

IP Addressing Space Design Issues for Internet Data Centers

Lynne Marchi, Intel Online Services, Intel Corp.
Sridhar Mahankali, Intel Online Services, Intel Corp.
Jeff Sedayao, Intel Online Services, Intel Corp.

Index words: Internet protocol, address space, data center

ABSTRACT

The implementation of Public Internet Protocol (IP) address space is a key factor in the size and growth of Internet data centers. IP addressing space decisions affect how many servers can be hosted at a data center, and they influence the kind of network connectivity technology that will be used and even how web sites are implemented. This paper describes IP addressing issues for Internet data centers. First, we provide an overview of Internet addressing and routing: we discuss IP networks, autonomous systems, and high-level Internet network routing. Key Internet constraints are described, particularly the finite amount of IP address space and autonomous systems and the current addressing and routing policies that result from those constraints. We then go over key IP address design decisions. The Internet data center builder needs to decide what address space to use, the size of that address space, the autonomous system number to use, and the address allocation policy to use with customers. These choices are constrained by the difficulty of obtaining space, the required speed of implementation, Internet Service Provider (ISP) routing policies, ISP connectivity decisions, and security requirements. Next, we describe how these design choices affect technology choice and implementation with the data center, by using virtual web site design and Network Address Translation (NAT) as examples. We then provide examples of how address space constraints affect the design of Intel® Online Services (IOS) data center address spaces and other technology choices. The last section discusses some trends and future technologies that may alleviate IP address constraints.

INTRODUCTION

Issues with Internet Protocol (IP) address space are critical, yet often overlooked, factors in building and maintaining Internet-accessible data centers and web server farms, such as those hosted by companies like Intel Online Services. A shortage of address space can limit the growth and expansion of data centers. Moreover, dealing with the scarcity of IP address space and with situations where the data center and customers have to communicate while sharing the same private address space drives technology decisions and complicates the debugging of server and network problems. This paper describes how IP address space concerns can impact the design, implementation, and operation of Internet data centers. First, we present an overview of how IP addressing and routing works on the Internet. We then discuss key address space design considerations. The next section describes the effects of addressing choices on data center design and implementation, and it is followed by a section in which we show examples of address space design decisions at Intel Online Services. We end with a discussion of future technology development trends regarding IP address space.

A BRIEF INTRODUCTION TO INTERNET ADDRESSING AND ROUTING

Internet protocol addressing and routing must first be understood before any discussion of IP address space design issues will be useful. This section goes over the original IP address scheme, its limitations, and the current methods used to deal with the finite number of IP addresses. This information is crucial to an understanding of the choices and constraints for IP addresses in a data center.

Class	Range of Network Numbers	Default Network Mask	Network vs. Host Portion	Number of Hosts
A	1.0.0.0 to 126.0.0.0	255.0.0.0	network.host.host.host	16,777,214
B	128.1.0.0 to 191.254.0.0	255.255.0.0	network.network.host.host	65,534
C	192.0.1.0 to 223.255.254.0	255.255.255.0	network.network.network.host	254

Table 1: Original IP version 4 address class

Table 1 shows the different classes of IP addresses. Note that two other classes of address space, class D and E, were not included in the above table. Class D addresses start at 224.0.0.0 and are used for multicast. Class E addresses start at 240.0.0.0 and are used for experimental purposes.

Original IP 4 Addressing Scheme

In order for two hosts to communicate over the Internet, there needs to be a way to uniquely identify hosts. In 1981, the Internet Engineering Task Force (IETF) created the Internet Protocol, IP version 4 (IPv4) [1], which defines the current method of uniquely identifying hosts. IPv4 addressing uses a 32-bit binary address. The IETF also incorporated support for decimal representation of addresses to make the addresses human-readable. In decimal form, an IP address consists of 4 octets (sets of 8 bits), separated by dots. Each octet can be a number ranging from 0 to 255. Examples of valid (decimal) IP addresses are 10.245.171.1 and 172.16.50.224.

IP addresses are partitioned into a network portion followed by a host portion. Hosts belong to a network, and that network is defined by the network portion of the IP address. The original design called for classes of address space that divided the IP space into large, medium, and small networks that could be assigned to organizations (businesses, universities, government agencies, etc). Included in the design was the notion of a network mask that defines what part of an IP address is the network portion (as opposed to the host portion of the address). In binary, the network portion of the address is a series of ones that is then followed by a series of zeroes representing the host portion of the address. In decimal, the network portion of the mask is equal to 255 for each octet.

Autonomous Systems

Another requirement for Internet communication is that each host needs to know how to reach all of the other hosts. To facilitate this, organizations advertise the path to their network to other networks. Devices called routers learn about networks in this way and forward packets appropriately. Routers exchange network and routing information through what is called a routing protocol. A *routing table*, which is a list of networks and the next hop (often another router) to forward packets for those

networks, is stored in the router's memory. The router will select the best path (next-hop router) to put into its table when multiple paths exist to the same network.

In the early days of the Internet, all connected routers shared their routing tables. As use of the Internet started to grow, more routers and networks were added, and the amount of overhead required to store the routing table and manage changes to the routing table also increased. In addition, as more companies began manufacturing different routers that ran their own implementation of the routing software, compatibility issues between different vendors arose. For these and a number of other reasons, it was decided to break the Internet into smaller routing domains, called Autonomous Systems (AS).

An autonomous system (AS) [2] is a set of routers and networks that are managed by one or more administrative entities (e.g., company, university, Internet Service Provider, etc.). Each AS is assigned a unique number so that communication between different autonomous systems can occur. Routers inside the AS run an Interior Gateway Protocol (IGP) such as RIP [3] and OSPF [4]. To communicate externally, one or more border routers are chosen. Border routers use an Exterior Gateway Protocol (EGP) to exchange routing information with routers in different autonomous systems. Today, the *Border Gateway Protocol Version 4* [5] (BGP4) is generally used for this purpose.

Each AS has a number associated with it. BGP4 uses 16 bits for AS numbers, so that AS numbers range from 0 to 64535. The upper 1024 are reserved as private AS numbers, usable only within an AS and not directly reachable from the Internet. This leaves AS numbers 1 to 64511 as valid, Internet-usable AS numbers.

Issues With IP Addressing

Since IPv4 was finalized, use of the Internet has grown exponentially, causing major addressing issues. In the early days of the Internet, organizations were able to obtain large blocks of IP space without proof that it was needed or even going to be used, and as a result, IP address space was being rapidly depleted. Another side effect of address space allocation policies was that the routing tables for Internet routers were once again becoming huge [6]. Remember, routers store a list of networks and next hop information in memory. When

routing tables are large, they take up more memory and more CPU processing time is required to search them.

Finally, the class of address space as defined in Table 1 did not always meet, and sometimes exceeded, the needs of the organization receiving it. For example, a small business that expected to grow to no larger than 300 hosts would require two Class C networks (508 addresses). This wasted 208 addresses (two 256 host networks minus four addresses that are network overhead and minus the 300 hosts)!

Address Allocation Authority

To slow the depletion of IP space, the Internet Assigned Numbers Authority (IANA) [7] was established to oversee allocation of the remaining IPv4 addresses. IANA further delegated this authority to the regional registries:

- American Registry for Internet Numbers (ARIN)
- Asia-Pacific Network Information Center (APNIC)
- Réseau IP Européens (RIPE NCC)

Today, it is much harder to obtain IP address space as the requesting body must provide a detailed plan that shows that the requested space is justified and how it will be used.

Subnetting Changes

Several new methods of addressing were also created so that usage of IP space was more efficient. The first of these methods is called *Variable-Length Subnet Masking* (VLSM) [8]. Subnetting had long been a way to better utilize address space [9]. Subnets divide a single network into smaller pieces. This is done by taking bits from the host portion of the address to use in the creation of a “sub” network. For example, take the class B network 147.208.0.0. The default network mask is 255.255.0.0, and the last two octets contain the host portion of the address. To use this address space more efficiently, we could take all eight bits of the third octet for the subnet.

One drawback of subnetting is that once the subnet mask has been chosen, the number of hosts on each subnet is fixed. This makes it hard for network administrators to assign IP space based on the actual number of hosts needed. For example, assume that a company has been assigned 147.208.0.0 and has decided to subnet this by using eight bits from the host portion of the address. Assume that the address allocation policy is to assign one subnet per department in an organization. This means that 254 addresses are assigned to each department. Now, if one department only has 20 servers, then 234 addresses are wasted.

Using variable-length subnet masks (VLSM) improves on subnet masking. VLSM is similar to traditional fixed-length subnet masking in that it also allows a network to be subdivided into smaller pieces. The major difference

between the two is that VLSM allows different subnets to have subnet masks of different lengths. For the example above, a department with 20 servers can be allocated a subnet mask of 27 bits. This allows the subnet to have up to 30 usable hosts on it.

Class	Private Address Space
A	10.0.0.0 to 10.255.255.255
B	172.16.0.0 to 172.31.255.255
C	192.168.0.0 to 192.168.255.255

Table 2: Private address space ranges

Private IP Space

In 1996, IANA set aside three blocks of the global IP space to be used by organizations solely for the purpose of internal communication [10]. This address space, called private IP space, meant that a company could assign private addresses to hosts inside the company that did not require direct access to the Internet. Any organization could use private space without fear of colliding with another organization’s address space. This allowed companies to conserve on the public IP space they had already acquired by assigning it to only those hosts that needed to communicate directly with the Internet. Table 2 shows which networks can be used for private addressing.

Classless Internet Domain Routing

So far, the discussion on IP address allocation has used the model shown in Table 1. This model is often referred to as a “classful” model because it relies on using the definitions of class A, B, and C networks. *Classless Inter-Domain Routing* (CIDR) [11, 12] eliminates classful addressing in the same way that VLSM eliminated fixed-length subnet masks. CIDR uses a prefix to indicate the number of bits used for the network portion of the address, while the remaining bits are used for the host address. For example, 147.208.61.8/20 is a CIDR address in which the first 20 bits contain the network portion of the address, leaving 12 bits for the host portion. The network mask for a /20 prefix is 255.255.240.0 and is equivalent to 16 traditional class C networks!

Another advantage of CIDR is it allows routes to be aggregated. This means many networks can be summarized into a single route. For example, 147.208.0.0/19, 147.208.32.0/19, 147.208.64.0/19, and 147.208.192.0/19 can be summarized as 147.208.0.0/17. Once CIDR was implemented, the growth in the size of Internet routing tables was significantly reduced.

ADDRESS SPACE DESIGN ISSUES

When implementing an Internet data center, there are a number of key decisions that need to be made about IP address space. In this section, we discuss four key design

points: what address space to use, the size of address space to advertise for a data center, what autonomous system number to use, and the IP address allocation policy. For each of these design issues, we talk about the different choices available and the tradeoffs involved with each choice.

What IP Address Space to Use

The first critical choice that data center implementers have to make is what IP address spaces to use. There are several choices here:

1. Private IP address space
2. Currently owned and used space
3. Space from the data center's ISPs
4. New space obtained directly from Internet registries
5. Customer space

These choices are not mutually exclusive within a data center, and we will go over the tradeoffs involved with each choice.

Private IP address space has the primary advantage of being plentiful and immediately available. It has the primary disadvantage of not being immediately usable on the Internet without some form of Network Address Translation (to be discussed later) or some kind of proxying technique. This disadvantage is at times not a problem. For applications and services that do not require direct access to the Internet, this is not a concern. Also, for hosts such as database or application servers that do not directly talk to other systems on the Internet, this has some security advantages, as these systems are not vulnerable to direct attack from the Internet. (Do not think that they are invulnerable because of this, however).

If data center implementers already have their own IP address space, another possibility is to use that space. This can be advantageous, as an organization may have plenty of address space to utilize immediately. The prime disadvantages can be with routing. For example, some ISPs do not accept network prefixes longer than a /16 for parts of traditional class B networks. So if you wanted to use part of a /19 part of a traditional class B for a data center, that data center would not be accessible from all ISPs.

Another option is to use space from an ISP. ISPs have address space that they will provide to customers. While this option has the advantage that there is address space to use immediately, this option has a number of powerful disadvantages. The first disadvantage is that using an ISP's address space will typically limit you to using one ISP. That address space is bound to that ISP, and you typically will not be able to have traffic routed through another ISP. Another disadvantage is that should you choose to discontinue your service with that ISP, you would have to give back all of the space you received,

forcing you to renumber all of the hosts directly accessible from the Internet.

Another option is to obtain new address space directly from the Internet registries that distribute space. This option has a number of advantages. A data center using space obtained from the Internet registries can change ISPs without having to worry about renumbering hosts. The registries usually allocate addresses in /19 blocks, making that address space immediately routable. The disadvantage of getting space from registries is that the process takes time, on the order of weeks, and longer if you need to first join the registry. The process also involves rigorous justification of address allocation and why currently owned addresses will not suffice. In addition, once space is allocated, it cannot be used all at once. To use more of an address allocation, another justification process is required, often requiring verification that previously assigned addresses have been used.

A final choice is using the data center's customers' address space. When possible, this is good, but it is often not possible, as Internet data center customers usually expect that you will use your own address space. Even if a customer is willing to do this, it may take some time to make changes to the data center's ISP's address filtering to make it possible to use that space. Also, customer address space is also vulnerable to the same problems as address space you already own. The customer may use a piece of existing address space, such as part of a class B, that some ISPs may refuse to accept as a route.

Size of Address Space to Advertise from the Data Center

A decision closely related to what address space to use is what size of address space to use and how to advertise it on the Internet. Clearly, you can only advertise the space that you have: that puts an upper limit on the address space advertised for the data center, and thus a lower limit on the prefix length of the network advertised. Many ISPs will not accept route advertisements for networks with prefix lengths longer than /19, which puts an effective upper limit on the prefix length of what you advertise and a lower limit on the size of the address space.

There are a number of factors that affect the size of the address space advertised. First, it depends on how many hosts within the data center need public addresses. You want to advertise enough address space to cover the hosts that need public addresses, both immediately and in the near future. To make routing more manageable and to help reduce the growth in the size of the Internet routing table, it is better to advertise fewer routes. Instead of advertising each network in a data center, if you advertise a single aggregation of those routes, there are fewer routes to manage. As mentioned above, some ISPs will not accept parts of a traditional class B network. One way to

deal with such ISPs is to connect data centers to ISPs who will accept parts of a class B. Each data center can advertise a prefix that is short enough to be accepted, while at least one data center can advertise the whole /16 class B network. This way, those ISPs who only accept a whole class B will see a route for the whole class B and send it to the data center's ISP. Once it is in the data center's ISP, the ISP will route traffic to the most appropriate data center because each data center is also advertising a route for its section of the class B.

Another factor in deciding what size address space to advertise is the backbone infrastructure connecting data centers. If data centers are connected by a backbone network that has enough capacity to route significant amounts of public Internet traffic coming into one data center that is bound for another (the worst case being that the backbone will handle all of a data center's traffic), then it is feasible to advertise a single route that aggregates all of the data center's networks into one. If the backbone connecting the networks doesn't have the capacity, advertising a single aggregated route can result in performance bottlenecks when end users accessing one data center access that data center through another.

Autonomous Systems Number

The issues and choices regarding Autonomous Systems (AS) numbers are very similar to those regarding IP address space. A data center's address space can be advertised from the following:

- Private AS
- Currently owned and used AS
- The data center's ISP's AS
- A new AS obtained directly from Internet registries
- Customer AS

Advertising a route from a private AS number is fast and immediately available, but those private AS numbers are only usable within an organization's public AS. Like private IP addresses, private AS numbers are not usable over the Internet. Advertising from an existing AS number that is already owned by a data center's administrators is also easy and quick. One consideration to keep in mind when using an existing AS is the routing policy implemented by ISPs and other organizations. Routing policies are often implemented by AS numbers, and each of the data center networks advertised from that AS will be affected by such a policy. This can become a great disadvantage when data centers are spread across geographies. Internet conditions can vary greatly: the routing policy made on one continent may be (and usually is) totally inappropriate on another.

If a data center's routes are advertised from an ISP's main AS number, the data center is locked into using only that ISP and cannot have connections into other ISPs (although it should be mentioned that some ISPs provide

a special AS for multihomed customers). Getting a completely new AS number from an Internet registry has the advantage that a data center can change ISPs with much less work, usually just changing entries in routing registries. Multihoming to multiple ISPs is now possible, and end-user access to the data center can be improved by routing traffic based on using the full Internet routing table. The down side to getting a new AS is that it takes time: you usually have to join an Internet registry and apply for an AS number. Also, AS numbers are limited in quantity, as mentioned above, with only 64511 AS numbers available for use directly on the Internet.

Finally, a data center can advertise address space/route from a customer's AS number. This has the advantage of allowing that address space to be served by multiple ISPs. It has a number of constraints and disadvantages. Only the address space/routes that the customer owns can be advertised as using that AS. As with using a previously owned AS, this option has the disadvantage of being affected by any other policy that organizations and ISPs may implement based on the customer's AS numbers. The data center effectively becomes an ISP, and the data center's ISPs often must change AS and network filtering policies to allow that route to be advertised. In addition, routing registry information concerning that network will also have to be changed.

IP Address Allocation Policy

Given that IP addresses are limited in quantity and their use has serious constraints, the allocation policy for IP addresses to data center customers is a serious concern. There are a number of constraints and tradeoffs. The first choice that needs to be made is whether to allocate a separate address space to each customer. From a customer, security, administrative standpoint, it is better to give each customer separate address spaces. Firewall policies are easier to implement on a subnet basis, and any special traffic policies, such as giving certain customers a different path or giving them priority over others, are much easier to implement if customers are on different subnets. Customers may be competitors, and the thought of a competitor on the same subnet may be unpalatable to a data center customer. The cost of separate subnets per customer is loss of usable address space. Each subnet has a subnet number, and typically that is not used as a host name to avoid confusion. Also, each subnet has a broadcast address that cannot be used for hosts. As a result, there are two addresses consumed as overhead for each subnet. The more subnets, the more addresses that are lost from subnetting overhead. Figure 1 shows the fraction of a subnet that is lost to varying degrees of subnet overhead if that space is divided into subnets with prefixes of the specified length. Half of all addresses in a /30 are consumed by subnet overhead even

with the lowest possible subnet overhead.

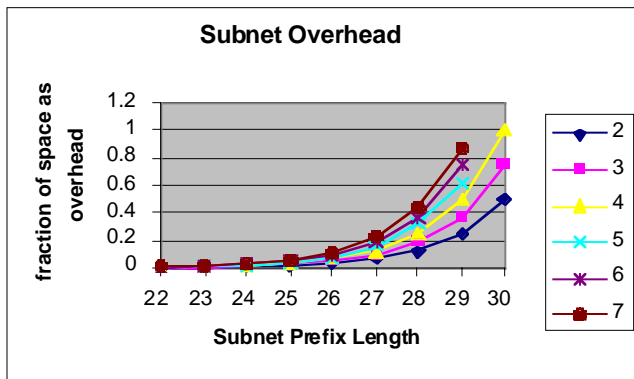


Figure 1: Fraction of overhead per subnet, depending on subnet length and overhead per subnet

The amount of overhead consumed per customer subnet affects usable address space availability. There are router redundancy techniques such as *Hot Standby Router Protocol* (HSRP) [13] that allow more than one router to handle traffic for a virtual interface. The overhead cost of HSRP is one virtual address and one address per router. Thus for a /29 segment with two routers using HSRP on the subnet interface, 62.5% of the available address space is consumed with overhead, leaving only three usable IP addresses. Figure 1 also shows how usable address space disappears with the increase in per subnet overhead.

From a customer and administration standpoint, it is also advantageous to have as much address space as possible. This makes it easy for a data center customer to expand operations by adding servers. Moving servers to a completely different, larger subnet in order to expand forces a customer to reconfigure all the hosts, typically involving significant downtime. As mentioned above, firewall and other access policies are often configured by subnet, and having as large a subnet as possible dedicated to a customer allows additions and changes to be made to servers without having to change those firewall and access policies.

Of course, since the supply of IP addresses is limited, customers cannot have all the space that might be convenient for them. Data center administrators must consider what happens when address space becomes nearly exhausted. In that case, they need to consider meeting Internet registry requirements to obtain new space. Typical registry requirements for new public IP space are as follows:

- 25% of the new space must be utilized immediately.
- 50% of the space must be used within one year.
- To get more space, the address space must be 80% utilized.

If these rules are not followed, the data center will be hard pressed to get more space if necessary. Again, the cost of giving smaller allocations of address space is that there

will be more subnets and more addresses consumed with subnetting overhead.

The final consideration that a data center needs to evaluate is economic. An IP address has economic value, and if a customer is willing to pay for space that is unused, the value needs to be weighed against another customer using that space and also generating income.

EFFECTS OF ADDRESS SPACE CHOICES ON DATA CENTER IMPLEMENTATION

The scarcity of IP addresses drives many implementation and technology choices. In this section, we discuss some of these choices, particularly Network Address Translation, the complications of implementing Network Address Translation, and web server implementations.

Network Address Translation

A very useful service that an Internet data center can offer is connectivity between a customer's internal network and their servers hosted at the data center. One problem that such a service can encounter is conflict between private address spaces. The resolution of private IP address conflicts can affect address space design choices made at the customer end as well as those made for data center internal networks. Another major design factor is the preference to hide customers' internal networks from being seen in the data center network.

A popular technology used for resolving IP address conflicts is *Network Address Translation* (NAT) [14]. NAT helps translate IP addresses to a non-conflicting IP space and can be used to resolve IP conflicts that occur between a customer network and data center internal networks, as well as those that occur between two different customers' networks. There are two different modes of NAT:

- many-to-one or many-to-few NAT
- static one-to-one NAT

Many-to-one or many-to-few NAT entails hiding a set of networks or IP addresses behind a single IP address or a small pool of IP addresses respectively. A key characteristic of this form of NAT is that in addition to the IP being translated to a non-conflicting IP space, the port numbers are also translated to dynamically assigned port numbers to enable differentiation among the set of networks or IP addresses being hidden.

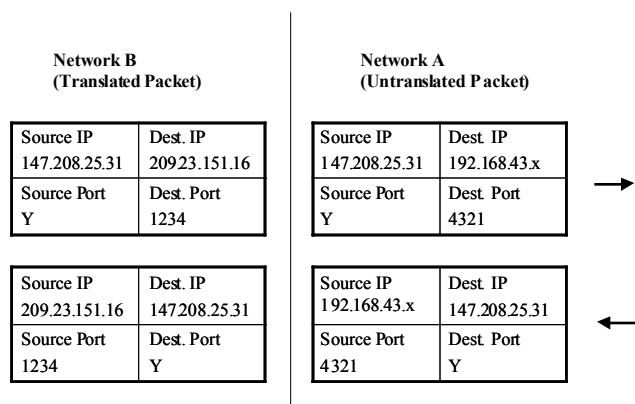


Figure 2: Many-to-one NAT packet flow

Figure 2 shows a sample flow of how a packet changes as it traverses the many-to-one NAT boundary between network A and B. Any traffic traversing from network A (designated by 192.168.43.0/24) to network B has its source IP address translated to a single IP address (209.23.151.16). To differentiate the multiple hosts on network A trying to communicate with hosts on network B, the source port 4321 is also translated to a dynamically assigned port 1234 as the NAT boundary is crossed. The device performing NAT maintains a dynamic table of these IP addresses and port translations to ensure appropriate communication between network A and network B.

A consequence of using many-to-one or many-to-few NAT is that network communication cannot be initiated bidirectionally. Consider the example in Figure 2. Network communication can be initiated from hosts on network A to those on network B. Hosts on network B cannot initiate connections to hosts on network A because network A is hidden behind a single IP address 209.23.151.16, and hosts on network B cannot uniquely address a host on network A for communication. This could potentially be considered a security feature as well.

Static one-to-one NAT entails translating an IP address uniquely to another IP address. In addition, static NAT does not involve any port translation. Further, there is no restriction regarding which direction network communication can be initiated in because the device performing NAT can uniquely translate back to a specific host on the network with the conflicting IP space.

Based on network communication requirements, many-to-one or static NAT or both can be used. Later on, we discuss how Intel Online Services uses NAT for IP conflict resolution under various remote access scenarios. Extensive use of NAT also drives the need for choosing a device that allows enabling of flexible NAT configurations to fit the diverse requirements of multiple data center customers.

NAT may need to be used more than once as the network traffic traverses between the customer networks and the data center networks and back. An example of such a situation is when one customer's internal IP space conflicts with another customer's internal IP space. In such a situation, NAT would need to be performed at one of the customer ends to resolve IP conflicts with the other customer's IP space. NAT would need to be used a second time at the data center end to hide the customer network from data center internal networks.

To further complicate matters, there could be situations where a customer needs to run applications such as DCOM [16] that do not work across NAT. In such situations, alternate solutions like readdressing customer end systems to a public IP space and allowing that IP space to be visible within the data center may need to be explored.

Debugging Complications from NAT

While NAT can be a very useful tool for resolving IP address conflicts and enabling end-to-end connectivity to customer-end networks, the use of NAT can lead to complex troubleshooting scenarios [17]. In situations where NAT is performed more than once as a packet transits from its source to its destination, the IP address of an end system will change as many times as a NAT boundary is traversed. Debugging access issues in such cases requires intimate knowledge of the end-to-end network path and also a clear comprehension of which IP address is associated with an end system on a given section of that network path.

A packet sniffer is critical when troubleshooting potential connectivity issues across a device performing NAT. Visibility inside the packets captured on both sides of the NAT boundary helps establish whether the IP address translation is occurring as desired. Looking at the payload in a packet can also help identify if an application will work across a NAT boundary. Usually, applications that contain IP addresses or application port information in the packet payload will not work across NAT as the NAT process modifies the IP address only in the IP header and does not modify anything in the packet payload. Without visibility into a packet, this kind of troubleshooting could be long and tedious.

Use of NAT can also lead to configuration complications on the end systems since the translated address is valid on one side of a NAT boundary and the actual IP address may be valid on the other side of that NAT boundary. For example, when printing to a printer whose IP address has been statically translated (one-to-one), let's say from the actual printer IP of 172.16.20.5 to an IP address of 10.81.249.23, the print job will need to be initially sent to the translated IP address i.e., 10.81.249.23. Consider another case where the printer IP address is translated from its actual IP address of 172.16.23.15 to 192.168.23.26 at the customer end and is further

translated to 10.81.249.35 at the data center. In this case, the system at the data center will need to send a print job to 10.81.249.35 in order to print to the actual printer at 172.16.23.15.

Furthermore, in the case of many-to-one or many-to-few NAT, the network traffic can only be initiated in one direction as discussed earlier. This should be kept in mind when troubleshooting connectivity issues across such address translation boundaries.

Virtual Web Server Implementation

Data center customers occasionally implement what are called *virtual servers*. This means that multiple web sites are implemented on the same set of servers. One way to implement this is to use a different IP address for each web site, with each web server having multiple IP addresses. Each IP address corresponds to a different web site. For example, let us say a customer has three web servers (1, 2, and 3) and two different virtual servers, www.site1.com and www.site2.com. Each web server would have two virtual IP addresses, one for www.site1.com and another for www.site2.com. www.site1.com would map to an IP address which is load balanced between the three site 1 IP addresses. www.site2.com would map to a different IP address which is load-balanced between the three site 2 IP addresses. This configuration consumes eight public IP addresses.

Extending this logic for m web sites and n web servers, $m * (n+1)$ public IP addresses are consumed. This is a very wasteful way to use IP addresses. A better implementation uses virtual sites for customers with only one IP address per server. The web server produces content depending on the HTTP host request header [17]. As a result, for m web sites on n servers, only $n+1$ IP addresses are consumed (including one virtual address per site). The shortage of IP address space dictates using this technique. Some registries will not accept virtual sites as an excuse for requesting additional IP addresses.

ADDRESSING DESIGN DECISIONS AT INTEL® ONLINE SERVICES

In this section, we discuss some of the implementation choices made at Intel Online Services that were driven by IP address space concerns. We first talk about the choice of IP address space. We then discuss remote access implementations, and conclude with a description of virtual web site implementations.

Addressing Choices

As we mentioned above, some types of address space are more appropriate than others in different situations. At Intel Online Services (IOS), we use a hybrid addressing approach for the data center network that uses the most appropriate type of address space depending on the

purpose of the host. We use private IP addressing for the data center internal networks and for customer servers that do not need to talk to the Internet. Since our internal networks do not need to talk to the Internet, there is no need to use precious public space. Also, back-end servers that do not need to talk to the Internet gain a measure of security because they are impossible to access directly from the Internet. (They are not immune to attack, however.)

Links to transit ISPs and other ISPs that IOS is peers with uses the address space of those ISPs on the router interfaces of the links. Since the IP address will need to change if the ISPs change and will go away if the ISP is no longer used, use of their space in this situation is not a problem and helps conserve IOS public address space.

For data center hosts and routers that need to communicate directly with the Internet, we have used a variety of address spaces. In North America, we use a class B (/16) that was made available to us. Using available public space allows us to have address space independent of our ISP selection, and it makes multihoming to multiple ISPs much easier. For our data centers in Asia and Europe, we have obtained space from the regional address registries that we own. Using the available class B was not feasible because some ISPs will not accept router advertisements of pieces of traditional class B address space. If we choose to use this space, we would have had to use the same ISPs in Asia, Europe, and in North America in order to have any kind of data center-specific routing policy. The process of obtaining new space took significant effort, but it is well worth it to have address space independent of ISP choice and the ability to multihome.

Size of Address Space Advertised

While IOS data centers are designed to handle thousands of hosts, not all of the hosts need to communicate directly with the Internet. Each data center advertises /19 networks, providing address space for up to 8192 hosts. /19 is the longest prefix that some ISPs will accept. For the data centers in North America using parts of a traditional class B, we also need to advertise the entire /16 network out of two data centers in order to deal with ISPs that do not accept parts of traditional class B networks. These ISPs will route traffic to IOS's ISPs, which will then route it to the individual data centers.

Autonomous System Number Choices

These choices are similar to our choices of address space. For North American data centers, IOS uses an autonomous system number that it had available. Separate autonomous numbers for each data center were considered wasteful, and in the North American environment, not particularly necessary. For IOS Asian and European data centers, we have obtained AS numbers for APNIC and RIPE, respectively. This allows the data

centers independence in ISP selection, and it avoids any possible routing policy conflicts with other data centers on different continents.

Address Allocation Policy

For security and ease of management, IOS has chosen to place each of its customers on separate segments. In doing so, IOS enforces requirements for address utilization that mirror those of the address registries mentioned above. This positions IOS to be able to meet the utilization requirements of the registries when IOS requires more space.

To meet those requirements, IOS uses variable length subnet masks extensively. VLSM allows IOS to assign the appropriate sized subnet to a customer while maintaining utilization requirements. One consequence of this is that for IOS infrastructure hosts that need routing information sent to them, a routing protocol that understands VLSM needs to be used. This eliminates RIP version 1 protocol [3] and basically limits the routing protocol used by hosts to OSPF [4] and RIP version 2 [15].

Remote Access Implementation

IOS data centers offer a variety of remote access options such as Virtual Private Network (VPN), ISDN, and dedicated leased lines (T-1, T-3 etc.) to customers as a way of providing access directly from their networks into their servers hosted at IOS. Some of these options, such as LAN-to-LAN VPNs and leased lines, create network channels into the data center across which customer-end network addressing, that may very likely be in the private IP ranges, becomes visible. In this section, we focus only on these remote access options.

Allowing customer-end IP addressing, whether it is in public or private IP space, inside the data center network makes routing extremely difficult. The data center routing policy needs to account for routing network traffic appropriately to multiple customers' home networks. Furthermore, across multiple customers, the networks at their ends can be spread all across the public and private IP address ranges, leaving little scope for summarizing networks and as a consequence leading to larger routing tables. In addition, private IP conflicts across customers as well as the data center networks need to be resolved. Considering all the above challenges, we made a decision to hide customer-end internal networks from data center internal networks.

In this light, let us discuss the salient features of IOS's remote access infrastructure, which is logically represented in Figure 3.

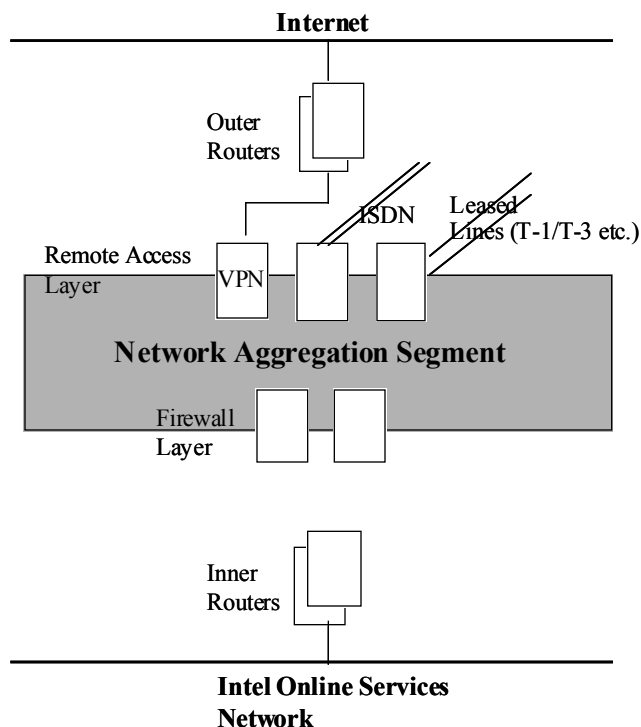


Figure 3: Intel Online Services remote access infrastructure for data center customers

The remote access layer constitutes the devices that provide the various remote access technologies supported at each data center. The firewall layer has devices that enforce security policies, among other things, on remote access traffic. The firewall layer also employs many-to-one as well as static NAT extensively to ensure that customer-end networks are not visible beyond the network aggregation segment. An exception to this rule is when a customer wants to run applications such as the Distributed Component Object Model (DCOM) [16] that does not work across NAT boundaries.

The actual design of the end-to-end network communication, across the above infrastructure, for various customers is dependent on the following factors:

- private IP addressing conflicts with other customers' networks
- where the network communication is initiated
- whether the applications that need to be run can work across NAT boundaries

Following is a discussion of the various remote access scenarios that can be encountered and how IOS supports end-to-end communication in those situations.

Scenario 1: Customer-end network does not conflict with any other customer's home networks or with data center internal networks. All network communication needs to

be initiated from the customer end and inbound into the data center. All required applications work well with NAT.

This is a simple case and can be resolved by translating all of the customer network to one IP address (many-to-one NAT) from the data center's IP addressing range. Each customer network that fits this scenario is translated to a different IP address. As discussed earlier in the NAT section, many-to-one NAT does not allow for communication to be initiated in both directions. In this design, servers in the data center cannot initiate communication back to customer-end systems.

Scenario 2: This is the same as Scenario 1 but it requires that communication needs to be initiated from the data center to a few customer-end systems.

In such situations, we still create a NAT rule to translate all of the customer's networks to one single IP address. In addition, we create static one-to-one NAT relationships between specific data center's IP addresses and specific IP addresses of systems at the customer end. Hence, bi-directional initiation of communication is allowed only for specific customer-end systems.

Scenario 3: Customer-end internal networks conflict with another customer's home network but do not conflict with data center IP address space. Communication is always initiated from customer end to the data center and all required applications work well in spite of NAT.

To resolve the private IP address conflicts here, we ask customers to perform NAT at their end to hide their internal network from IOS behind either non-conflicting private IP address space or public address space. Furthermore, we translate that IP address space, at the firewall layer in the data center remote access infrastructure, to a single IP address from the data center's IP address space.

Scenario 4: This is the same as Scenario 3 except that communication needs to be initiated from the data center to some systems at customer end.

For allowing needed end-to-end communication here, we create static one-to-one NAT relationships for the specific systems at the customer end in addition to the solution implemented for Scenario 3. If customers are using many-to-one NAT on their end as well, they will need to set up corresponding one-to-one NAT relationships on their end in order to allow connections to be initiated from the data center to systems at their end.

Scenario 5: Customer-end networks have private IP conflicts with data center's private IP space.

With the exception of catering to applications that do not work across NAT, we have decided to allocate the data center's public IP space to the internal networks that such customers would need to reach at the data center. Customers in turn will need to perform NAT at their end

to translate their networks to a non-conflicting IP space. This non-conflicting IP space is still hidden from the data center for reasons discussed earlier.

Scenario 6: Customer needs to run applications that do not work across a NAT boundary across one of the remote access channels.

In such cases, we must ensure that the data center's networks as well as the customer-end systems that need such communication use public IP addressing. In addition, we have to accommodate the routing for customer's public IP address range in the data center routing policy as it is not possible to hide customer-end networks behind NAT here.

Other Options: End-to-end connectivity design, across the remote access channels discussed in this section, can get quite complex and potentially difficult to troubleshoot. IOS also provides remote access options, such as client-to-LAN VPN and ISDN that provide direct connectivity from an individual desktop system to the data center network and are devoid of design complexities of the remote access options discussed so far. These remote access options hide the customer end IP addressing from the data center network, eliminating the possibility of address conflicts and the need for technologies such as NAT. Instead, these technologies assign an additional IP address from a designated pool of addresses in the data center's IP address space to the customer-end system while the system is connected. This IP address is used for all communication with systems within the data center. Applications that carry IP address or application port information in the packet payload work without any issue across these remote access channels, since the IP address and port information never changes anywhere between the customer desktop system and the systems in the data center.

This option works well for customers who travel a lot and access their systems from a number of locations or for customers who do not need continuous access to their systems. It does not work well for customers whose connectivity to their systems in the data center must be up all of the time.

Virtual Web Server Implementations

IOS uses the HTTP 1.1 host header technique [18] mentioned above for virtual web site public addresses. This means for m virtual web sites and n web servers, only $n+1$ public addresses are used. With this implementation, the number of IP addresses required is not sensitive to the number of virtual web sites. Some IOS customers have web usage analysis packages that require that different virtual web sites have different IP addresses. IOS minimizes the impact on address space from such customers by mapping IP virtual addresses to web servers on ports other than the HTTP standard port 80. A different web server virtual IP address might map

to port 81 on the web servers, while a different virtual site's virtual IP address might map to port 82 on those same servers. In this way, for m virtual web sites and n web servers, $m + n$ public addresses are used. This is not as good as the host header implementation, but is much better than the IP address per web server implementation.

TRENDS IN IP ADDRESSING

IP address space is finite, and as the number of available addresses gets smaller, new addresses and AS numbers will be harder to obtain. As of February 2000, about half of the available IP addresses will be utilized [19]. We anticipate that depletion of addresses will accelerate as the Internet becomes more pervasive worldwide and as more and more devices (cellular phones, game consoles) are becoming able to communicate directly on the Internet.

One technical solution we are evaluating to reduce the number of addresses being used is to use NAT on public web servers. With NAT, a public virtual IP address could be mapped to multiple private addresses, reducing the need for a public IP address per web server. This solution has the potential drawback of complicating monitoring of web servers and debugging of problems since the individual web servers can no longer be individually contacted by the Internet. Also, various security schemes can be broken by using NAT [20].

A longer term solution to the depletion problem is for the Internet to move to IP version 6 [21]. The address space for IPv6 is much much larger, and many of the actions necessitated by IPv4 address scarcity will not be necessary. The address registries have already begun allocating IPv6 address space. Unfortunately, IPv6 is not backward compatible. At the moment, there is not enough economic incentive to undertake large scale conversion to IPv6. We anticipate that this may change as the amount of IPv4 address space is depleted.

CONCLUSION

IP address space is clearly one of the most critical resources that an Internet data center needs to manage. The fact that IP address space is a limited resource drives many technology and operational decisions. Even private address space, once thought to provide relief from addressing problems, can be the source of problems as two organizations find themselves trying to address private space address collisions. IPv6 holds some promise for relieving many of these problems, but the Internet has quite a ways to go before there is widespread adoption of IPv6.

ACKNOWLEDGMENTS

We acknowledge our technical reviewers, Sanjay Rungta and Matt W. Baker for their time and helpful feedback.

REFERENCES

- [1] Postel, J. (editor), "DARPA Internet Program Protocol Specification," *RFC 791*, September 1981, <http://www.ietf.org/rfc/rfc0791.txt>.
- [2] Hawkinson, J., and T. Bates, "Guidelines for Creation, Selection, and Registration of an Autonomous System (AS)," *RFC 1930*, September 1996, <http://www.ietf.org/rfc/rfc1930.txt>.
- [3] Hedrick, C., "Routing Information Protocol," *RFC 1058*, June 1988, <http://www.ietf.org/rfc/rfc1058.txt>.
- [4] Moy, J., "OSPF Version 2," *RFC 2328*, April 1988, <http://www.ietf.org/rfc/rfc2328.txt>.
- [5] Rekter, Y., and T. Li, "Border Gateway Protocol 4," *RFC 1771*, March 1995, <http://www.ietf.org/rfc/rfc1771.txt>.
- [6] Cerf, V., "IAB Recommended Policy on Distributing Internet Identifier Assignment and Recommended Policy Change to Internet 'Connected' Status," *RFC 1174*, August 1990, <http://www.ietf.org/rfc/rfc1174.txt>.
- [7] Nesser II, P., "An Appeal to the Internet Community to Return Unused IP Networks (Prefixes) to the IANA," *RFC 1917*, August 1990, <http://www.ietf.org/rfc/rfc1917.txt>.
- [8] Manning, B., and T. Pummill, "Variable Length Subnet Table For IPv4," *RFC 1878*, December 1995, <http://www.ietf.org/rfc/rfc1878.txt>.
- [9] Mogul, J. and J. Postel, "Internet Standard Subnet Procedure," *RFC 950*, August 1990, <http://www.ietf.org/rfc/rfc0950.txt>.
- [10] Rekhter, Y., B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," *RFC 1918*, February 1996, <http://www.ietf.org/rfc/rfc1918.txt>.
- [11] Rekter, Y. and T. Li, "An Architecture for IP Address Allocation with CIDR," *RFC 1518*, September 1993, <http://www.ietf.org/rfc/rfc1518.txt>.
- [12] Fuller, V., T. Li, J. Yu, and K. Varadhan, "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy," *RFC 1519*, September 1993, <http://www.ietf.org/rfc/rfc1519.txt>.
- [13] Cisco Corporation, *Hot Standby Routing Protocol*. <http://www.cisco.com/warp/public/619/index.shtml>
- [14] Evegang, K. and P. Francis, "The IP Network Address Translator (NAT)," *RFC 1631*, May 1994, <http://www.ietf.org/rfc/rfc1631.txt>.
- [15] Malkin, G., "RIP Version 2," *RFC 2453*, November 1998, <http://www.ietf.org/rfc/rfc2453.txt>.

- [16] Microsoft Corporation
http://msdn.microsoft.com/library/backgrnd/html/msdn_dco_mfirewall.htm
- [17] Srisuresh P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations," *RFC 2663*, August 1999,
<http://www.ietf.org/rfc/rfc2663.txt>.
- [18] Fielding, R., J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1," *RFC 2068*, January 1997,
<http://www.ietf.org/rfc/rfc2068.txt>.
- [19] Mankin, A., "Will there be a IPv6 Transition?" 1999 USENIX Annual Technical Conference, Monterey, 1999,
<http://www.east.isi.edu/~mankin/usenix99/tsld001.htm>
- [20] Carpenter, B., "Internet Transparency," *RFC 2775*, February 2000, <http://www.ietf.org/rfc/rfc2775.txt>.
- [21] Deering, S., "Internet Protocol, Version 6 (IPv6) Specification," *RFC 2460*, December 1998,
<http://www.ietf.org/rfc/rfc2460.txt>.

Legal notices at
<http://developer.intel.com/sites/developer/tradmarx.htm>.

AUTHORS' BIOGRAPHIES

Jeff Sedayao is a network engineer at Intel Online Services. Between 1987 and 1999, he was the architect of and ran Intel's Internet connectivity. His primary interests are in Internet performance, security, and policy implementation. He also serves as Intel's representative to the Cross Industries Working Team's Internet Performance Team and has participated in the IETF's IP Performance Metrics Workgroup. His e-mail address is jeff.sedayao@intel.com.

Lynne Marchi received her B.S. degree in computer science from California State University, Sacramento. She began work at Intel in 1992 as a co-op and then joined the Corporate Information Security group in 1993. In 1996, she joined Internet Connectivity Engineering where she focused on the secure implementation of new firewalls. Recently, she has joined the Internet Firewall Engineering group of Intel Online Services. Her e-mail address is lynne.c.marchi@intel.com.

Sridhar Mahankali is an M.S. graduate in electrical and computer engineering from the University of Rhode Island, Kingston. He worked as a systems and network engineer with the Board of Governors of the Federal Reserve System before joining Intel's e-Business group in 1998, where he focused on firewall and intrusion detection implementation. Currently, he is part of the Internet Firewall Engineering group in Intel Online Services and specializes in engineering VPNs and network consulting. His e-mail address is sridhar.mahankali@intel.com.

Copyright © Intel Corporation 2000. This publication was downloaded from <http://developer.intel.com/>.