, the DNS server might response multiple A (stands for IPv4 Address) or AAAA (stands for IPv6 Address) records for a domain name query, with different IP addresses. In other words, the value of an A/AAAA record is always an IPv4/IPv6 address, and multiple A/AAAA records can be mapped for one ADN. Therefore, an ADN represents an associated set of unicast addresses. If the DNS server then alternatively rotates addresses for any one FQDN that has multiple A/AAAA records, it is known as DNS round-robin, which is a mechanism of load balancing, or fault-tolerance provisioning multiple, redundant IP service hosts by managing the DNS's responses to address requests from clients according to an appropriate distribution model. In proposed scheme, we adopt the characteristic of DNS service to resolve the multiple servers hosting the anycast domain's DNS settings, which we named Type-I anycast Address Resolution — "*domain-Name to Unicasts (N2U)*", as shown in Figure 1(a). Note that there is no need to change the original DNS.

In some situations, clients might prefer to use anycast address instead of ADN. A reverse DNS lookup [9] is the querying of the DNS to determine the domain name associated with an anycast address. The process of reverse resolving an anycast address uses PTR (Pointer) records. Two steps are involved in obtaining an anycast address list from a DNS server: the process actually first contacts the DNS server to obtain the ADN that matches the anycast address client provided; then the client uses that ADN to obtain the unicast address list that matches the domain-name. Figure 1(b). shows the two-phase anycast address resolution by means of a small example, we called Type-II anycast address resolution — "*Anycast to domain-Name to Unicasts (A2N2U)*", there is also no need to change the original DNS.

Consider that not all IP addresses have a reverse entry. Sometimes clients need DNS respond more effectively and directly to queries. In this case, the DNS servers need some modifications: They should introduce a new type of query called "Anycast Query" in the DNS service. Resource records would record the anycast address for a unicast addresses with the Type fields set to "anycast address". While a clients will request the unicast address list in their DNS query by given anycast address. Once if the DNS is anycast aware, it will reply the unicast address list to the client. We called Type-III anycast address resolution — "*Anycast to Unicasts (A2U)*" and illustrated in Figure 1(c). In either situation, the client gets at least the unicast address of the desired anycast group that it can use for the further connection. This ensures that the anycast group member will be incrementally deployable.

B. Replica Probing, Ranking and Selection

Following the anycast address resolution operation, the most critical step in the application-layer anycasting is the anycast replica probing, ranking, and selection of a server amongst an anycast group that provide equivalent content. If the selection is done well the client will experience improved performance including low-latency and high-throughput. Thus the client can communicate with an optimal anycast replica chosen from multiple anycast servers by specifying the ADN or address. As mentioned previously, a client queries a well-known DNS server for a list of registered anycast servers' unicast address, and then it need to probe each listed anycast server in turn prior to the start of communication. This process emits thousands of probe packets over a short period of time. A client informs individual availability of anycast server from the reply packet, and estimates the Round Trip Time (RTT) to each anycast server based on the time between emitting a probe and receiving a reply. Note that the probe procedure requires additional network bandwidth to probe packets, which wastes network resources. To resolve the performance problems associated with a lack of that probing the corresponding unicast addresses one by one, we can used Xcast-based Internet Control Message Protocol (ICMP) [10,11] manner to evaluate each candidates at once.

Since the anycast address should not be set in the source address of the packet header, the client (source) enumerates the list of anycast servers' (destinations) unicast address in the Xcast extension header of the ICMP ECHO request packet instead of the anycast address, and then sends the packet to an intermediate Xcast-enabled router. The quantity of unicast address list is much larger than the theoretical maximum number of receivers, 374 and 92, in Xcast4 and Xcast6 respectively. Each Xcast-aware router along the way parses the Xcast header of ICMP ECHO request packet, lookups all the destinations in its unicast routing table, then partitions the destinations based on each destination's next hop, and duplicates and forwards a regrouped Xcast packet with an appropriate Xcast header to each of the next hops. When there is only one destination left, the Xcast packet can be converted into a normal unicast packet, which can be unicasted along the remainder of the route. Therefore, each anycast membership server that receives the ICMP ECHO request packet with its unicast address finally. Each anycast server would then respond ICMP ECHO reply using its unicast address [12]. For a client that may receive multiple replies—one reply from each available anycast server. Thus, the client can measure each anycast server's distance with proactive step for once, and select an optimal anycast server amongst an anycast group with appropriate selection algorithm (e.g., minimum latency). Typically, the client accepts and uses the first ICMP ECHO reply received as the selected optimal anycast server, something which was previously infeasible without the continual use of expensive ICMP tests or access to generally restricted routing information. Once the IP-based communication session is established, the client and selected server can exchange data in both directions until the connection between each is terminated by either side. In this way Xcast routing is used for initial ranking and selection of anycast replica, there is no need to modify the protocols to ensure that communication works properly.

Figure 2. depicts a reference network that consists of entire anycast replicas $\{D_1, D_2, \ldots D_n\}$, multiple Xcast-aware routers, a DNS server, and a client *S* that is fining a nearby anycast server. In this scenario, client *S*, sends an ICMP ECHO request to probe an entire anycast group of {*anycast.com*} simultaneous.

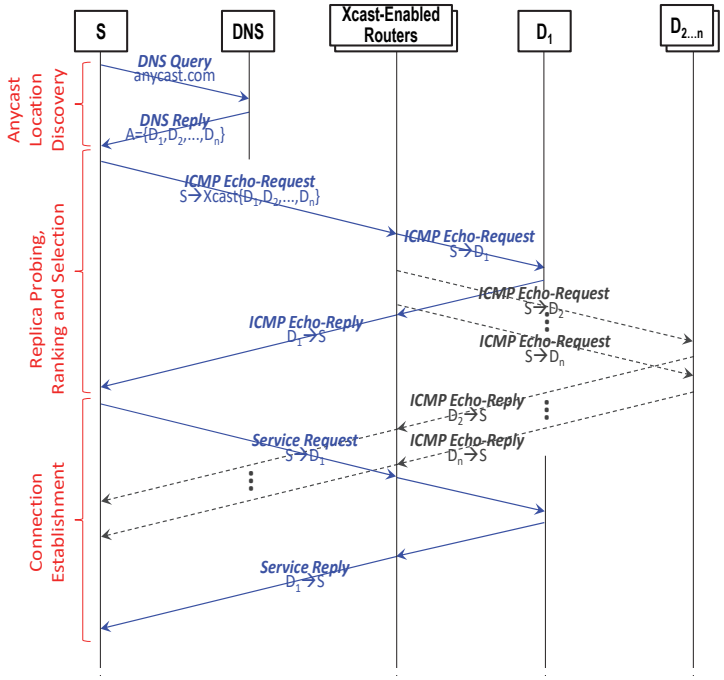


Fig. 2. Anycast replica selection by Xcast-based ICMP probing.

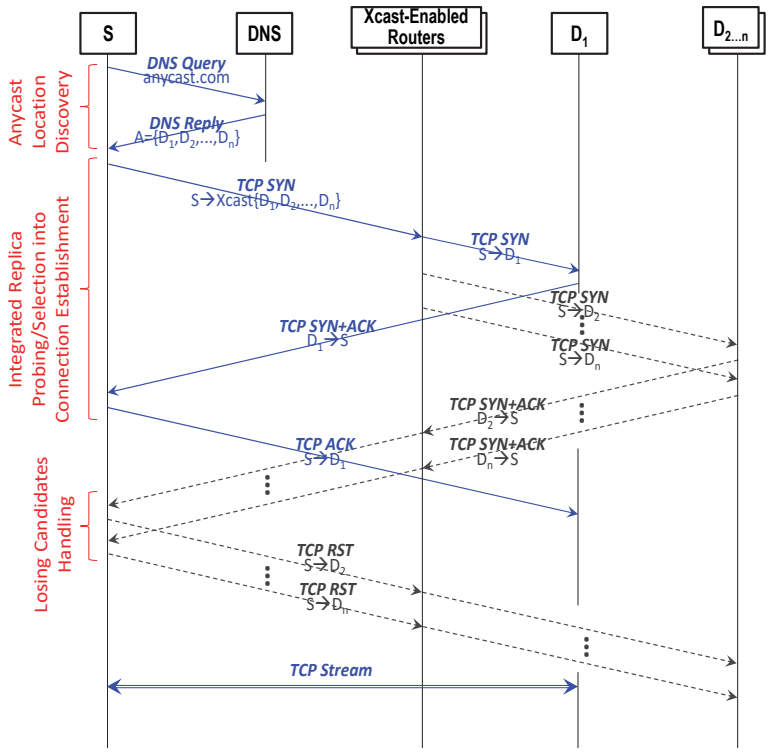


Fig. 3. Integrated anycast replica probing/selection into anycasted TCP connection establishment.

The ICMP ECHO request message is then duplicated and delivered to all anycast replicas through xcast-award routers, and replies with multiple anycast replicas by turn. Client *S* selects anycast server D_1 as an optimal communication party. In this model a participating application (e.g., Xcast and ICMP protocol stacks) must be modified to use the ADN or address mapper to resolve multiple unicast addresses before beginning communication with an anycast server. And there is no need to modify the TCP/IP protocol stack. For either stateful or stateless communications to be used with anycast, it seems one of the above solutions or something similar will need to be implemented and released.

Our approach allows a client to obtain the all Round Trip Time (RTT) of anycast replicas by using ICMP with Xcast routing protocol. Compared to unicasted ICMP probing, Xcast-based ICMP probing can quick determine the nearby replica from the desired anycast group. Consider a sequential unicast ICMP probing sends ECHO request message to all destinations in turn, and records each RTT, which are the time between an ICMP ECHO request and its reply. To determine the minimum among a series of RTTs may not be as accurate as expected, because the unicasted ICMP probing is not complete until all replies from each anycast replica to the probe have been received. Note that each replica may also timeout if there is no response. The selected server may not be optimal for the client selects the server that replies first. As with Xcast-based Probing, the client can communicate the server as soon as it receives the first reply. Besides, both unicasted and Xcast ICMP probing, the accurate RTTs still are affected by the hop queueing delay variations along an Internet path.

C. Integrated Replica Probing/Selection into TCP Connection Establishment

To accelerate the start-up of anycast communication, an alternative solution is proposed—integrated anycast replica probing/selection into connection establishment. Such a scheme will enable the anycast replica probing/selection operation to integrate with existing anycast connection establishment operation by means of a network harnessed by the Xcast-based TCP connection establishment. The two operations should be piece together without a break, thus startup latency of anycasting TCP is hugely reduced compared.

A client sends a TCP SYN connection request with its source address *S* to an anycast service based on Xcast extension header, which containing the entire unicast address of corresponding anycast group $\{D_1, D_2, \ldots D_n\}$. Similar previous manner, the Xcast-based TCP SYN massage is then routed using Xcast-award routers to all the anycast membership servers identified by the individual unicast address. The assumed nearest anycast replica responds using its unicast address D_1 in the TCP SYN+ACK message that first arrive at the client *S*. The client now sends the TCP ACK message to the nearby server D_1 in response to a TCP connection confirmation and completes the TCP connection setup with selected replica D_1. If the TCP connection is successful established with the optimal replica D_1, following TCP SYN+ACK messages from the remainder of anycast replicas $\{D_2, D_3, \ldots D_n\}$ are rejected (or ignored). In this case, each subsequent returned TCP SYN_ACK messages are answered by client with the TCP RST messages, which will terminate the remainder connection immediately and are stateless, namely the losing candidates

handling. The detail flow diagram of proposed manner is shown in the Figure 3.

III. Performance Evaluation

In this section, we compare our proposed scheme with unicasted replica probing/selection mechanism. The main concept of proposed scheme is that the source (prober) will obtain multiple ICMP echo replies from a single Xcast-based ICMP echo request. So it can choose the first replier as the optimal replica, which aims to accelerate and accurate the judgment of the optimal anycast replica. Apparently, the one-by-one unicasted replica probing/selection scheme will cause long waiting time to identify an optimal replica.

In order to simulate networks realistically, we must obtain the topology of the Internet first. We randomly select 10,000 real IP addresses as destination nodes and send the trace-route request toward them. If the nonrepetitive site replies the trace-route request without error messages or loops, it will be regarded as reliable topology information. Such IP addresses are randomly spread in the range of legal IP address space (reference to Internet Assigned Numbers Authority (IANA) [13]). We then assign arbitrary combination of destinations to each anycast group for observing the probing and selection time. Figure 4 shows each probing and selection time (ms) in two different approaches. We can see that the unicast probing generates the largest probing and selection times. The reason is that the source always has to wait for the last response from all destinations and then decide the fastest (the smallest interval) one among all the destinations. By contrast, proposed Xcast probing generates the shortest response time (in millisecond). For each probed message is sent repeatedly by the sequential unicasted ICMP until the expected reply message arrives or the inactivity timeout expires, the procedure incur more transmission latency. Therefore, with the anycast group size increases, the performance of our proposed scheme performs better and becomes more obvious especially in larger group.

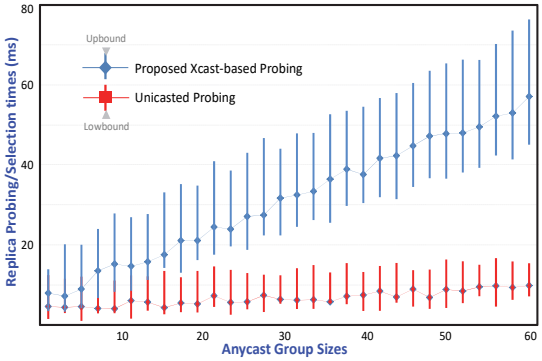


Fig.4. The anycast replica probing/selection times of two approaches with various anycast group sizes.

Proposed scheme also aims to lower down the signaling overhead of replica probing, but it does not consider the fairness of service time for each application server. The signaling costs (amount of transmissions) of simulations are calculated as total traffic volume of transmissions and retransmissions for ICMP probe messages around whole network. It incurs long latency, heavy network bandwidth consumption, large processing and memory overhead per client, server, and router. Figure 5 shows the total signaling cost by traffic volume (in bytes) in two different approaches. We can see that the scheme of unicast generates larger traffic volume than that of our approach. Proposed Xcast-based scheme generates smaller traffic volume. The curve of total traffic volume is similar to that of total hop count, which is resulted from the total traffic volume that is the sum of each hop data volume.

The accuracy of a unicasted probing/selection scheme does not take into account the ability of the perfect selected to the nearest server as it has infinite time, the accuracy cannot permanently be guaranteed in a limited time. Therefore, we can guarantee the given accuracy of solutions for some limited time only. In order to test the accuracy abilities we setup four different simulations with 20, 30, 40, and 50 timeouts set, respectively, to ensuring that the prober goes to complete state as soon as probing timeout expired, without waiting for the contingency ICMP replies because some replicas may be never respond. Figure 6 shows the relation between the accuracy and various group sizes (number of addresses per anycast group) and timeouts of two approaches. With the increasing of anycast group size, the latencies are becoming higher, which bounded by different timeout. Therefore, the accuracy of unicast probing will be affected by the group size and probing timeout.

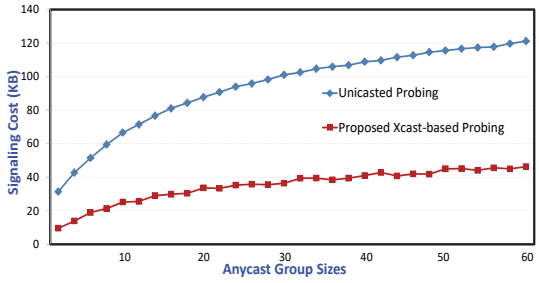


Fig.5. The signaling costs (traffic volumes) of two approaches with various anycast group sizes.

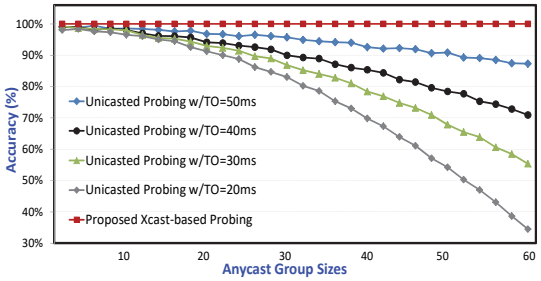


Fig.6. The average accuracy comparison with two approaches with various replica group sizes.

IV. Conclusions

Today, implementing native network-layer IPv4 anycast service is still a theoretical concept and cannot be implemented on the real Internet because implementing this service requires special agreements between ISPs to allow such traffic to pass between them just like multicast. Also, network-layer IPv6

anycast requires central management to allocate an anycast address and provide a particular mechanism to use it just in an ISP-scale network. Therefore, to widely supporting such a service is still limited in some Internet-scale routing protocols. Based on these situations, a comprehensive application-layer anycasting framework that combines standard DNS anycast location discovery and Xcast-based replica probing/selection is presented, and it can be used to provide either IPv4 or IPv6 anycasting with the same performance as a native anycast service. Proposed scheme was compared to ideal implementation of traditional unicasted probing and selection scheme using simulation. Simulation results show that proposed scheme achieves better performance than the original unicasted replica probing/selection scheme under different network conditions, in terms of the replica probing/selection times, signaling cost, and accuracy. Besides, the deployment complexity of proposed scheme is expected reduction, in term of unmodified DNS service and anycast server platforms, and minor-revised protocol stacks of clients and network equipment. Hence, the proposed scheme is completely simplified, supports a wide range of anycast operating. Although Xcast using by proposed scheme is not yet fully applicable to the current Internet at large, it could be deployed on large-scale enterprise, campus, datacenter, service provider, and telecom networks. It still can be a real alternative for anycast service with extra features in near future Internet.

ACKNOWLEDGMENT *(Heading 5)*

This work was supported by the special funds of the National Natural Science Foundation of China (Grant No. U1805262 and No.61871131).

REFERENCES

[1] C. Partridge, T. Mendez, and W. Milliken, “Host Anycasting Service”, IETF RFC 1546, November 1993.

[2] M. Hosseini, et al. “A survey of application-layer multicast protocols,” IEEE Commun. Surveys Tuts., vol. 9, no. 3, pp. 58–74, Sept.2007.

[3] S. Weber and Liang Cheng, “A survey of anycast in IPv6 networks,” IEEE Communications Magazine, Vol. 42, Issue 1, pp. 127–132, January 2004.

[4] H. Ballani and P. Francis, “Towards a global IP anycast service,” In Proc. of ACM SIGCOMM, August 2005.

[5] P. Mockapetris, “Domain Names - Concepts and Facilities,” IETF RFC-1034, November 1987.

[6] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C.-G. Liu, and L. Wei, An Architecture for Wide-Area Multicast Routing, In Proceedings of SIGCOMM '94, London, U.K., September 1994.

[7] R. Boivie, N. Feldman, Y. Imai, W. Livens, and D. Ooms, “Explicit Multicast (Xcast) Concepts and Options,” IETF RFC-5058, November 2007.

[8] W.K. Jia, “A Scalable Multicast Source Routing Architecture for Data Center Networks,” IEEE J. Select. Areas Commun. Vol. 32, Issue 1. December 2014.

[9] C. Partridge, T. Mendez, and W. Milliken, “Host Anycasting Service,” IETF RFC 1546, November 1993.

[10] J. Postel, “Internet Control Message Protocol,,” IETF RFC-792, September 1981.

[11] A. Conta, S. Deering, and M. Gupta, “Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification,”. IETF RFC-4443, March 2006.

[12] M. Oe and S. Yamaguchi, “Implementation and evaluation of IPv6 anycast,” in Proceedings of the 10th Annual Internet Society Conference, 2000.

[13] Internet Assigned Numbers Authority (IANA). http://www.iana.org/