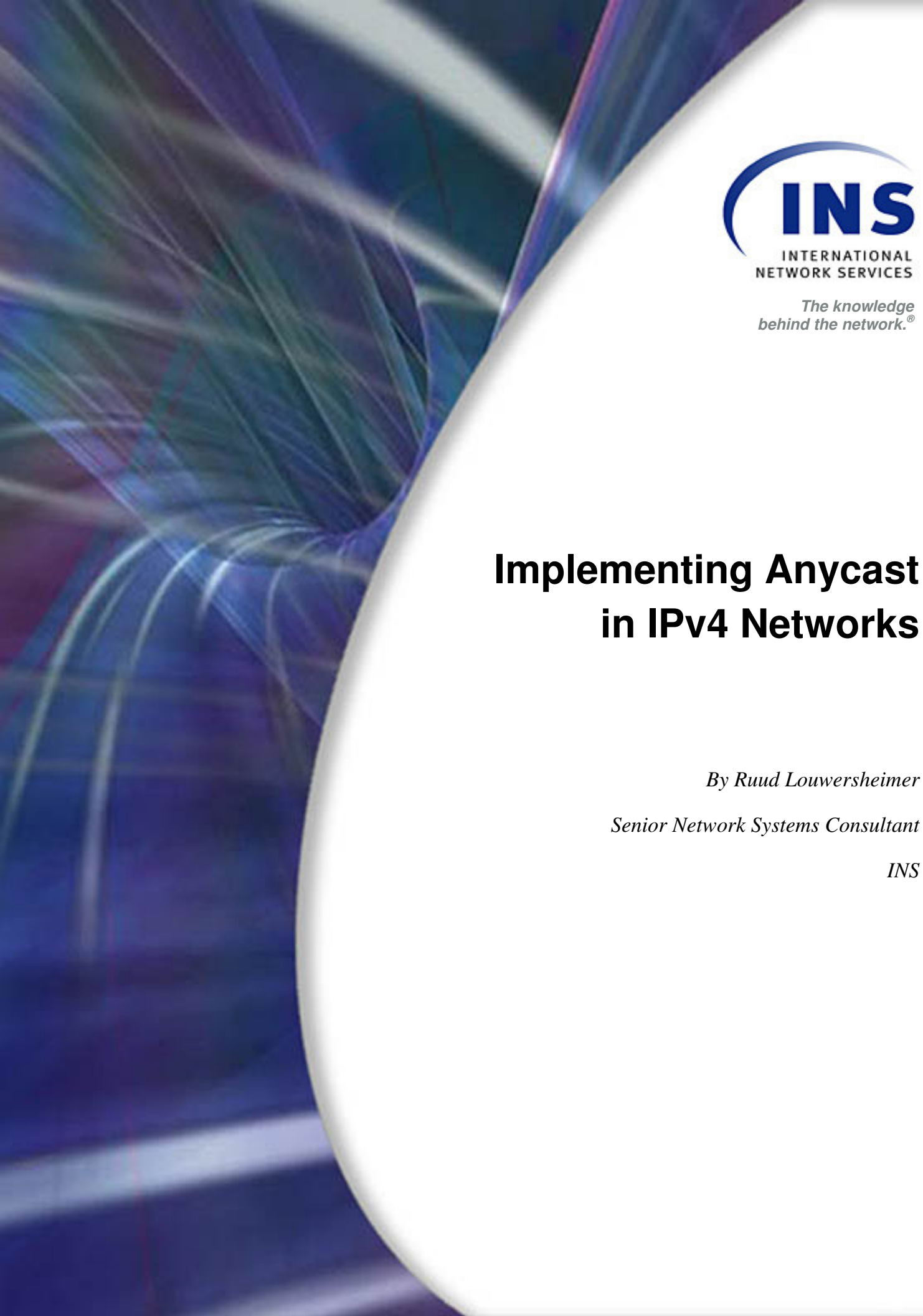


INS
INTERNATIONAL NETWORK SERVICES

The knowledge behind the network.®

Implementing Anycast in IPv4 Networks

By Ruud Louwersheimer

Senior Network Systems Consultant

INS

Implementing Anycast in IPv4 Networks

By Ruud Louwersheimer, Senior Network Systems Consultant

Introduction

Only three connection types are commonly known and used in Internet Protocol version four (IPv4) networks: unicast, multicast and broadcast. A fourth connection type, Anycast, was unknown until IPv6 made it a standard connection type. Anycast is not standardized in IPv4 but can be emulated. IPv4 Anycast addressing is a good solution to provide localization for services and servers in order to obtain robustness, redundancy and resiliency.

The basic idea of Anycast is very simple: multiple servers, which share the same IP address, host the same service. The routing infrastructure sends IP packets to the nearest server (according to the metric of the routing protocol used). The major benefits of employing Anycast in IPv4 are improved latency times, server load balancing, and improved security.

This white paper will explain the basics of Anycast and why networking organizations should consider implementing it in their IPv4 networks. Anycast IPv4 implementations usually involve enabling routing on the server or setting up a static route at the router connected to the server. Each solution has its own pros and cons. Tradeoffs are made according to the situation where Anycast is to be used.

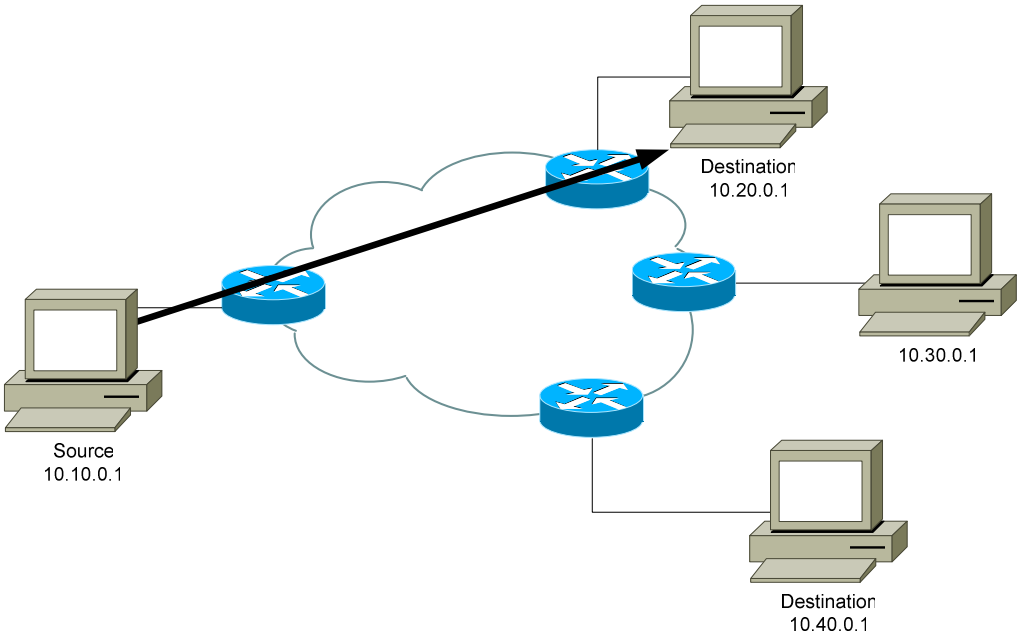
Methods of Communication

The three most well-known methods of making a connection in IPv4 are unicast, multicast, and broadcast. IPv6 introduced a fourth method of communication, Anycast, which IPv4 adopted.

Unicast

Unicast is defined as a point-to-point flow of packets between a source (client) and destination host (server). The server in Figure 1 is identified by a unique IP address (10.20.0.1), which is contained in the header of each packet sent from the client. The client is also identified by a unique IP address in the packet header. The network will then make a best-effort attempt to deliver the packet to the destination server identified by this unicast address. This best effort attempt is based on the information in the unicast's routing tables of routers in the network. Email delivery using SMTP (Simple Mail Transfer Protocol) and FTP (File Transfer Protocol) are two examples of unicast applications. Unicast addresses can be both a source and destination IP address.

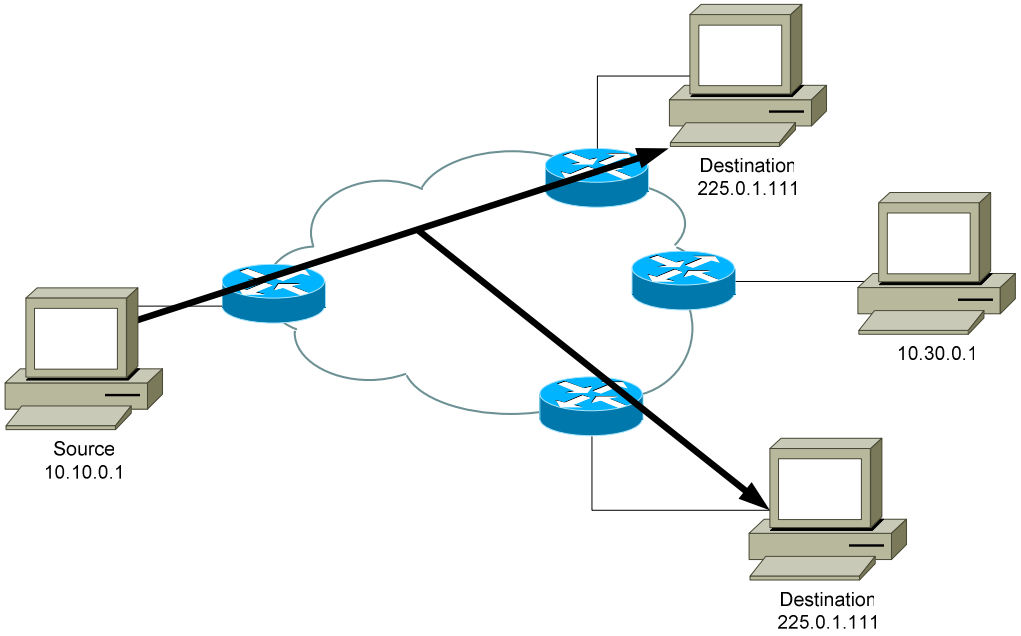
Figure 1: Unicast Connection



Multicast

Multicast is defined as a point-to-multipoint flow of packets between a single-source server and one or more destination clients (Figure 2). Rather than send a copy of the same packet to the unicast address of each destination client, the source server sends a single copy of the packet to a group address (e.g., 225.0.1.111). A special block (224.0.0.0 – 239.255.255.255) in the IPv4 address range has been set aside for multicasting. Any number of unique destination clients that wish to receive the multicast packet will be configured with a multicast group address. The network of routers will then make a best-effort attempt to deliver the multicast packets to all destination clients identified by the multicast group address. The routers in the network use a special multicast routing table to route the traffic through the network. Broadcast-style videoconferencing is an example of an application that employs IP multicast. Note that multicast addresses are only destination IP addresses, they cannot be a source address.

Figure 2: Multicast Connection



Broadcast

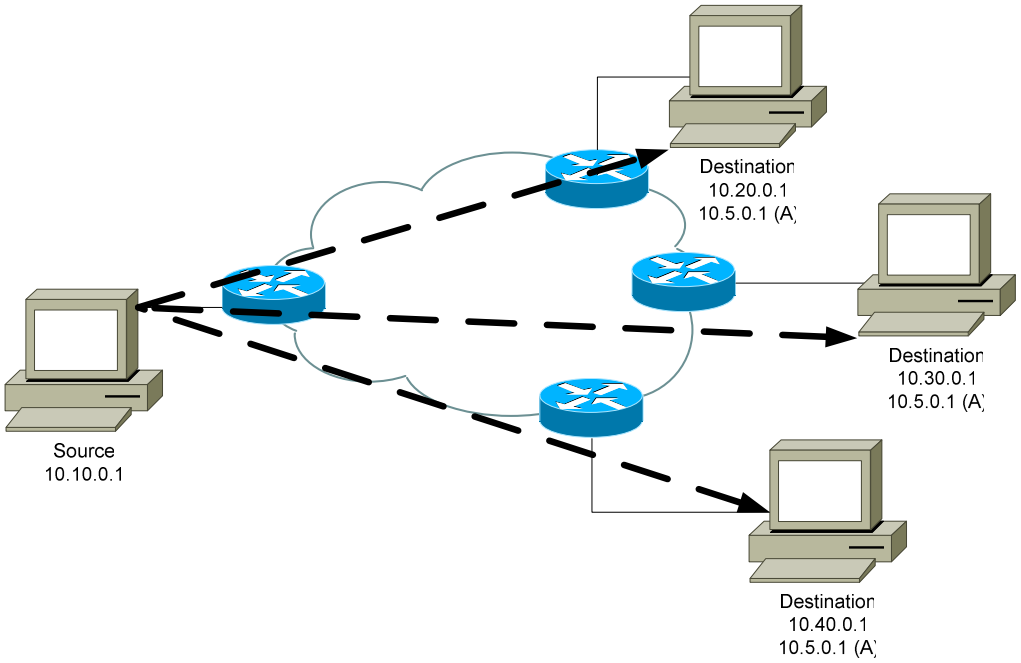
The third IPv4 connection type is broadcast. Broadcast addresses have been used to support discovery of services or servers. An example of a protocol using broadcast is ARP (Address Resolution Protocol). Packets sent to an IP broadcast address are delivered to all hosts on a particular physical data link network or IP subnet. Broadcast addresses are used only in the destination address field of an IP header because it is impossible for a unique packet to be sourced by more than one interface. IP broadcast addresses take the form of either the link local broadcast address (255.255.255.255) or the directed broadcast address (the network part plus all 1's in the host part of the intended subnet). It is obvious that broadcasts have to stay local on the subnet.

Anycast

The original definition of Anycast in RFC1546 is: “A host transmits a datagram to an Anycast address and the internetwork is responsible for providing best effort delivery of the datagram to at least one, and preferably only one, of the servers that accept datagrams for the Anycast address.”

In practice, Anycast is a point-to-point flow of packets between a single client and the nearest destination server identified by an Anycast address (Figure 3). The idea behind Anycast is that a client would like to send packets to a server offering a particular service or application, but it is not important which server is chosen. To accomplish this, a single Anycast address is assigned to one or more servers within a so-called Anycast group. A client sends packets to the server by placing the Anycast address in the packet header. Just as with a unicast flow, the client and server are unaware that Anycast is used. The network of routers will then attempt to deliver the packet to the closest server with the destination Anycast address, which will then reply.

Figure 3: Anycast Connection



Anycast was first introduced in IPv6, but was soon adopted in IPv4. One of the best known adopters is ISC, which anycasted the DNS F-root servers. In IPv6 a special address range is set aside for Anycast purposes; in IPv4 this has not happened. The Anycast addresses used in the IPv4 network must, therefore, be carefully chosen to make network management and troubleshooting easier.

Some additional characteristics of Anycast are:

- Multiple nodes are configured to accept traffic on the same IP address, but also have a unique address for management purposes.
- Anycast packets can be dropped like any other kind of traffic. Packets are not specifically marked or tagged.
- Preferably only one Anycast server receives a packet, but there is no absolute guarantee. It is possible that sequential packets from a client to an Anycast address are delivered to different servers. If servers are not synchronized incorrect data may be sent back.
- The server that receives a specific packet is solely determined by the unicast routing protocol used in the domain. There is no special Anycast routing table equivalent to a separate routing table for multicast traffic.
- Clients, servers, and routers require no special software/firmware. The only special configuration is needed on servers and routing infrastructure (discussed in the *Why Anycast* section). Therefore, it does not negatively interfere with existing networks or services. Anycast just leverages the existing infrastructure.

Anycast in IPv4

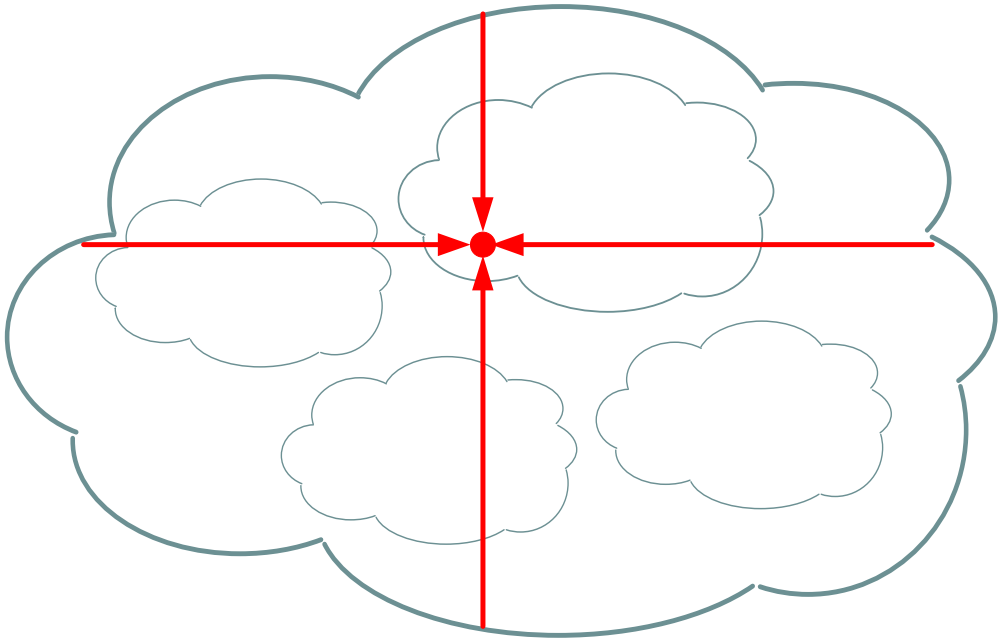
IPv4 routing is not designed to use Anycast optimally, but it can significantly leverage the infrastructure for some applications. Anycast usage is at this moment limited to mostly short (one-request-one-reply packet), connectionless exchanges of information. Domain Name Service (DNS) is a very good example. The exchange with DNS only takes two User Datagram Protocol (UDP) packages: request and answer. The next request can be sent to another server without any problem. No exchange of state is necessary. Some other good uses for Anycast are Network Time Protocol (NTP), Syslog (traps sent by devices to a management system), Rendezvous Point (RP) information in Protocol Independent Multicast Sparse Mode (PIM-SM) and export of flow information (example Cisco's Netflow).

Transmission Control Protocol (TCP) needs complicated state-change mechanisms to make Anycast work. TCP was not designed to handle these state-change mechanisms, and is therefore, less suitable for use of Anycast.

Why Anycast

Anycast addressing is a useful technique for providing reduced latency, simplified configuration redundancy, load balancing, and enhanced security to specific types of network services on the Internet. Anycast addressing is nothing more than assigning a common IP address to multiple instances of the same service, which are located at strategic points in the overall network topology (Figure 4). By utilizing the underlying routing infrastructure of the Internet, IP packets are forwarded to the nearest instance of an Anycast service.

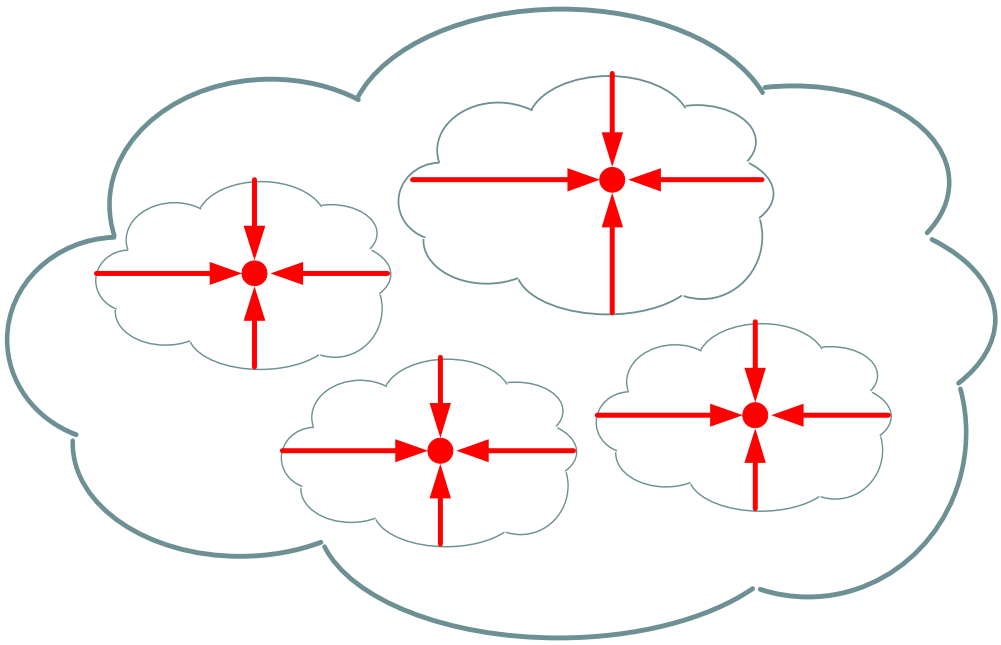
Figure 4: Traffic Flow with One Server for the Entire Network



Anycast can be an attractive way of distributing a service through the network for the following reasons:

- *Reduced use of router and link resources*: Standard IP routing will deliver packets over the shortest path to closest available server (Figure 5).
- *Simplified configuration*: A client only needs to be configured with a single Anycast address that identifies one of a group of possible servers that offers a particular service or application. It then becomes a simple task for the client to locate a server.
- *Network redundancy*: If a server in the Anycast group goes away, the network will deliver packets to the next closest Anycast server. The service will become more reliable.
- *Load balancing*: Anycast servers distributed over the network topology will have the effect of balancing the traffic load from many clients.
- *Reduced latency*: The average network latency between client and server across the entire Internet is reduced. For services whose transaction latencies are closely aligned to network latencies between client and server, Anycast may represent a useful optimization.
- *Security*: Denial-of-Service (DoS) or other malicious traffic is distributed with client query traffic. This can lead to the effects of an attack on a service being isolated to particular parts of the Internet or corporate network rather than causing impact networkwide. The pattern of attack traffic seen at different service nodes can also aid in the location of attack sources.

Figure 5: Traffic Flow Attracted by the Nearest Server



Common network services that can most easily take advantage of Anycast addressing include the following:

- *DNS* is probably the most suitable candidate for Anycast. The message exchange is just a query and response. A real-life example is the F-root server deployed by the Internet Systems Consortium (ISC). These servers have been successfully using Anycast addressing since November 2002.
  (Note: DNS messages can use TCP connections. Usually those connections are relatively short and will not hinder Anycast deployment for the DNS service.)
- *Multicast Rendezvous Points* (RPs) is also widely used with Anycast addressing. The multicast RP brings together sender and receiver(s) for a specific multicast group. When only one RP is used for a specific multicast group, a significant point of failure in a large network is created. As all traffic will be aggregated toward that single RP, this might then overload the network in that area. RFC 3446 has been released recently to update the original PIM-SM specifications. It now allows multiple RPs per group via Anycast addressing. Multicast Source Discovery Protocol (MSDP) has also recently been developed to let RPs exchange information about the group registrations they know about.
- *NTP* will also normally work with Anycast. A small risk with NTP is that it generally requires at least two packets from both server and client to get a proper synchronization. If the server fails after the first packet, it will take an extra packet to synchronize with the next available NTP server. The Simple Network Time Protocol (SNTP) does not have this problem.
- *Syslog trap collectors* can use Anycast addresses because generally no response is expected from this type of server. If the server is also used as a two way SNMP manager, a globally unique address could be used for SNMP management.
- *Network flow export*, such as for Cisco's Netflow, can use an Anycast address for the flow-record collectors.
- *IPv6-to-IPv4 relay routers* as defined in IETF RFC 3068 have their own netblock, which acts as an Anycast service for IPv6 networks to talk to IPv4 networks. These v6-to-v4 relay routers use Anycast to help ease network management of IPv4 to IPv6 protocol transition.
- *Sinkhole networks* are used to route bogus netblocks of a network. The problem with sinkhole networks is that often routers will transport bogus traffic across the network to reach the sinkhole network. With Anycast, these sinkhole networks can be distributed across the domain and minimize the traffic across the backbone of the network. Anycast can be used to monitor bogus address space.

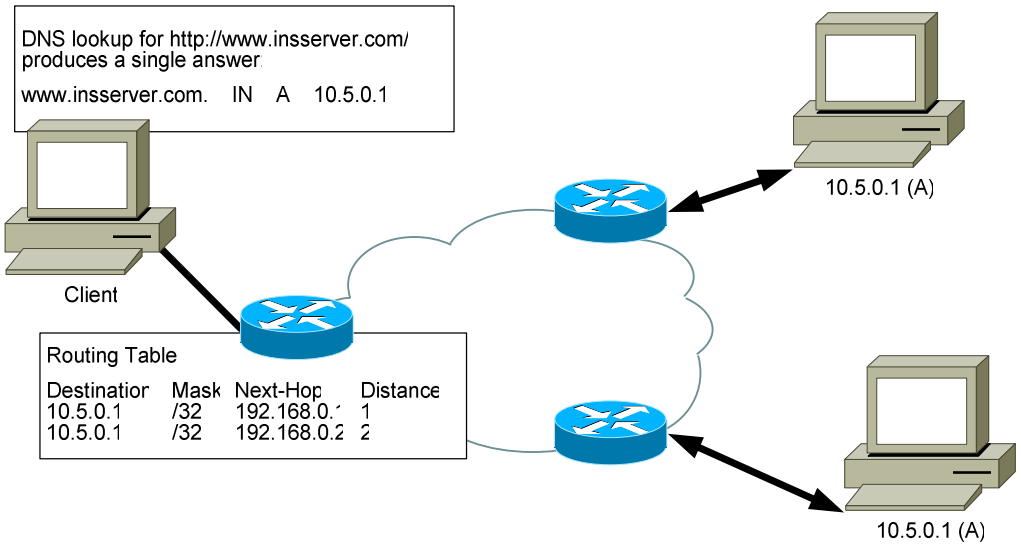
DNS and Multicast RP are the primary network services that can quickly and easily benefit from Anycast.

Routing Implications

The following sequence describes the chain of actions that occur for a client to look for and connect to an Anycast server (Figure 6).

- Client does a DNS lookup for the IP address belonging to www.insserver.com.
- DNS server (maybe by Anycast) returns in a single answer the Anycast address of the server. So far nothing has changed compared to traditional Unicast.
- First hop router for the client does a route lookup and finds two (or more) routes in its routing table, each with a different distance. (How these addresses come into the routing table will be discussed in the next section.) The route with the shortest distance will be selected to route the traffic.
- Traffic gets routed to the server. The server will respond to the traffic and sends traffic back.
- Connection is established.

Figure 6: Unicast Destinations in the Routing Table



When multipath load balancing is supported, the router may have more than one route entry in the routing table. If load balancing is not possible, the route selection method in the routing protocol used will automatically select the nearest destination.

Implementing Anycast

The first step to implementing Anycast should always be careful planning about what's to be achieved. After your objectives are clearly defined, the following steps will be necessary:

- *Address selection*: Current practice is to assign small subnet(s) of Anycast addresses from unicast IP space. Usually a class C or /24 range is sufficient. The subnet chosen cannot be attached to any existing interface in the network. The choice depends on the current address ranges you have in use.
- *Location of the servers*: The network will perform best if servers are widely distributed, with higher density in and surrounding high-demand areas. Lower initial cost sometimes leads implementers to compromise by deploying more servers in existing locations, which is less efficient.
- *Server configuration*: Servers need to get a second address used for Anycast (detailed in the next section).
- *Service configuration*: Configure the service on the server to respond to the Anycast address. For Microsoft applications this might be difficult. For example Microsoft Internet Authentication Server (IAS) cannot set the address from which it responds, so it will keep using the primary IP address. A RADIUS client will reject an answer coming from a different source address.
- *Network configuration*: Configure the proper redistribution and filtering on the router connecting the Anycast server (detailed in the section *Network Configuration*). When necessary multiple path load-balancing can be enabled. But this is not a requirement for Anycast.
- *Anycast testing*: The Anycast service is ready to be tested and used when all is working properly. Only monitoring of the Anycast service is now needed, for instance, periodically checking the reachability of the Anycast servers from different points in the network. Check reachability of the Anycast address and of the network management addresses of the servers.

Server Configuration

General preparations for any type of configuration include configuring the Anycast on the server. This can usually be done by adding a second address to the server. Always use a unique unicast second address management purposes. Typically, Anycast addresses are configured on the server as an additional loopback interface. But a secondary address applied to the same interface is also possible. Remember that when an Anycast address is added on a server, the filters (when present) on the router interface need to be adapted.

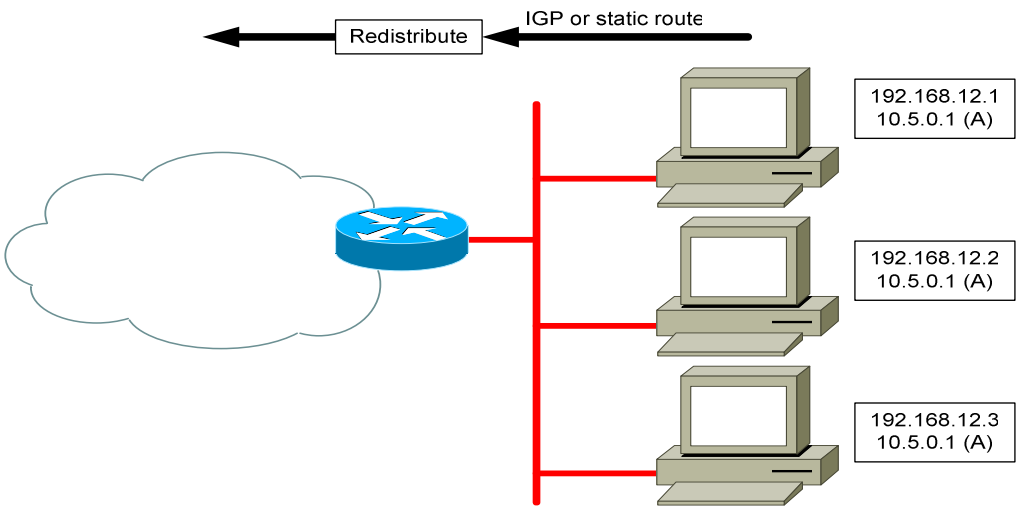
Server configuration example for LINUX:

```
# ifconfig lo:1 10.5.0.1 netmask 255.255.255.255 up
# ifconfig lo:1
lo:1      Link encap:Local Loopback
          inet addr:10.5.0.1  Mask:255.255.255.255
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
```

Network Configuration

Anycast can be implemented on the router in two ways: either use a static route on the first hop router or enable a routing protocol on the server. With both choices the router needs to redistribute the routing information into the Internal Gateway Protocol (IGP) in use (Figure 7).

Figure 7: Redistribution in Global Routing Table



Simple implementation: configure static routes on first-hop routers (host routes) pointing toward the primary address of the server. For a Cisco IOS router, this might look like[1]:

```
ip route 10.5.0.1 255.255.255.255 192.168.12.2 <distance>
```

Make sure that the static routes are propagated through the IGP by redistribution. For OSPF on a redistribute static must be configured:

```
Router ospf
Redistribute static metric-type <1/2> subnets
```

Filtering must be applied if not all static routes configured on the router need to be redistributed.

Pros and cons of using static routing at the first hop router are:

- *Pro*: Simplest to configure
- *Pro*: No risks with running a routing protocol on server. Implementations may not be as stable as on a router's routing protocol. A server is not basically designed to do routing; a router is.
- *Con*: There is no response to server failure. If the server fails, the static route needs to be removed manually. The address will be advertised to the rest of the network whether the server is up or not.

[1] All configuration examples are for Cisco IOS routers

Enhanced Implementation

For an enhanced implementation, run a server-based routing daemon on the Anycast server, for instance ripd, ospfd, GateD, Zebra/Quagga in a UNIX environment, or RIP or OSPF on a Windows server. The protocol also needs to be enabled on the router. Redistribution can take place into the IGP. Be careful to accept only the wanted Anycast address(es) and nothing else otherwise false routes may enter the network. Configuration of routers and servers is obviously specific to the IGP used and to the server itself. If the Anycast address on the server is a loopback also redistribution needs to be configured on the server.

Pros and cons of enabling a routing protocol on the server are:

- *Pro*: Server itself is route originator, so when the server is down the route is automatically withdrawn from the routing table after the time taken for the routing processes to converge.
- *Con*: The server being up does not also imply that the service is running, only that on level three the server is reachable. Some management software should guard that the service also is running.
- *Con*: There is no mechanism for withdrawing routes automatically when the service becomes unusable.

The choice for either implementation depends on the specific requirements for Anycast and has to be for reinvestigated every deployment.

Security issues

There are at least two security threats when using Anycast. First, the possibility that a malicious server could divert traffic from a legitimate server to itself (this can affect ‘normal’ servers as well). In an implementation with static routes this will not be a problem because there is no static route configured to any other than the intended server. Securing the routers against unwanted reconfigurations is not in the scope of this document, but could be a real threat. In a dynamically routing solution, special measures must be taken to protect from unwanted servers advertising routing information. This can be done either with proper filtering or authenticating the route updates received.

Second, an eavesdropping server replying to queries with inaccurate information is a security threat. This is a more difficult problem, requiring extensive actions beyond the scope of this paper.

Conclusions

IPv4 Anycast addressing can be a good solution to providing localization for services and servers in order to obtain robustness, redundancy, and resiliency. It is a useful technique to make a distinct set of services available to the network. For these services some scaling, redundancy, and/or security problems are improved significantly. When considering Anycast, carefully choose the address range to be used.

- Anycast is best used for simple request response type protocols based on UDP. It is not very well adapted to TCP-type connections.
- Correctly configuring the network is the trickiest aspect of Anycast. Carefully choose between static routing and enabling a routing protocol on the server.

- The network will perform best if servers are widely distributed, with more density in and surrounding high demand areas.
- Lower initial cost sometimes leads implementers to compromise by deploying more servers in existing locations, which is less efficient.
- Finally, be aware of maliciously configured servers in the network diverting traffic or eavesdropping.

References

http://www.net.cmu.edu/pres/anycast

Anycast addressing on the internet, jtk, , Jan 2004.

IETF RFC 1546 – Host Anycast Services, Partridge, C. et al., IETF, Nov 1993.

IETF RFC 3068 – An Anycast Prefix for 6to4 Relay Routers, IETF, Jun 2001

IETF RFC 3330 – Special-Use IPv4 Addresses, IETF, Sep 2000

Hierarchical Anycast for Global Service Distribution, Joe Abley, ISC, 2003.

Deploying IP Anycast, Kevin Miller, Carnegie Mellon Network Group, Oct 2003.

About INS

INS (International Network Services Inc.) provides IT infrastructure consulting services, software, and solutions to help companies build, secure, and manage business-critical networks. Our end-to-end consulting solutions address customers' needs in IT Strategy and Planning, IT Infrastructure, Operating Systems and Directory Services, Storage Systems and Services, Security, and Network and Systems Management, helping them optimize their businesses to better face competitive challenges and meet future demands. The Diamond IP family of software from INS provides flexible and scalable solutions for today's complex IP networks. We are one of the world's largest independent IT infrastructure consulting solutions providers with a track record of thousands of successful engagements. INS is headquartered in Santa Clara, Calif. and has offices across the U.S. and Europe. For additional information, please contact INS at 1-888-767-2788 in the U.S., 44 (0) 1628 503000 in Europe, or 1-408-330-2700 worldwide, or visit www.ins.com